

Configuring ship locks using a product platform based on DSM methods

Citation for published version (APA):

Knippenberg, S. C. M., Pennings, W. J., Etman, L. F. P., Rooda, J. E., & Vogel, J. A. (2021). Configuring ship locks using a product platform based on DSM methods. In H. Stowe, T. R. Browning, S. D. Eppinger, & J. Trauer (Eds.), *Proceedings of the 23rd International Dependency and Structure Modeling Conference, DSM 2021* (pp. 21-30). Design Society. <https://doi.org/10.35199/dsm2021.3>

DOI:

[10.35199/dsm2021.3](https://doi.org/10.35199/dsm2021.3)

Document status and date:

Published: 01/10/2021

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Configuring Ship Locks Using a Product Platform Based on DSM Methods

S.C.M. Knippenberg¹, W.J. Pennings¹, L.F.P Etman¹, J.E. Rooda¹, J.A. Vogel²

¹ Eindhoven University of Technology, 5600 MB, Eindhoven, The Netherlands

² Rijkswaterstaat, 3526 LA, Utrecht, The Netherlands

Abstract: Rijkswaterstaat (RWS), the executive branch of the Dutch Ministry of Infrastructure and Water Management, has announced that a significant number of ship locks in the Netherlands are due for renovation during the coming decades. The renovation of ship locks allows RWS to reconsider current existing preferences for lock design and to investigate options to develop a product platform for locks. This paper presents a method for the development of such a platform, based on the functions of a lock and the component alternatives to provide that function. By applying design structure matrix (DSM) modeling and value analysis methods, a ranked set of feasible lock configurations can be obtained that meets the specific requirements for a particular lock location while trading against the desired commonality in the family of locks. Preferred configuration attributes are to be incorporated in a product platform for locks to support RWS with the planned lock renovations.

Keywords: Ship locks, Product platform, Component alternatives, Component interface matrix, Value analysis

1 Introduction

Ship locks are essential for the Dutch infrastructure to facilitate waterborne transportation as they regulate differences in water level between waterways. When a ship approaches a lock, the water level inside the lock is adjusted to the water level at the ship side of the lock (Figure 1a). Next, the gate in front of the ship opens and the ship can enter the lock chamber (Figure 1b). When the ship has entered the lock, the gate is closed and the water level is adapted to the level on the other side of the lock (Figure 1c). Finally, the gate towards the waterway on the other side of the lock opens and the ship can continue its way (Figure 1d).

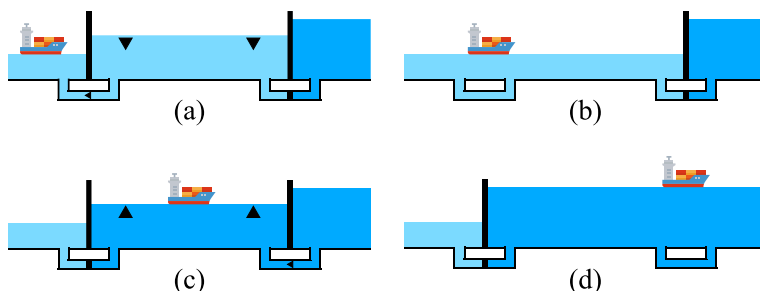


Figure 1: Way of operating of a lock.

Rijkswaterstaat (RWS), the executive branch of the Dutch Ministry of Infrastructure and Water Management, is responsible for 132 locks in the main waterways in the Netherlands. RWS has investigated its lock portfolio and concluded that a major part of the locks reach the end of their predicted lifetime in the upcoming decades or do not meet modern-day standards. Moreover, several locks do not have sufficient capacity to transfer ships, given the projected growth of waterborne transportation. Based on these findings, RWS has planned to renovate a major part of its lock portfolio. To support the renovation of locks, RWS has founded the MultiWaterWerk (MWW) project. The goal of this project is to accommodate the serial renovation process of locks by modularizing and (partially) standardizing lock components, improving the maintainability of the portfolio, and, reducing life cycle costs. The number of locks to be renovated allows RWS to reconsider existing preferences for lock design and to investigate options to develop a product platform for ship locks.

A product platform can be defined as a set of common components, modules, or parts that can be applied to generate a stream of derivative products (Meyer and Lehnerd, 2011). Meyer and Lehnerd state that the development of product families for complex products might reduce the costs of product development. For the development of product families, Du et al. (2014) distinguish scalable and module-based product family design. In scalable family design, the dimensions of a platform (consisting of pre-designed product configurations) can be scaled to satisfy various applications of the platform (Simpson, 2004). The development of a scalable platform for locks has been investigated in Knippenberg et al. (2019), resulting in five basic lock configurations that form a basis for a scalable platform. An alternative for the scalable platform is the module-based platform. In module-based family design, product variants can be configured by combining different modules (Ulrich, 1995). Wilschut et al. (2019) have identified a wide range of functional variety in current lock design, which can be addressed by module-based family design. In prior work, Knippenberg et al. (2020) introduced the first steps of a method for the development of a module-based product platform, by applying design structure matrix (DSM) techniques (Eppinger and Browning, 2012). Meyer and Lehnerd (2011) suggest that a combination of modular and scalable platform leveraging strategies may yield a more effective platform.

To arrive at a product platform for the renovation of locks in the Netherlands, this paper presents a follow-up on the previous work by Knippenberg et al. (2020). This previous work considered in particular the determination of feasible lock configurations from component alternatives and the selection of choices in favor of the lock family. To continue the development of a product platform, the upcoming sections present a method to obtain a ranked set of feasible lock configurations. This method is based on a selection of twelve functions of a lock and the accompanying component alternatives for each function that are considered for inclusion (Section 2). Next, the compatibility of component pairs is presented in Section 3. To determine feasible lock configurations from the compatible component pairs, an algorithm has been developed (Section 4). In Section 5, a value analysis method is discussed to quantitatively rank lock configurations. Subsequently, a lock case study is presented that demonstrates the steps of the proposed method in Section 6. Finally, conclusions about the method for the development of a product platform for locks are presented in Section 7.

2 Functions and components of a lock

Based on the decomposition of a lock as presented by Wilschut et al. (2019) and the lock construction principles presented in Josephus Jitta (1947) and Glerum and Vrijburcht (2000), twelve functional elements have been selected to illustrate the steps in the development of a product platform. These twelve elements are known to have a major influence on the civil-mechanic design of a lock and on the component alternatives for the other elements. The selected elements correspond to the functions of a lock, as considered in earlier work (Knippenberg et al., 2019; Knippenberg et al., 2020). Table 1 presents an overview of the elements and their function within a lock.

Table 1: Twelve functional elements of a lock with their function and the number of considered component alternatives.

ID	Functional element	Function	# Component alternatives
1	Leading jetty	Guiding ships into the lock chamber.	2
2	Leveling system	Leveling water inside the lock chamber.	3
3	Gates	Retaining water during water leveling when closed and allowing ships to pass when open.	7
4	Gate actuators	Facilitating the actuating of gates.	8
5	Leveling system actuators	Facilitating the actuating of the leveling system.	2
6	Lock head	Providing support, turning points, and guides for gates, some leveling systems, and actuators.	24
7	Lock chamber	Isolating a part of the waterway in which ships can be raised or lowered.	2
8	Seepage screens	Preventing construction instabilities caused by water flowing through the soil outside the lock.	1
9	Ice prevention system	Preventing ice to obstruct gates, leveling systems, actuators, and passing ships.	2
10	Soil protection	Protecting the soil outside the lock for propeller turbulences and for the outflow of water.	2
11	Salt/fresh water separation system	Separating fresh and salt water between the waterways separated by a lock.	2
12	Supervisory control system	Controlling the components of a lock to ensure correct and safe dynamic behavior.	2

For each of the functions in Table 1, component alternatives have been identified that present viable technical solutions for the respective function. The number of component alternatives per function has been included in the utmost right column of Table 1. In total, 57 components are considered for the twelve functions. The upcoming section presents the feasibility of combining particular component alternatives to configure the locks.

3 Component interfaces

To provide a structured overview of compatible component pairs, a component interface matrix (CIM) has been constructed. This matrix is based on the compatibility DSM, as presented in the work of Hellenbrand and Lindemann (2008) and Knippenberg et al. (2020). An entry in the CIM indicates the feasibility of combining the respective alternative of one component with the alternative of the second component that shares an interface.

Figure 2 presents the CIM of compatible component pairs to design a lock. The grey blocks on the diagonal of the matrix represent the twelve functions and have been marked with their ID, corresponding to Table 1. The size of the blocks indicates the number of considered component alternatives to provide the respective function. The green entries in Figure 2 represent the compatible component pairs that can be considered to base a module in the platform upon.

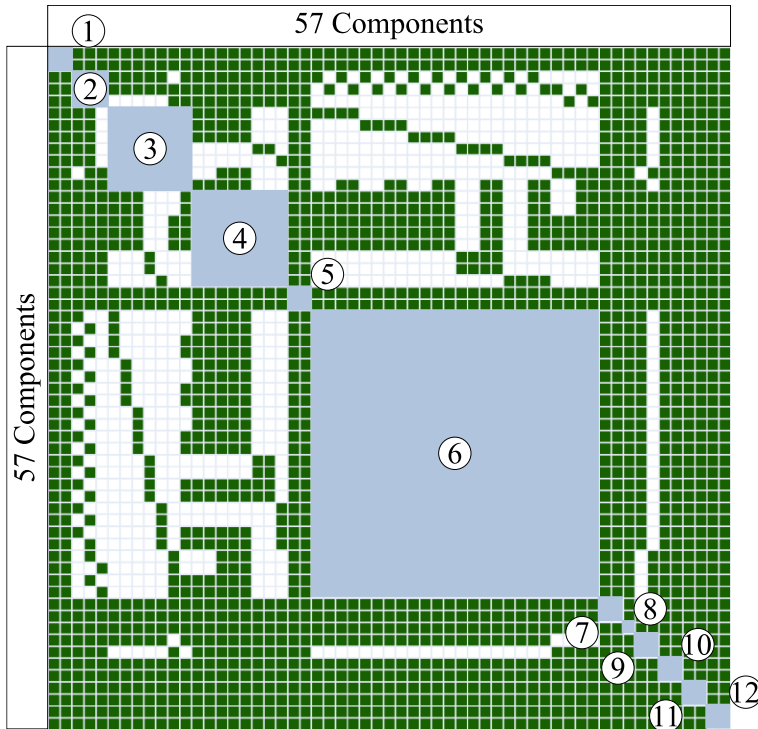


Figure 2: CIM of the 57 component alternatives with compatible component pairs in green for the twelve lock functions indicated by grey blocks.

The next step towards a product platform is to determine for each lock in the portfolio the subset of components that are feasible to configure that specific lock. This subset is derived from object-specific characteristics that determine whether a particular component alternative is allowed for the configuration of a given lock case. Examples of these characteristics include the waterway class, the water level difference over the lock, and the type of water retention. The value ranges of these characteristics have been collected from Glerum and Vrijburcht (2000), RWS documentation, and RWS expert interviews.

To determine feasible configurations of components for a specific lock, an algorithm has been developed that eliminates components from the component subset that are not part of any feasible configuration. In the next section, the principle of this algorithm is elaborated.

4 Configuration finding algorithm

Though the CIM reveals compatibility of component pairs, further analysis is needed to deduce whether compatibility is guaranteed when the system requires a choice of components for more than two functions. If we consider for each function a given component alternative, this requires $\binom{n}{2}$ pair checks for n functions. Consider, for example, a system of three functions A, B and C , as presented in Figure 3. The compatibility of component alternatives regarding interfaces AB, AC and BC must be verified. If any combination is incompatible, the considered configuration of A, B and C is infeasible. Each function must be realized using exactly one component. Since multiple component alternatives may realize a function, compatibility of the various possible configurations of components must be verified. The number of configurations to verify is given by $\prod_{i=1}^n N_i$, with N_i the number of component alternatives for function i . The number of configurations may grow rapidly.

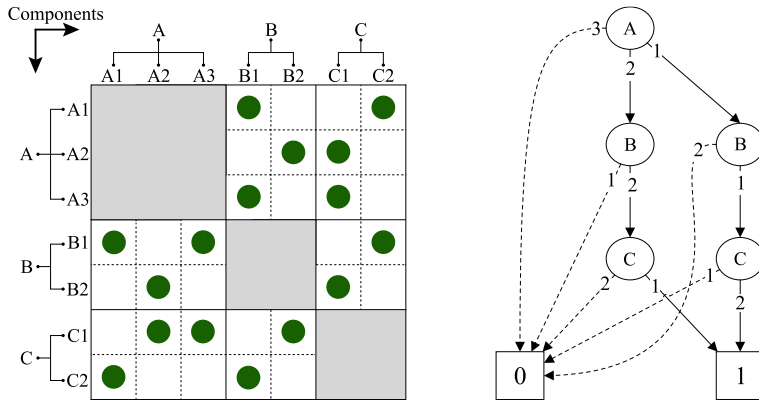


Figure 3: Schematic CIM to illustrate the principle of finding feasible configurations and the feasible (solid lines towards 1) and infeasible (dashed lines towards 0) configurations.

An exhaustive enumeration algorithm is used to find all feasible component configurations for a given CIM for a specific lock. The algorithm first generates a list of all possible lock configurations by calculating the Cartesian product of all sets of admissible components. For example and like in Figure 3, let $A = \{a1, a2, a3\}$, $B = \{b1, b2\}$ and $C = \{c1, c2\}$ with all set members admissible; the Cartesian product equals $A \times B \times C = \{(a1, b1, c1), (a1, b1, c2), (a1, b2, c1), \dots, (a3, b2, c2)\}$. This yields a total of 12 ($3 \times 2 \times 2$) possible configurations.

Subsequently, the algorithm identifies for each possible configuration all pairs of components and verifies the feasibility of the respective combinations using the CIM. The number of pairs of components to be verified for each configuration equals the aforementioned binomial coefficient. If and only if all pairs are compatible, the component configuration is feasible, and it is stored in the list of feasible configurations. For computational efficiency, the algorithm terminates a configuration check as soon as an incompatible pair of components is encountered.

5 Value analysis

The configuration finding algorithm may produce many feasible configurations, which need to be prioritized over one another to obtain the best configuration. We use the value analysis (VA) method of Pahl and Beitz (2007) to quantitatively rank components and subsequently value configurations in their entirety.

This VA method utilizes a hierarchical evaluation criterion structure in which the criteria become progressively more specific, and thus more easily quantifiable, as the level of decomposition increases. Each criterion is assigned a positive real weighting factor (WF) to indicate its relative importance (Figure 4). Weights are assigned through a step-by-step process. At each level, criteria are weighted with respect to their parent objective, subject to the condition that the sum of weights corresponding to a single parent criterion is one. A global WF is obtained by multiplication of a criterion’s weighting factor by that of all its parent criteria.

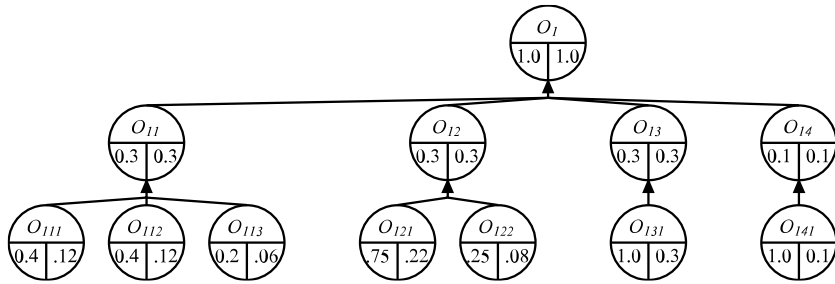


Figure 4: Hierarchical arrangement of criteria O_i , O_{ij} , and O_{ijk} in three decomposition levels, adapted from Pahl and Beitz (2007).

Criteria are scored according to a consistent discrete rating interval that runs from 0 to 4. The measurable parameters that are represented by the criteria must therefore be mapped to this rating interval. The lock-related parameter values are obtained from a lock design handbook (Glerum & Vrijburcht, 2000), RWS documentation, and RWS expert interviews. Consider for instance the gate actuation function. Its corresponding evaluation criteria and weighting factors are shown in Table 2. Six component alternatives are considered for this function: panama wheel, rack and pinion, electromechanical cylinder, electrohydraulic cylinder, servohydraulic cylinder, and cable winch. For each component, reliability data have been collected, resulting in value scores for this criterion of 2, 2, 4, 2, 3, and 2, respectively. Similarly, the other criteria have been scored.

Table 2: Decomposition of value analysis criteria with weighting factors.

Criterion	WF	Sub-criterion	WF	Global WF
Quality	0.30	1: Reliability (failures per year)	0.40	0.12
		2: Availability (uptime over planned uptime)	0.40	0.12
		3: Inspection frequency (per year)	0.20	0.06
Sustainability	0.30	4: Energy usage (kWh/y)	0.75	0.22
		5: Hazardous material leakage risk (e.g. oil)	0.25	0.08
Costs	0.30	6: Life cycle costs (LCC) in present value (€)	1.00	0.30
Adaptability	0.10	7: Ease of moving to a new alternative	1.00	0.10

The adaptability criterion in Table 2 is discussed in more detail since it is calculated using a DSM-based method. A so-called adaptability DSM is composed (Figure 5). The rows and columns represent the component alternatives. A 0–4 score is assigned to each cell, providing a rating for the ease with which the component in the row can be replaced by the component in the column. The scoring guideline that has been used is shown in Table 3. The average of the scores in a particular row yields the adaptability score. Conversely, the average of the scores in a particular column yields an inflexibility score, i.e. the difficulty of moving away from a particular component. It is assumed that the adaptability scores for the gate and leveling system actuators are the same, such that the use-value scores can be applied to both components.

Table 3: Scoring guideline for the adaptability DSM.

Score	Description
0	The existing structure must be replaced in its entirety.
1	Significant parts of the existing structure must be changed or replaced.
2	The amount of modifications or new structures is neither large nor small.
3	The required modifications are simple and can be realized easily and cheaply.
4	No modifications or new structures are required at all.

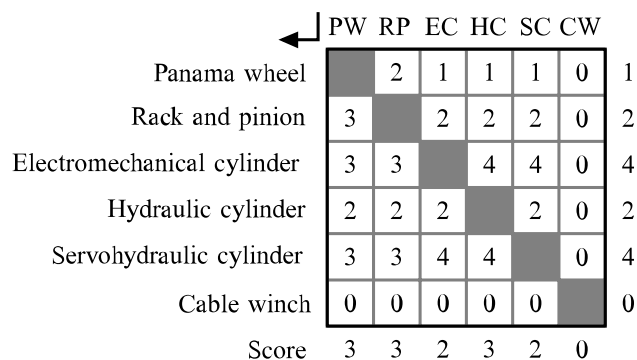


Figure 5: Adaptability DSM of gate actuators.

6 Case study Spooldersluis

In this section, the proposed method is utilized for the Spooldersluis lock complex. The Spooldersluis is one of the locks in the Netherlands that is due for a major renovation in the coming years and therefore relevant to determine its feasible set of configurations following the proposed method.

First, the relevant functions for the Spooldersluis are determined, based on the selection of twelve functions as presented in Table 1. Eleven of these functions are considered to configure the object. The salt/fresh water separation system (ID = 11) is not incorporated since the lock is not located near the sea and therefore does not need the function of separating fresh and salt water. For the eleven functions, a total of 55 components are considered for the configuration of a lock (Table 1). By combining the object-specific characteristics of the Spooldersluis and the CIM of Figure 2, it is determined that 35

components are allowed. The compatible component pairs for the Spooldersluis have been presented in the object-specific CIM of Figure 6. Again, the size of the grey blocks indicates the number of allowed components per function, the functions have been marked with their ID (Table 1), and the green entries represent the compatible component pairs.

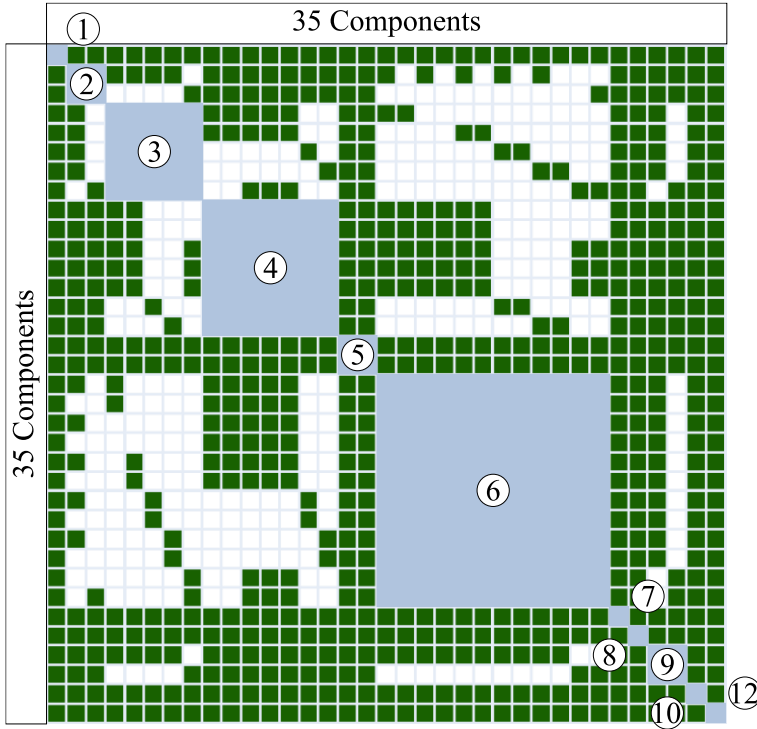


Figure 6: CIM of the Spooldersluis, consisting of 35 component alternatives that have been derived from object-specific characteristics.

To find the set of feasible configurations, the algorithm as presented in Section 4 is applied. The algorithm finds a total of 30 configurations for the Spooldersluis. These configurations incorporate 28 out of the 35 component alternatives and their interactions from the CIM of Figure 6.

From the CIM of Figure 6, it is derived that the component choices for five functions (1, 7, 8, 10, and 12) are fully determined by the object-specific characteristics of the Spooldersluis. The choices for the leveling system and the ice prevention system are influenced by the choice for the gates. For the lock head, the component choice is based on the choice for the gates, the leveling system, and the gate actuators. For the choices of gates, gate actuators, and leveling system actuators, value analysis has been conducted to assess the components to configure the object. The results of this value analysis have been used to rank the configurations, based on their combined component score. The five configurations with the highest scores for the Spooldersluis are presented in Table 4.

Table 4: Five configurations for the Spooldersluis with the highest aggregate value analysis scores.

Configuration and score	Gates (ID = 3)	Gate actuators (ID = 4)	Leveling system actuators (ID = 5)
#1: score 44.8	Single miter gates	Electromechanical cylinder	Electromechanical cylinder
#2: score 29.1	Single miter gates	Rack and pinion	Electromechanical cylinder
#3: score 26.9	Single miter gates	Servohydraulic cylinder	Electromechanical cylinder
#4: score 21.3	Single miter gates	Electromechanical cylinder	Hydraulic cylinder
#5: score 21.3	Single miter gates	Hydraulic cylinder	Electromechanical cylinder

To select the appropriate configuration for the Spooldersluis, the results of Table 4 can be combined with a coverage analysis of the lock portfolio, of which the first steps have been presented in prior work (Knippenberg et al., 2019). The coverage analysis provides insight into the feasibility of applying identical component configurations for multiple locks. The combination of the results of this work and the coverage analysis is considered as future work in developing a product platform for the renovation of locks in the Netherlands.

7 Conclusion

Prior work by Knippenberg et al. (2020) presented the first steps towards a modular product platform for the renovation of locks in the Netherlands. In this earlier work, a method to conduct a coverage analysis of the lock portfolio has been presented, aiming to provide insight into the allowed configuration options for locks in the portfolio of RWS. As a follow-up on this research, we have developed a method to arrive at a set of ranked configurations for the design of a lock.

The proposed method applies DSM modeling and value analysis methods, to determine a ranked set of feasible configurations for locks that meet the object-specific requirements. First, the set of functions of a lock are determined. Next, the set of component alternatives and the compatible component pairs are defined for each of the functions. By confronting these components with the object-specific requirements, the set of allowed component combinations are determined for that specific lock. From this, an enumeration algorithm generates the list of feasible configurations for the lock object. The list of configurations is subsequently ranked by means of a value analysis to compare alternative component choices. The configurations with the highest scores indicate the most suitable configurations for that lock.

To illustrate the proposed method, a case study has been conducted for the Spooldersluis. In this study, a set of 30 lock configurations has been obtained, based on 11 functions, 28 components, and their interfaces. For three functions incorporated in the Spooldersluis, value analysis has been performed to rank the list of configurations. Future work seeks to combine the method presented in this work with a coverage analysis of other locks planned for renovation to develop a product platform for locks that can be applied by RWS to support the renovation process.

8 Acknowledgements

We thank Rijkswaterstaat, part of the Dutch Ministry of Infrastructure and Water Management, for supporting this research. In particular, we thank Harry Lammeretz, Maurice de Graaf, and Maria Angenent for their enthusiastic involvement.

Furthermore, we thank Tim Wilschut from Ratio CASE B.V. for supporting the development of a product platform for locks.

We also thank Professor Steven D. Eppinger for his comments on our DSM 2020 presentation, which helped to shape our follow-up work presented here.

References

- Du, G., Jiao, J., Chen, M., 2014. Joint optimization of product family configuration and scaling design by Stackelberg game. *European Journal of Operational Research* 232 (2014) 330–341.
- Eppinger, S.D., Browning, T.R., 2012. *Design structure matrix methods and applications*. The MIT Press, Cambridge, MA.
- Glerum, A., Vrijburcht, A., 2000. *Design of locks*. Bouwdienst Rijkswaterstaat, Utrecht, The Netherlands.
- Hellenbrand, D., Lindemann, U., 2008. Using the DSM to support the selection of product concepts. *Proceedings of the 10th International DSM Conference (DSM 2008)*, Stockholm, Sweden, 11-12 November 2008, pp. 363-374.
- Josephus Jitta, J.P., 1947. *Sluizen en andere waterbouwkundige kunstwerken in en langs kanalen*. De Erven F. Bohn N.V., Haarlem, The Netherlands. In Dutch.
- Knippenberg, S.C.M., Etman, L.F.P., Rooda, J.E., Houwing, E.J., Vogel, J.A., 2019. Development of a product platform for ship locks using DSM methods. *Proceedings of the 21st International DSM Conference (DSM 2019)*, Monterey, CA, 23-25 September 2019, pp. 153-163.
- Knippenberg, S.C.M., Etman, L.F.P., Rooda, J.E., Wilschut, T., Vogel, J.A., 2020. Towards a module-based product platform for ship locks using DSM methods. *Proceedings of the 22nd International DSM Conference (DSM 2020)*, MIT, Cambridge, MA, 13-15 October 2020, pp. 53-62.
- Meyer, M.H., Lehnerd, A.P., 2011. *The power of product platforms: building value and cost leadership*. The Free Press, New York, NY.
- Pahl, G., Beitz, W., Feldhusen, J., Grote, K.H., 2007. *Engineering design: A systematic approach*. Springer-Verlag London, London, UK.
- Simpson, T.W. (2004). Product platform design and customization: Status and promise. *AIEDAM*, 18(1), 3–20.
- Ulrich, K. (1995). The role of product architecture in the manufacturing firm. *Research Policy*, 24(3), 419–440.
- Wilschut, T., Etman, L.F.P., Rooda, J.E., Vogel, J.A., 2019. Similarity, modularity, and commonality analysis of navigation locks in the Netherlands. *Journal of Infrastructure Systems*, 25(1): 04018043.

Contact: S.C.M. Knippenberg, Eindhoven University of Technology, Department of Mechanical Engineering, PO Box 513, 5600 MB Eindhoven, The Netherlands, s.c.m.knippenberg@tue.nl