



# MANAGEMENT OF DESIGN ITERATIONS ON COUPLED PARAMETERS IN DESIGN TEAMWORK USING MULTIPLE DOMAIN MATRIX AND COLOURED PETRI NETS

J. Juranić, N. Pavković ✉ and D. Jurinić

University of Zagreb, Croatia

✉ [neven.pavkovic@fsb.hr](mailto:neven.pavkovic@fsb.hr)

## Abstract

A new way of structuring and interpretation of multiple domain matrix is proposed as the basis for categorisation of design parameter relations complexity. Depending on the kind and the degree of coupling of the parameters, the developed methodology activates the appropriate coloured Petri net (CPN) models for semi-automatic support of communication between the members of the design team. The proposed extension of MDM combined with CPN is a novel approach to predicting and managing communication patterns necessary during teamwork coordination on critical interfaces between product components.

*Keywords:* design process, design teams, design structure matrix (DSM), multiple domain matrix (MDM), coloured petri nets

## 1. Introduction

Matrix-based methods for identification and visualisation of relationships and dependencies between design objects reached a high maturity level and have found their place in everyday design practice in many companies to support dealing with complexity. However, those methods may be characterised as mainly static in the context that only the relationship existence is identified and recorded. For some categories of design objects, this is sufficient during the whole design process. In contrast, for some other design object categories, there are evident needs to trace and manage several aspects of design process dynamics that are tightly connected with those relationships. The product development projects whose scope and complexity requires collocated or distributed teamwork are usually divided into subsystems (or hierarchy of tasks) allocated to different team members. Product division into subsystems and corresponding design task allocation often inevitably results with sets of design parameters that will be coupled among two or more design team members. That means either that those team members have to establish a hierarchy of responsibility or sequence for the determination and change management of parameter values during design iterations or, more frequent, that they have to intensively and timely communicate and negotiate about the coupled (shared) values among them. Equally important is to timely (ASAP) update all the other interested team members with changed values and, preferably, the reasoning behind (Karniel and Reich, 2011).

The identification and visualization of the parameters coupled between multiple team members in this paper are modelled by Multiple Domain Matrix (MDM) (Browning, 2016; Lindemann et al., 2009) in which domains are the designers, the subsystems of a complex system being designed, and the design

parameters. The focus of the presented study is on the parameters whose values must be shared by team members, most often being coupled as actually the same dimensions belonging to different parts on interfaces between product components. Such parameters could be interdependent in multiple ways, as shown in Figure 2 and Figure 4 used as a case study presented in Section 4 and Section 5.

The presented research aims to explore how to semi-automatically facilitate and manage the dynamics of design team communication on such multiple coupled design parameters whose relationships are initially recorded using MDM comprising previously mentioned three domains.

We expect that in this way some misunderstandings, delays and additional unnecessary iterations can be avoided, which often happens in practice because of untimely or missing communication. We can find examples in the literature (Clarkson and Hamilton, 2000; Eckert et al., 2001; Flanagan et al., 2003) showing that in teamwork, designers are often unaware or neglect the existence of these connections and consequently the need for timely communication. Such problems are most often the result of insufficient or inadequate implementation of complexity management methods. Such thinking leads to the following research questions:

- How to upgrade and combine MDM (representing product and design process architecture) with a process management model to comprise the dynamic aspects of managing the values of coupled parameters during the design iterations in teamwork?
- What are the advantages and possible obstacles of implementing such an approach if MDM is combined with Coloured Petri Nets (CPN)?

The basic idea of the presented research is to apply, adequately modify and extend the methodological framework for modelling the patterns of engineering design collaboration activities (Juranic et al., 2019) which is based on Coloured Petri Nets. The presented approach upgrades the static representation of design parameter relations with the management of design iterations' dynamics in which the values of multi-dependent parameters are frequently changed by different design team members. The paper is structured in following sections: background and related work, basics of CPN, methodology, industry application example, detailed description of the proposed framework and process model, discussion of results and analysis of benefits comparing to the current industry practice, conclusions and further work in the context of broader research scope.

## 2. Background and related work

The motivation for this work is based on discussions with design researchers from the automotive industry who highlight the static nature as one of the obstacles for MDM and DSM usage in everyday practice. The development of complex design parameter management system (Toepfer and Naumann, 2016) is an example of an effort to fully manage design parameter iterations. Such an approach shows that the initial recording of design parameter relationships should be upgraded by using the methods and tools to manage the iterative determination and especially the timely design team coordination on values of coupled parameters.

Based on the initial representation of the designer-component relationships, Sosa (2008) developed a model that predicts the interactions of the designers around the interfaces of the individual product components. His work is oriented to software development, so it does not focus on parameters but seeks to help project managers to encourage timely communication of team members. In a very similar way, the approach proposed in this paper extracts design parameters coupled between multiple designers, which automatically implies the need for their communication. The presented methodology aims to partially automate the communication and negotiation process within the design team in the iterative coordination of the coupled parameters values.

König et al. (2008) investigated how a process can be modelled to be analysed using a Multiple-Domain Matrix. They conclude that further work needs to look into how interaction with such models can be made more intuitive, above all, to extend MDM from a tool that is oriented to the analysis of structures to a methodology that embraces all aspects of managing a complex process structure.

Kreimeyer et al. (2009) suggested an approach to combine the analytical advantages of matrix-based notation with the modelling capabilities of graphical notation to model process flows, including logic operators. The matrix-based notation is exemplified by Multiple Domain Matrices (MDM) and

graphical notation by Event-driven Process Chains (EPC). Similarly, in this research, we combine MDM with graphical and programming features of Coloured Petri Nets to develop a novel contribution to computer supported collaborative design technologies.

Brisco et al. (2019) presented the first systematic method for engineering design teams to evaluate and select the most suitable computer supported collaborative design technologies (CSCD). The authors compared technology functionality and project requirements established in peer-reviewed literature. Their paper presents 220 factors that influence successful CSCD.

## 2.1. Coloured Petri Nets

Coloured Petri Nets (also called CP-nets or CPN) (Jensen and Kristensen, 2009) is a discrete-event modelling language combining Petri nets with the functional programming language Standard ML. Petri nets provide the foundation of the graphical notation and the basic primitives for modelling concurrency, communication, and synchronisation. Standard ML provides the primitives for the definition of data types, describing data manipulation, and for creating compact and parameterisable models. Tokens in a CPN have a data value and have a timestamp attached to them. The data value often called token colour, describes the properties of the object modelled by the token. The timestamp indicates the earliest time at which the token may be consumed. An example of simple CPN nets is shown in Figure 1. In the net, two places may have float type tokens and one may have integer tokens. When the transition is fired, it will perform a predefined action. For this research, “CPN Tools” software tool has been used to create high-level Petri nets. It is a toolset providing support for the modelling, analysis and simulation of all Petri nets extensions (Jensen et al., 2007).

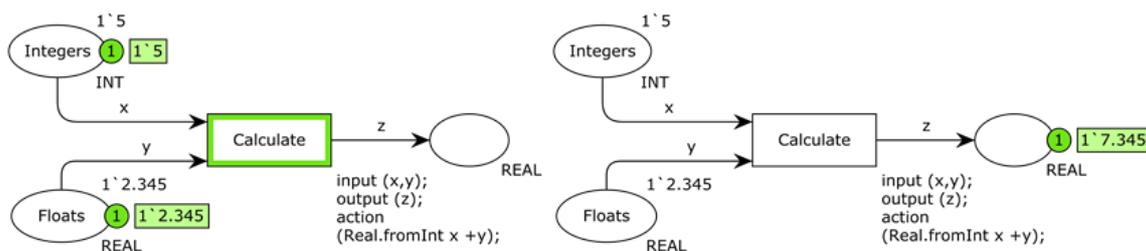


Figure 1. Two states of a simple CPN model

## 3. Methodology

From a methodological viewpoint, this work builds on research on the application of CPN models for collaboration and reasoning activities in engineering design developed by Juranic et al. (2017, 2019). In that article, the authors showed that it is possible to control the process of communication semi-automatically. The framework proposed in Juranic et al. (2019) is adjusted and extended to use the data about the process and product architecture stored in MDM. This data structure is a starting point for planning and initialization of a process where CPN models support the dynamics of the communication process during the new product development, preferably for variant design.

The research is focused on embodiment design phase, where multiple domain matrix with three domains was studied (Figure 3): *designers* (organizational affiliation), *subassemblies* (product architecture) and *design parameters* (objects of potential designers' interactions).

To reduce the complexity of the studied process, it has been assumed that each subassembly is designed by only one designer. The developed methodological process starts with a software tool which automatically recognizes and extracts from MDM the relationships between parameters that are coupled between two or more designers. Based on these relationships, the tool generates a list of potential interactions between designers, including all the necessary data for further processing using the proposed CPN-based framework, which is in detail described in Section 5.4.

Such a list is a basis for executing a semi-automatic management process that supports the dynamics of iterative determination of coupled design parameter values. The framework (implemented in Java) runs a cyclic process designed as a conditional loop. This methodology is applied only to the relations among parameters which are meaningful both for the responsible designer and for other designers who

will also use or share the value of that parameter and consequently be involved in communication about that parameter. The goal of the developed methodology is to take partial control over the collaboration and communication process and thus help the designers with iterations on coupled parameters. Semi-automatic guidance of designers through the collaboration process should evade them from the long-lasting and exhausting negotiations about final parameter values. Continued on a static representation of the relations between design parameters, an innovative method of design iteration process modelling, monitoring and traceability is proposed and studied.

## 4. Industrial application example

To demonstrate and verify the effectiveness of the described methodology, we have selected an example of a relatively complex design project in which the authors also have participated as a part of cooperation with an industrial partner. The project task has been defined as the development of a submersible remotely operated device for inspection of welds in a nuclear reactor pressure vessel (RPV). The welds in the RPV have to be periodically examined in order to find if micro-cracks have appeared and/or propagated. Inspection is being done with non-destructive testing methods such as ultrasound or eddy current testing. The industrial partner is renowned for technological and service excellence in the nuclear industry, providing systems for nuclear power plant examination and repair services, supported by intensive research and development programmes. The following list of most challenging requirements illustrate the project complexity:

- the device should be able to move through the water inside the RPV in all directions and rotate about the vertical axis, near-neutral buoyancy should be achieved;
- linear velocity in all direction should be at least 100 mm/s, as well as scanning speed. Scanning should be made simultaneously along vertical and horizontal axes;
- device should be able to be fixed anywhere inside the vessel;
- two driven axes are needed for surface scanning - one scanning axis and one incremental axis -increment should be 10 mm, the scanning area is 300 mm wide along the weld.

After the completion of the conceptual design phase, at the beginning of embodiment design, the device has been divided into five main subsystems: vertical rail, horizontal rail, chassis, scanning module and, fixation module. In the described project, each subsystem has been assigned to one designer. These subsystems fulfil different functions but share many sets of coupled design parameters that are multi-dependent.

Figure 2 schematically shows the basic system components and some of the coupled parameters that are dimensions. It is characteristic that most of the coupled relations are the part or feature dimensions on the places where several movable components are joined in the chassis. Other types of coupling relate to masses of parts, total mass and volume with respect to the requirement for neutral buoyancy. Based on such division to subsystems and assigned designers, all crucial design parameters for each subsystem were identified and recorded in MDM shown in Figure 4. Here we are proposing a novel way of structuring the MDM in which we differentiate relations between coupled parameters in four categories, depending on how many parameters and how many designers are involved (or how many designers must share a particular parameter).

## 5. Predicting, deriving and supporting designers' interactions

This section introduces the proposed process and developed methodology, which aims to improve the CSCD in the area of intensive iterative teamwork on resolving coupled parameters during embodiment and detail design of relatively complex products. The process comprises three phases described in following subsections. The methodology is focused on predicting and timely inducing and supporting necessary communication interactions between design team members.

### 5.1. Capturing and structuring the organizational and product architecture

To reduce the number of possible combinations, this proposal of structuring and interpretation of MDM is based on an already mentioned assumption for the studied process - that each subassembly

(or product component) is being designed by only one designer. To illustrate generally such simplified MDM structure, let us consider an organizational structure where the three designers are developing a product consisting of three subassemblies. Figure 3 shows the simple hypothetical affiliation matrix that captures how the designers are affiliated with the design of each of product subassemblies.

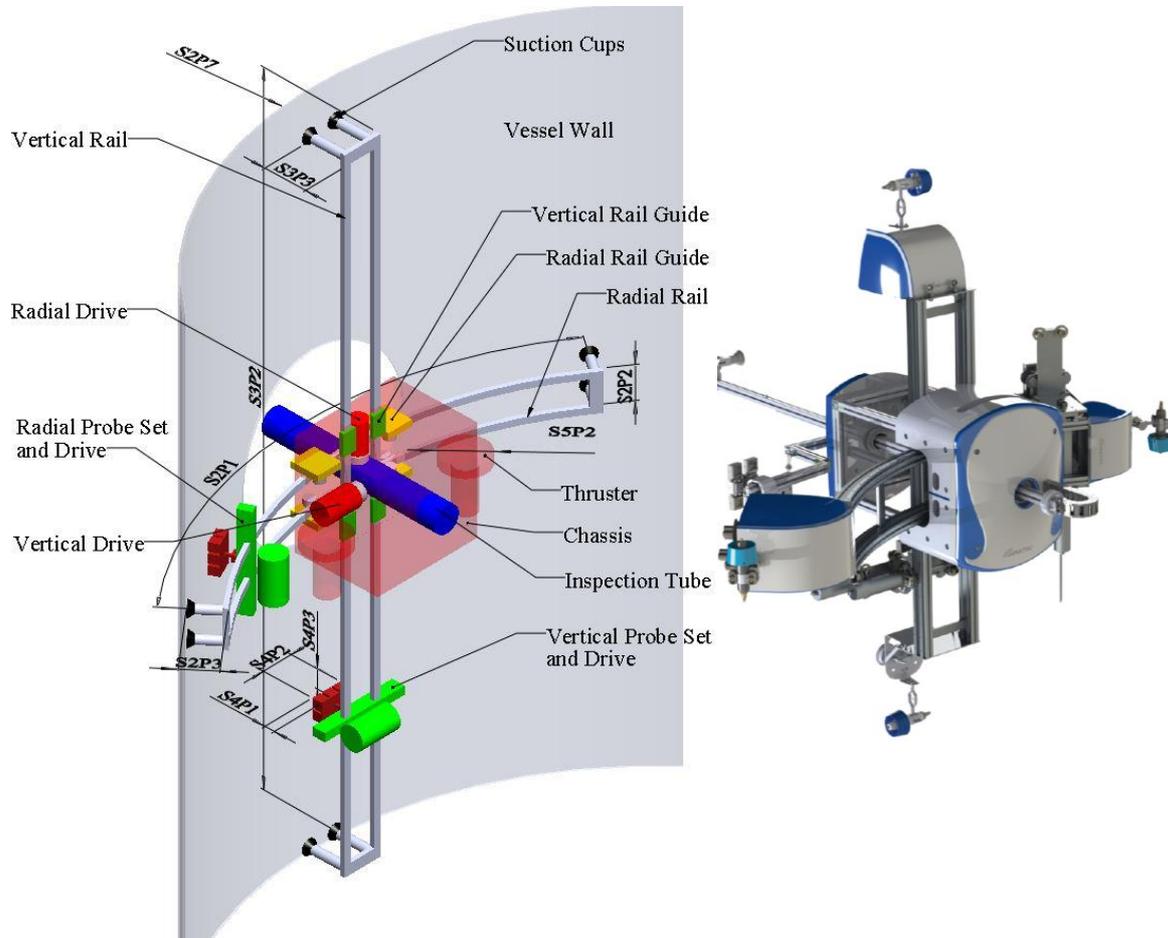


Figure 2. Embodiment design components and coupled dimensions; the finished prototype

	designer 1	designer 2	designer 3	subassembly 1	subassembly 2	subassembly 3	parameter 1	parameter 2	parameter 3	parameter 4	parameter 5	parameter 6
designer 1	X						X	X				
designer 2		X			X				X	X		
designer 3			X			X					X	X
subassembly 1				X			X	X				
subassembly 2					X				X	X		
subassembly 3						X					X	X
parameter 1							X			X		
parameter 2								X			X	
parameter 3									X			
parameter 4									X			X
parameter 5										X		
parameter 6											X	

Legend:  
 Green domain - organizational affiliation  
 Yellow domain - product architecture  
 Blue Domain - design parameters

Figure 3. The proposed structure of MDM as the basis for management of coupled parameters



parameters is not complex since a second designer should just wait for the value of the parameter from the first designer. In such cases, the most common communication is about why the value is not known yet, and when it will become available.

- *Parameters from different subassemblies which are coupled* (marked with ‘D’). Coupled means that the value of one parameter could not be calculated without the value of the second one, while at the same moment, the second value could not be known if the first value is unknown. If these parameters originate from different subassemblies, designers ought to collaborate, usually to negotiate during several iterations to find the compromise solution.
- *Multiple coupled parameters – when values of several parameters (more than two) must be shared between two or more designers.* This kind of relationship has not been found for the observed product, but theoretically, such situations might occur.

### 5.3. Determining the potential and necessary interactions between designers

The proposed form of a matrix with categorized relations is the basis for the further process in which the necessary interactions between the designers should be anticipated in order to timely initiate, stimulate and support their communication. In the first phase, interactions based on category C relations were extracted, from which we obtain a list of sequential parameter determination shown in Table 1. This is followed by the generation of a list of interactions based on the coupled parameters (category D) shown in Table 2. For research purposes, the MDM was created in Excel, while the procedures for identifying and marking types of interactions were created in Visual Basic.

**Table 1. A segment of list of sequentially related parameters interactions**

No.	Category	Request by	Dependent parameter value	Requested from	Known parameter value	Status
1	C	Designer 1	S1 P7	Designer 2	S2 P2	Completed
2	C	Designer 1	S1 P4	Designer 3	S3 P1	Completed
3	C	Designer 3	S3 P4	Designer 1	S1 P5	Completed
4	C	Designer 4	S4 P4	Designer 2	S2 P3	Not started
.....						

**Table 2. A segment of list of coupled parameters interactions**

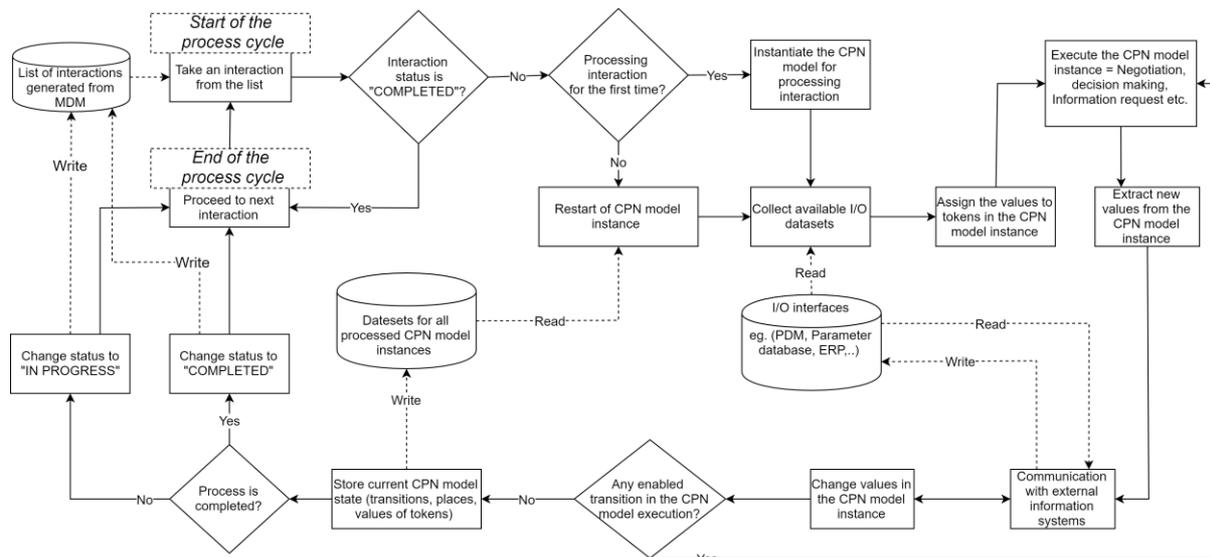
No.	Category	Designer A	Designer B	Parameter A	Parameter B	Status
1	D	Designer 2	Designer 4	S2 P3	S4 P1	Not started
2	D	Designer 2	Designer 4	S2 P3	S4 P4	In progress
3	D	Designer 3	Designer 4	S3 P3	S4 P1	Completed
.....						

Finally, to support and partially manage the designers’ collaboration on relations of categories C and D, the framework described in the following section has been developed. The interaction lists shown in Table 1 and Table 2 serve as inputs for instantiating and triggering the appropriate CPN models.

### 5.4. Framework for managing iterations on coupled parameters

Figure 5 shows the flow of the cyclic execution process of engineering design collaboration activities. The main feature of the developed processing algorithm is that each instantiated CPN model could be stopped and restarted several times if necessary. If the model is stopped, its state is saved at the end of the processing cycle. In the next cycle, it is restarted again after the previously saved state is loaded. Such a solution, which divides and runs the whole process in discrete steps, was chosen because a larger number of simultaneously active CPN models, which are relatively complex, would require

excessive computer resources. More details about the developed framework can be found in [Juranić et al. \(2019\)](#). The process runs continuously, assuming that all changes (parameter values and relations) are updated immediately. Each cycle starts with reading all necessary data about a particular designers interaction from the list generated from MDM.



**Figure 5. Framework for design parameter collaboration supported by CPN**

The process further checks if the interaction has a status “completed”. The interaction has completed status if all designers agreed on the final parameter value (usually after several iterations). If the status is “completed”, this interaction will be excluded from further processing and if not, the process will:

- instantiate the corresponding CPN model (for interactions that are processed for the first time)
- or restart CPN model (for already processed interactions).

Each time CPN model executes, it collects all available data at that moment to assign data to tokens in order to execute the model. During execution, the CPN model will calculate new values, initialize needed collaboration processes or help with communication among stakeholders. After execution, all values from CPN tokens are stored in according databases. CPN model execution will be stopped if one of two events happen:

- When all transitions were fired as well as the last one – the process is completed (finished).
- When some transitions are still not executed, because there are places (which precede those transitions) that do not have all the required tokens.

Tokens are missing if no relevant data existed in the database in the moment of collecting. If the CPN model is not fully completed, the process will store the current state of the CPN model in order to skip several steps in execution next time the model is executed. Depending on the status at the end of execution, the currently processed interaction gets status “completed” or “in progress”.

As noted, in each cycle of the framework, the particular CPN model is executed. We have created several CPN models, but due to paper length limitation, only one model will be presented here. This model supports negotiation about parameter value among designers. The framework will collect all the relevant data for processed interaction and instantiate the model. The model will further find corresponding designers (negotiators) and a person who will observe the process and make a final decision if needed. Before the negotiation begins, the model will check if there is available solution space which could be calculated in order to define parameter value limits to help the designer and automatically put aside values which are not compatible with other parameters. If a solution space does not exist, this step is skipped. The negotiation starts with an initial proposal, which is sent to all stakeholders. After the responses are received, they are automatically processed, and all stakeholders are asked for their opinion. This inner cycle could run several times until a common decision is made.

If a decision could not be made, the decision-maker will be invoked to help with the decision. After the final decision is made, the results are checked, and all stakeholders should agree with the results. Beneath each transition in the CPN model, there is a subroutine which enables the negotiation process. That part of the CPN model is not shown here.

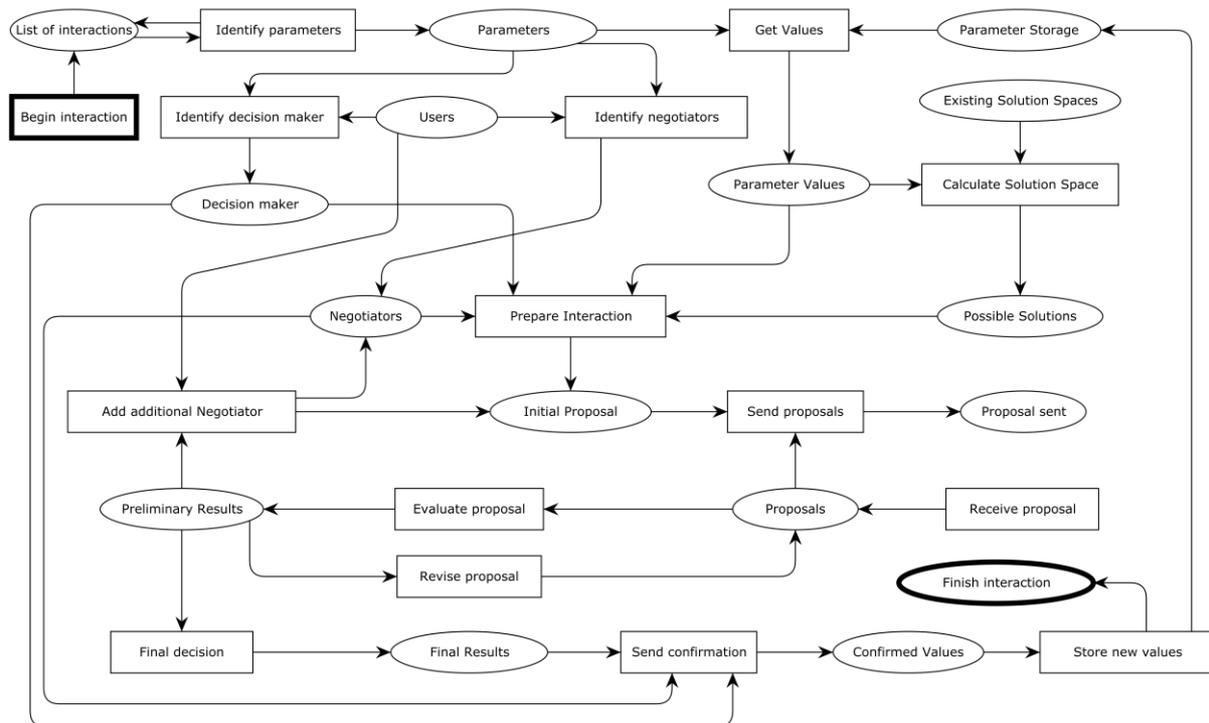


Figure 6. Coloured Petri Net model of process of resolving coupled parameters

## 6. Discussion and conclusion

The paper proposes a new way of structuring and interpretation of MDM based on which the categorisation of design parameter relations complexity has been developed. Depending on the kind and the degree of coupling of the individual parameters, the proposed methodology activates the appropriate CPN models for semi-automatic support of communication management between the members of the design team. Communication is supported by a customised framework that has been developed and realised in previous research (Juranić et al., 2019).

We consider the presented method of MDM restructuring and classification of relations between coupled parameters as a useful innovative contribution because it enables us to distinguish and adequately organize necessary communication channels and processes. The proposed methodology for upgrading MDMs and linking them to CPN models is the answer to the first research question.

The novelty of the presented approach is that we focus on the parameters that are coupled among two (or more) designers. We believe that the coupled parameters that are within the area of responsibility of only one designer are far less demanding for project management. Hence they should not cause delays and additional iterations arising from known communication problems in teamwork (Eckert et al., 2001).

At this research stage, the sequence of resolving sets (in the studied case pairs) of parameters coupled between multiple designers was not considered. This is planned as the future work, as well as a generalization of the proposed methodology by abandoning the assumption that only one designer works on a single subassembly. This will significantly increase the complexity of the methodology but might achieve a significant contribution to the development of CSCD technology.

Concerning the second research question, the main obstacle in industrial implementation might be to coordinate several simultaneously active CPN models on the level of the whole design project management. This issue could be partially solved by further extending MDM to visualize the state of the process through cell colours and possibly additional links to diagrammatic representations of the

state of interactions and iterations in real time. This way, it could be visualized which communication (interaction) has not yet started, which is in progress and which is finished. If there are requirements to record and display traceability, a log file can easily be obtained from the CPN model which would display the history of changes in parameter values through the performed iterations.

To conclude, the contribution of this paper reflects in a novel approach to predicting, classifying and managing communication patterns that are necessary during teamwork coordination on critical interfaces between product components developed by different team members.

Brisco et al. (2019) presented 220 factors that influence successful CSCD. They summarised categorised factors into 19 statements which represent CSCD requirements. We believe that the methodology presented in our paper corresponds with at least the following three statements:

- “Supports communication through synchronous and asynchronous multi-threaded and multi-channel software for prompt discussion in a way which supports the context of the message.”
- “Allows for greater productivity through fast objective focused communication, organisation of work and a greater quantity of output to promote collaboration readiness, reflection and reduced rework time.”
- “Encourages a shared understanding by defining and framing conversations within a common context which makes it easy to understand information, clarify meaning and reduce miscommunications.”

## References

- Brisco, R., Whitfield, R.I. and Grierson, H. (2019), “A novel systematic method to evaluate computer-supported collaborative design technologies”, *Research in Engineering Design*, <https://doi.org/10.1007/s00163-019-00323-7>
- Browning, T.R. (2016), “Design Structure Matrix Extensions and Innovations: A Survey and New Opportunities”, *IEEE Transactions on Engineering Management*, Vol. 63 No. 1, pp. 27-52. <https://doi.org/10.1109/TEM.2015.2491283>
- Clarkson, P.J. and Hamilton, J.R. (2000), “Signposting”, a parameter-driven task-based model of the design process”, *Research in Engineering Design*, Vol. 12 No. 1, pp. 18-38. <https://doi.org/10.1007/s001630050021>
- Eckert, C., Clarkson, J. and Stacey, M. (2001), “Information flow in engineering companies: Problems and their causes”, *Proceedings of the 13th Int. Conf. on Engineering Design (ICED'01)*, Glasgow, UK, pp. 43-50.
- Flanagan, T.L., Eckert, C.M. and Clarkson, P.J. (2003), “Parameter trails”, *Proceedings of the 14th International Conference on Engineering Design*, Stockholm, Sweden, 2003, Design Society, Glasgow, pp. 1-10.
- Jensen, K., Kristensen, L.M. and Wells, L. (2007). “Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems” *International Journal on Software Tools for Technology Transfer*, Vol. 9 No. 3-4, pp. 213-254. <https://doi.org/10.1007/s10009-007-0038-x>
- Jensen, K. and Kristensen, L.M. (2009), *Coloured Petri Nets*, Vol. 1, Springer Berlin Heidelberg, Berlin, Heidelberg. <https://doi.org/10.1007/b95112>
- Juranić, J. et al. (2017), “Modelling the design parameters dynamics with Petri nets”, *Proceedings of the 21st Int. Conf. ICED17*, Canada, Design Society, pp. 91-100.
- Juranić, J. et al. (2019), “Patterns of engineering design collaboration and reasoning activities modelled with Coloured Petri Nets”, *Journal of Engineering Design*, Vol. 30 No. 10-12, pp. 563-598, <https://doi.org/10.1080/09544828.2019.1630803>
- Karniel, A. and Reich, Y. (2011), *Managing the Dynamics of New Product Development Processes*, Springer London, London, <https://doi.org/10.1007/978-0-85729-570-5>
- König, C., Kreimeyer, M. and Braun, T. (2008) “Multiple-Domain Matrix as a Framework for Systematic Process Analysis”, *10th International Design Structure Matrix Conference, DSM '08*, Stockholm.
- Kreimeyer, M. et al. (2009), “Extending multiple domain matrices to allow for the modeling of boolean operators in process models”, *ICED'09 August 2009*, Stanford university, USA.
- Lindemann, U., Maurer, M. and Braun, T. (2009), *Structural Complexity Management*, Springer Berlin Heidelberg, Berlin, Heidelberg, <https://doi.org/10.1007/978-3-540-87889-6>
- Sosa, M.E. (2008), “A structured approach to predicting and managing technical interactions in software development”, *Research in Engineering Design*, Vol. 19 No. 1, pp. 47-70, <https://doi.org/10.1007/s00163-007-0039-5>
- Toepfer, F. and Naumann, T. (2016), “Management of vehicle architecture parameters”, *Proceedings of DESIGN 2016, the 14th International Design Conference*, Design Society, Glasgow, UK, pp. 1679-1688.