INTERNATIONAL DESIGN CONFERENCE – DESIGN 2022
https://doi.org/10.1017/pds.2022.72

The Value of Information in Clustering Dense Matrices: When and How to Make Use of Information

F. Endress [1,2,✉], T. Kipouros [1], T. Buker [3], S. Wartzack [3] and P. J. Clarkson [1]

[1] Department of Engineering, University of Cambridge, United Kingdom,
[2] TUM School of Engineering and Design, Technical University of Munich, Germany,
[3] Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
✉ felix.endress@tum.de

Abstract

Characterising a socio-technical system by its underlying structure is often achieved by cluster analyses and bears potentials for engineering design management. Yet, highly connected systems lack clarity when systematically searching for structures. At two stages in a clustering procedure (pre-processing and post-processing) modelled and external information were used to reduce ambiguity and uncertainty of clustering results. A holistic decision making on 1) which information, 2) when, and 3) how to use is discussed and considered inevitable to reliably cluster highly connected systems.

Keywords: cluster analysis, design structure matrix (DSM), socio-technical systems, dense matrices, design models

1. Introduction

The amount of publicly available information has never been greater than today. The steady growth of complexity, expressed through huge amounts of data connecting diverse elements, raises the need to develop methods and techniques, which help us to access and interpret information (Eppinger and Browning, 2012). Within this context, the Design Structure Matrix (DSM) is an analytical model often applied to better understand and improve complex systems (Wynn and Clarkson, 2018). Detecting and translating structures of an engineering system into modules, bears the potential to better manage complexity, facilitate parallel work and accommodate future uncertainty (Baldwin and Clark, 2006).

A distinction is made between binary and numerical (analogue) DSMs, the latter being characterised by their increased information content resulting from their ability to model graded or stepped dependency weights rather than just binary values (Maier et al., 2017). At a certain level of granularity, the higher the connectivity of a network is, the denser its respective DSM model becomes, due to the many dependencies between the individual elements. To extract information about a system's structure from DSMs, algorithmic cluster analyses are generally conducted. Unfortunately, when searching for structures in highly connected DSMs, such analyses require special attention (Pimmler and Eppinger 1994). *This paper seeks to overcome this challenge by generally improving the clustering results in the context of dense matrices*. By contrasting research in the fields of computer science and mathematics into networks and graphs, this paper provides meaningful information that will be of use in engineering practice. It focuses on DSM applications in product development, where clustering solutions cannot be determined and assessed purely by mathematical means. To this end, Section 2 defines the cluster types relevant to socio-technical systems and describes the background of clustering DSMs. Section 3 presents two use cases that exemplify the problems of clustering dense

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

matrices. Several approaches for potentially improving the clustering of dense DSMs are introduced in Section 4. This is followed by a discussion of the results in Section 5 and a conclusion in Section 6.
As there is no quantitative definition of a dense matrix (cf. Newman 2010), the non-zero fraction (NZF) will be used to quantify the density of a DSM in this paper. Thus, matrices with a NZF of 50% and above will be referred to as 'dense' and the corresponding network as 'highly connected'. The NZF is calculated as the fraction of the actual number of dependencies to the maximum number of dependencies n * (n-1) in a DSM, where n is the number of elements in the modelled system.

2. State of the Art

Recent models of social, technical and socio-technical systems in engineering design are characterised by a triadic relation linking the target system with the model's structure and its purpose (Maier et al., 2017). Thereby, various requirements for models exist and are defined by the stakeholders' needs (Little, 1970). Applied to various engineering problems (cf. Eppinger and Browning, 2012), their potential for analytical and structural investigation makes DSMs a fitting tool for characterising a system. The concepts of granularity (Maier et al., 2017), method-based analyses using metrics (Kreimeyer, 2009), and clustering approaches form the basis for analysing the information modelled as a DSM.

2.1. Clustering

Known as 'community detection' in graph theory (Newman, 2010), the purpose of cluster analysis is to find underlying structures in a system. There are four **types of clusters**, beginning with 1) a **module**, defined as "a relatively independent chunk of a system that is loosely coupled to the rest of the system" (Hölttä-Otto et al., 2012). Modules comprise more than one element of the modelled system. Besides modules, there is usually at least one more cluster within each technical or socio-technical system, comprising a single element (Hölttä-Otto et al., 2012). These may be 2) a **receiving bus**, 3) a **sending bus**, or 4) a **framework**. Based on the selected DSM convention of 'columns influence rows', receiving buses lead to many entries in a certain row in the DSM. The bus connects more than one module and primarily receives information from the elements it connects. In contrast, a sending bus is defined as an element influencing several other elements in different modules. A framework is defined as a sending and receiving bus at the same time and is thus bi-directionally connected to other parts of a system. A set of clusters is called a clustering or a clustering solution. Clusters are usually visualized as boxes drawn into a DSM with correspondingly ordered elements, as shown by the example of the upper-left DSM in Figure 1.
Revealing a hidden or underlying structure in a system is not always easy, as increasing complexity and scale (e.g. the number of elements) complicate the system's characterisation (Sarkar et al., 2013, Sharman and Yassine, 2004). Thus, the robustness of a found solution (Sharman and Yassine, 2004) and the sensitivity of an identified clustering must be considered for characterisation (Chiriac et al., 2011). There are several algorithms that can be used for clustering DSMs, each pursuing a different direction; these include spectral methods (e.g. Sarkar et al., 2014) and modularity metrics (e.g. Jung and Simpson, 2014). While spectral methods do not require a priori assumptions regarding a system's structure (Sarkar et al., 2013), they are usually limited to detecting clusters comprising a single element, i.e. frameworks or buses. Another clustering algorithm, using a 'minimum description length (MDL)' as an objective function, was introduced by Yu et al. (2007). It can detect modules and frameworks and uses a 'genetic algorithm (GA)' as a search strategy. Within the conducted use case in Section 3 this clustering approach was further extended with a receiving and sending bus detection and serves as an algorithm precisely detecting clusters for benchmarks in the following.

2.2. Pre- and Post-Processing

A basic clustering approach that pursues the creation of a DSM model comprises three stages: pre-processing, the clustering process itself, and post-processing. In the "pre-processing [step,] the DSM model is prepared for the subsequent clustering" (Helmer et al, 2010). The post-processing performed later in the clustering procedure describes "the correction and improvement of the results obtained by

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

application of the clustering algorithm" (Helmer et al, 2010). A review of literature relating to DSM and clustering applications reveals that the processing of modelled data is usually adjusted to the respective application. In this context information is often extracted from the model and analysed using general metrics. As an example, Kreimeyer (2009) discussed a comprehensive set of 52 structural metrics that can be used generically to investigate complex systems and processes. Application-specific pre- and post-processing measures, such as the use of a modified rating scale (-2 to +2), symmetry, or perspective reduction (combining various modelled interaction types), can be found in Helmer et al. (2010).

3. The challenge of extracting information from dense matrices

To identify limitations in the clustering of dense matrices, a model of a **hand drilling machine (HDM)** was created and clustered by applying three different algorithms. Results are shown in the first row of Figure 1. Numerical dependency weights represent interaction strengths between engineers that are responsible for one particular part of a power tool. Algorithmic implementations applied to cluster the system follow a coordination cost approach (Thebeau, 2001) and a modularity strength metric, both of which are implemented in the Cambridge Advanced Modeller (CAM). Additionally, the extended MDL (GA) based clustering was applied (see Section 2.1).

An **artificial binary test system (ATS)** was also modelled. This is illustrated in the bottom row of Figure 1. Various reasonable clustering options are highlighted in ATS I to III. All three clusterings include all modelled connections within the clusters and thus represent theoretically ideal clustering solutions. A preference for a particular clustering cannot be expressed without knowing the context.

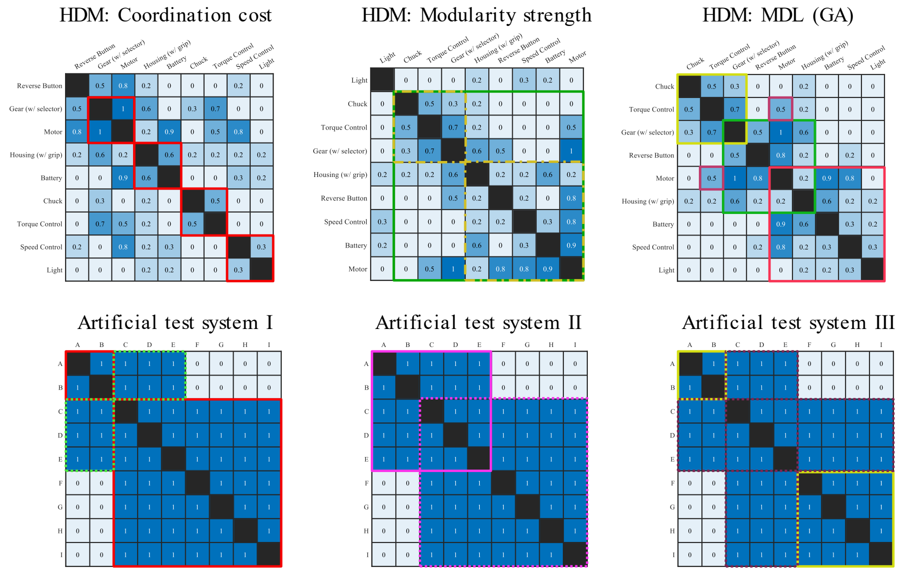


Figure 1. Use case of a hand drilling machine model (NZF = 58.3 %) that was clustered using three different algorithms. An artificial binary test system (NZF = 77.8 %) shows potentially ideal clusters, neglecting the context. Different clustering solutions are indicated by the coloured boxes.

Two key problems occur when clustering the highly connected systems of an HDM or ATS. First, there are no guidelines for selecting an algorithm (including an abstract measure of modularity) to cluster highly connected systems. As clustering algorithms differ greatly, detected clusterings can be very different, too (cf. Figure 1). This shows that selecting the right clustering algorithm for characterising dense matrices requires knowledge of the system's structure. However, such knowledge is often not available prior to clustering. This limitation is not unique to highly connected systems, but it is reinforced by the density of the DSMs. The challenge of selecting a modularity metric is down to the major differences that exist between the individual measures (Hölttä-Otto et al., 2012). The chosen objective function has a major impact on the results of the analysis. Overlapping clusters represent a

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

further level of complexity, aggravating the clustering. However, to make the best use of modularity in mechanical systems, overlaps need to be considered and selected early in the design process (Helmer et al., 2010). A system's modularity also depends on the granularity chosen for a model (Chiriac, 2011). The concept of granularity thereby connects the modelling process (including the information collected and contained in the model) with cluster analyses and their results. Pre-processing the data can also change the granularity, and it can impact results when searching for structures. As a state of the art approach for clustering DSMs, the generation and comparison of different clustering solutions is generally recommended before a result is accepted (Eppinger and Browning, 2012).

The second problem is that even with the same algorithm, the results require interpretation; they do not represent robust solutions free of uncertainty (here referred to as ambiguous clustering solutions). Even if clustering solutions are detected in each clustering run (conducted to produce Figure 1), it is necessary to decide whether the resulting solutions are reasonable. If a system is highly connected, available methods often fail to deliver reliable clustering results, as multiple (pareto optimal) structural interpretations of a model potentially exist, even if the same algorithm is used. The most appropriate solution is often dependent on the context and follows interpretations and modifications. Whereas some algorithms do not scan the entire solution space (Behncke et al., 2015), others miss configurations due to a path dependency (Sharman and Yassine, 2004). Manual clustering therefore generally follows the application of an algorithm to reveal comparable or possibly even better clusterings (Behncke et al., 2015).

4. An approach for clustering dense matrices

In the big picture of clustering highly connected systems, the overall aim within Section 4 is to test different measures in the clustering procedure and their effects on clustering results, to make best use of the information available in models and beyond (e.g. context, modelling approach, etc.).

Therefore, in the following three sources of density are introduced. They can be used to broadly categorize highly connected systems, which provides a baseline to classify dense matrices representing technical or socio-technical systems. The use of information at various steps in the clustering procedure is then investigated. For pre-processing, various thresholds are applied and their respective effects on the detected modular structure compared, based on the limitations discussed above. This is followed by the investigation of manual interpretations of clustering results for post-processing. Therefore, an optimisation was performed to enhance visual clarity of the DSM test case of the HDM. Finally, three simple guidelines are given for interpreting clusterings a posteriori.

4.1. Sources of density

Figure 2 presents the categories for the sources of density. In the first category, 'unequal cluster sizes', interpretation and modularisation of the system is a simple matter, due to the clear borders between the individual chunks. The second category, 'modular overlap', refers to a group of systems defined by the existence of overlapping clusters, i.e. elements that are assigned to more than one cluster (see Section 2.1). A clustering algorithm will still easily be able to detect the modules, frameworks, and buses, but it will be increasingly dependent on its parameters being set a priori. There might also be multiple reasonable clustering solutions, which means that more than one detected structure is likely to exist with the best (or close to best) value for a chosen objective function. The third category, 'random links', stems from a mix of modules and single dependencies between elements. Detecting clusters by simply looking at the DSM visualisation generally does not produce sufficient solutions, but automated cluster detection algorithms might also result in structures with 'imperfect clusterings', i.e., blank spaces (non-existent dependencies) in a DSM assigned to a cluster, or very small and unrealistic clusters. This type of system is usually the most challenging for cluster detection.

If a system is highly connected, one of these categories will usually apply. A review of the system's element and interaction types, as well as its environment may reveal the prevalence of one of the categories introduced. Knowing the source of density facilitates understanding the structure of a system. It enables a more appropriate choice of objective function (or clustering algorithm and its parameters), as well as reasonable pre-processing.

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

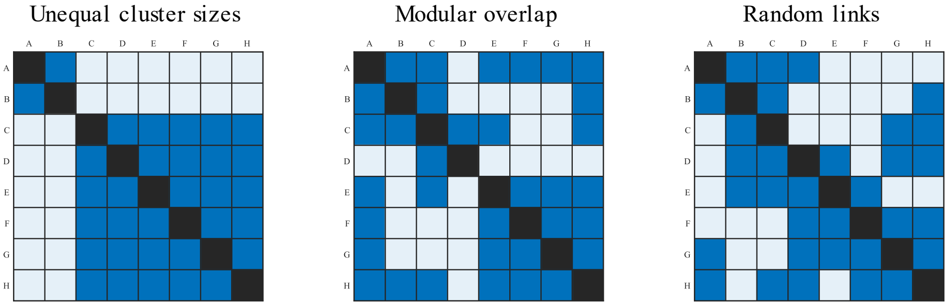


Figure 2. Source categories for highly connected systems represented as binary DSMs.

4.2. Pre-processing via thresholds

Pre-processing is the first step of a clustering procedure in which an interpretation of the modelled information is feasible. Thereby we examined whether a suitable pre-processing via threshold allows a better detection of the underlying structure, as a reduction of existing dependencies is expected to result in sparser matrices and thus reduced ambiguity in clustering solutions. A threshold follows the assumption that low dependency weights can be 'ignored' when searching for an underlying structure, as weak connections in a network are eliminated. In the following, two different types of thresholds are applied as a filter, with common values chosen for the threshold 't'. In the case of a numerical threshold, interactions between elements are eliminated below the threshold value and retain their initial value above it. Applying a binary threshold results in a binary DSM, as all dependency weights above and equal to the threshold value are set to one, whereas all other dependencies are removed, i.e. set to zero.

To investigate the effects of a threshold, an experiment was carried out based on the HDM system in Section 3. Since it is not possible to prescribe an ideal or correct clustering for a real system (as already shown, there are only opportunities or proposals), the clustering is evaluated as a comparison between two consecutive solutions, in which a distance measure expresses the difference between the two clusterings. With each threshold setting, the MDL-based clustering algorithm was applied 51 times. If the structure of a system is clearly defined, the same clustering solution can be expected, resulting in a mean for the measured distance of zero. If there is ambiguity in the clustering solutions for an applied threshold, the distances between consecutive clusterings found for the same system are expected to increase, leading to the higher distance averages shown in Table 1. The similarity of the solutions can thus be expressed and compared for the different thresholds. Measures for similar clusterings are performed in line with Goldberg et al. (2010), who examined dynamics in a social network by comparing clusterings found at different time steps. The best match for each cluster in one clustering C1 (i.e., a set of clusters) is searched for in the other clustering C2 and vice versa. The distance measure used is based on information theory, as introduced by Lancichinetti et al. (2009). For more details on cluster comparison, the reader is referred to (Goldberg et al., 2010) and (Lancichinetti et al., 2009).

The results of the study are presented in Table 1. There are generally two reasons for any variance in clustering results. It can stem either from the probabilistic nature of the genetic algorithm or from the ambiguity added in the clustering procedure. Interpretation of the distance measures shown in Table 1 is, however, feasible, as identical results were found in multiple runs.

4.2.1. Binary thresholds

The binarisation of dependency weights led to variance in the clustering results. In such cases, multiple runs can reveal different clustering opportunities for the same system. Of course, the results depend on the system modelled (here, a HDM). Compared to the clustering of the original system, corresponding to a numerical threshold of value 0.00, the ambiguity in the clustering solutions increased when a binary threshold was applied, even if the number of dependencies in the DSM and, in turn, the density of the matrix, was reduced. How great the difference between the solutions found was, can be seen in the mean of the distance measures in Table 1.

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

Table 1. Effects of different (pre-processing) threshold types and values on the variance of clustering solutions found for the HDM.

Thres-hold value t	Binary threshold		Numerical threshold	
	Frequency of different clusterings detected in subsequent analyses	Mean of distance measures greater than zero	Frequency of different clusterings detected in subsequent analyses	Mean of distance measures greater than zero
0.00	38/50	0.34	0/50*	-
0.33	30/50	0.36	0/50	-
0.66	37/50	0.38	0/50	-

* corresponds to a clustering of the initial system

Only a slight increase was noted for greater threshold values, and it is use-case-specific (e.g. there are several dependencies which are smaller or equal to 0.3). Generally, binarisation always led to multiple optimal clustering solutions and thus increased the level of ambiguity in the clustering.

In practice, whether or not a binary threshold is reasonable greatly depends on the type of interactions modelled. Binarisation leads to a significant loss of information, when the context of modelled data is not known. However, it might be used to explore further clustering opportunities or to compensate for modelling inconsistencies.

4.2.2. Numerical threshold

It can also be seen from Table 1 that when a numerical DSM was processed, the clustering algorithm always detected the same solution for the corresponding system (0/50). However, it should be noted that different threshold values resulted in different clusterings (not shown in Table 1). The threshold applied changes the structure of the system; in a real-world application, practitioners would need to decide (prior to knowing the structure of a system) which links in the network are irrelevant, i.e. where to set the threshold value. Even if the detected clusterings seem correct (the same structure is detected in all fifty comparisons), a different threshold might lead to a very different, yet optimal, solution. Unfortunately, in most cases, a reliable decision regarding how to use the information in a system for eliminating weak connections is not possible, as any changes will only have an effect later on in the procedure (during clustering) and lead to different structures. In the case of the HDM, it might be reasonable to conduct a manual evaluation of the resulting clusterings, although manual clustering is not always easy to perform in socio-technical or larger systems. It is of fundamental importance to decide whether low-dependency weights (or interaction strengths) can be interpreted as irrelevant prior to determining a reasonable structure of the system.

4.2.3. Summary of the threshold study

To sum up, we found that a pre-processing step using thresholds can decrease the reliability of a clustering in two ways. The first impact resulted from the reduced information content (from binarisation) of the model, leading to various optimal clusterings (exacerbating the first problem addressed in Section 3). This was only observed in the present experiment with the binary threshold, but can theoretically occur for numerical thresholds, too. The second loss of reliability stems from the early interpretation of the information modelled, prior to a structure definition. Even though the threshold creates a system with a unique clustering solution, this solution may be biased by a user who defines a threshold value prior to clustering. Setting this value without knowing the structure (the reason why the clustering is actually being performed) is challenging and strongly influences the structure detected in the next step of the clustering procedure. Both types of thresholds are therefore considered non-beneficial with respect to the two problems defined in Section 3 when information on the context is missing.

4.3. Post-Processing: Improving the manual cluster exploration

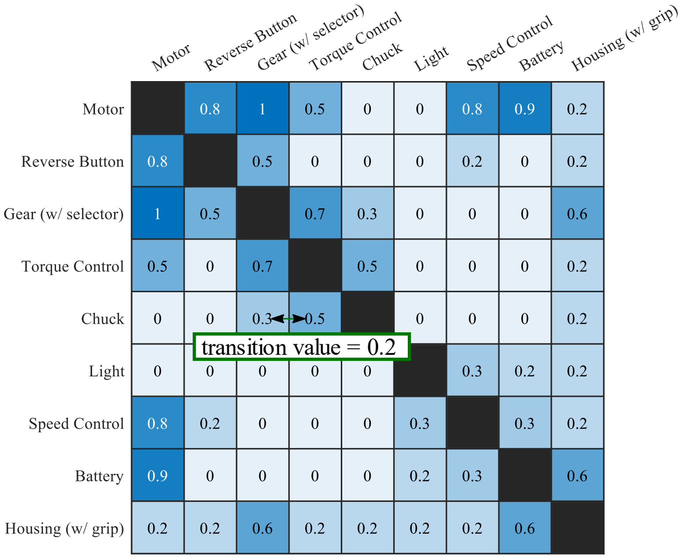
Decision makers, like managers or product developers, generally take into consideration further implications and constraints relating to a system (information not explicitly contained in the model),

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

leading to clustering solutions that are comparable with or even better than those detected by algorithms (Behncke et al., 2015). In addition, experience of a method combined with prior knowledge of the system enables enhanced extraction of information from a network modelled as a DSM (Keller et al., 2006). Schöttl and Lindemann (2015) combined a system's engineering perspective with fundamentals of psychology, defining complexity as "a perceived parameter in socio-technical systems" (Schöttl and Lindemann, 2015). Especially for managing processes a visualisation of the model is one of the key factors determining the success of an implementation (Heisig et al., 2009). For clustering DSMs, often a manual processing of the clusters found is added at the end of a cluster analysis (Behncke et al., 2015). If automated clustering procedures still require a human being to process the DSM, the question can be raised, if skipping the clustering (usually based on abstract metrics) is reasonable. Thus, an investigation follows next, where the order of elements is not determined by an artificial abstract value (e.g. defined as modularity), but rather fulfils the purpose of clearly visualising chunks in the DSM. Especially for systems specified as 'Random links' as the origin of density, a manual clustering is expected to improve clustering results, if elements are ordered to enhance clarity in the visualization (Section 4.3.1). Additionally, literature reviews reveal a lack of guidelines for clustering interpretations. A systematic guidance, using in- and external information of a model, is therefore introduced in Section 4.3.2, to improve clusterings in the post-processing.

4.3.1. Enhancing clarity: Transition reduction algorithm (TRA)

To facilitate manual clustering, we assumed that if borders between modules are clearly observable and blank spots in a DSM are brought together, the system would have a very 'clear' (or better, 'clearly observable') structure. Hence, an algorithm was developed to minimize the 'transitions' within a DSM by reordering elements. A transition value (TV) is defined as the absolute difference between two neighbouring dependency weights in the matrix, vertically and horizontally. A factorial design was chosen, in which all possible orders (n! = 9! = 362,880) for a system with nine elements were formed and the TV calculated. After 13.4 seconds, the transition value was calculated for all orders. The resulting DSMs, with elements ordered for minimum TVs, are then shown to a user for manual clustering explorations. Applying the TRA to the test system of the HDM, four orders of elements with minimum sums of TVs were found. One such solution is presented in Figure 3.



	Motor	Reverse Button	Gear (w/ selector)	Torque Control	Chuck	Light	Speed Control	Battery	Housing (w/ grip)
Motor		0.8	1	0.5	0	0	0.8	0.9	0.2
Reverse Button	0.8		0.5	0	0	0	0.2	0	0.2
Gear (w/ selector)	1	0.5		0.7	0.3	0	0	0	0.6
Torque Control	0.5	0	0.7		0.5	0	0	0	0.2
Chuck	0	0	0.3	0.5		0	0	0	0.2
Light	0	0	0	0	0		0.3	0.2	0.2
Speed Control	0.8	0.2	0	0	0	0.3		0.3	0.2
Battery	0.9	0	0	0	0	0.2	0.3		0.6
Housing (w/ grip)	0.2	0.2	0.6	0.2	0.2	0.2	0.2	0.6	

transition value = 0.2

Figure 3. An ordered DSM, in which the minimized objective function is the sum of all absolute TVs. A single TV is highlighted with an arrow as an example.

As shown in Figure 3, the TRA by ordering elements visualises the DSM in a way that makes it easier for humans to search the system for clusters. To evaluate the results of this visual inspection, components are divided into mechanical and electrical elements, in which the purely visualisation-based algorithm enables reliable detection of the domain-specific modules (e.g. light, speed control

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

and battery as electric modules). Since elements located on the outside do not have any neighbours, the algorithm also presents suggestions for frameworks (see motor and/or housing as a framework). A user is able to observe different clustering opportunities at first sight, without the need for any abstract measure (such as a modularity metric) being defined mathematically prior to the analysis. For example, the upper left-hand corner could be clustered as two overlapping, but smaller, modules, or as one with zero-dependencies within the module. This facilitates the interpretation of interfaces, overlaps and opportunities for structuring the system. However, already small threshold values would eliminate important information in the electric domain, consisting of dependency weights between 0.2 and 0.6. Some of these dependencies were removed in Section 4.2, when a threshold value of 0.3 was applied to the model prior to the algorithmic clustering. Another optimisation strategy (e.g. a GA) is recommended if applying the TRA to larger systems. However, if clusters exist, it will not be necessary to search the entire solution space. For example, in the 'unequal cluster sizes' matrix (see Figure 2), the order 'A-B-C-D-…' would have the same sum of TVs as 'B-A-C-D-…' and allow the same interpretation. It is therefore the case that the TRA performs particularly well for dense systems.

4.3.2. Clustering interpretations

As shown previously, if there are multiple options for optimally clustering a system, human post-processing is inevitable. There are three principles according to which modelled and non-modelled information can be used later in a clustering procedure for interpreting clustering solutions from algorithmic analyses. Further guidelines will form the subject of subsequent research.

- **Awareness of limitations**

Since a model is by its nature an abstraction of a target system, the inevitable differences between the real world and the modelled system lead to limitations in clustering. Knowing these differences can be used to exclude potential clustering solutions that do not appropriately depict reality. For example, the clustering of a binary system might be limited because an algorithm treats each dependency the same way while differences in the real target system might exist. A pairwise comparison of two opportunities for clustering a system might help to prioritize clustering solutions a posteriori and overcome modelling uncertainties.

- **Adding (domain) knowledge**

If clustering pursues a certain purpose, information about a system usually exist which is not explicitly contained in the model. Clustering might be improved by using this knowledge, which can, for example, relate to interactions modelled, element types, domains, or the environment in which the system is embedded. Also, a realistic view of indirect dependencies can support the rating of clustering solutions (if 'A' follows 'B', and 'B' follows 'C', does 'A' also follow 'C'?). This domain knowledge can help to pre-process and interpret models in an appropriate manner.

- **Accepting suboptimal solutions**

Frequently, there are several answers to the question of what constitutes ideal clustering for a certain system. There are several algorithms that can be used to explore different possibilities. Since an optimal structure found using an objective function may be different to the optimal structure of the real system, 'optimal clusterings' must always be defined by an appropriate context. Sometimes it is beneficial to accept 'non-ideal solutions', i.e. clusters in a DSM with blank spots included. An example of these suboptimal clustering solutions can be seen in Figure 1, where the MDL and modularity strength algorithm provided solutions that contained clusters of elements that included zero-weighting interactions. The density within the clusters may be mathematically suboptimal, i.e. in terms of a different abstract modularity metric, yet potentially depicts reality well enough.

5. Discussion

Highly connected systems require special attention when searching for underlying structures by applying cluster analyses. The aim of this paper is to seek strategies to use of the information contained in dense matrices along with external knowledge of the modelled system and its environment. A distinction is proposed between information content and information value. Applying pre- and post-

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

processing techniques to a model enables available information to be used, thus enhancing its value and importance. On investigating several measures within a clustering procedure, post-processing and results interpretation were found to be the preferred steps at which to process and analyse models if no context is known and external knowledge is not availabe. Interpretations of modelled information in pre-processing (for example by applying a threshold) can be misleading, as the characteristics of the system (modularity, etc.) are still unknown at this stage, even if they are required so as to enable an appropriate choice of threshold values. Once clustering opportunities have been found, the user can make systematic use of modelled and non-modelled information in post-processing. The power of manual clustering interpretation was shown using the TRA, where the clear visualization helped to investigate the underlying structure of the model, without the need to detect clusters algorithmically.

The modelling context also plays a major role in defining when and how to apply measures in the course of the clustering procedure. All steps (pre-processing, algorithmic search for (an usually abstract) modularity, and post-processing) offer the potential to interpret and make use of information affecting the clustering results. However, reducing the information content in a model at an early stage of the clustering procedure can have a significant effect on the structure detected by an algorithm (e.g. deleting low dependency weights when applying thresholds). A more holistic view is required, along with information on both the modelling and the system environment, to decide whether an early measure, such as a threshold, is reasonable. A trial and error approach to investigating various pre- and post-processing measures and clustering techniques can still be considered a reasonable, if time-consuming, procedure. The assumption that decreasing density (NZF) will reduce the dependence of a clustering solution on the chosen algorithm could not be proven. Rather, the application of thresholds resulted in the detection of different structures. It was noted that binary thresholds amplified the second challenge of clustering dense matrices, that is the increased number of optimal clusterings.

To enable a structured interpretation of clustering solutions, the awareness of limitations, non-modelled knowledge, as well as thoughts on imperfect clusters were helpful when enhancing clustering solutions a posteriori. These approaches do not claim to be comprehensive but should be regarded as a basis for discussing clustering results in post-processing and for further research and case studies.

6. Conclusion

Two problems were identified in the context of clustering highly connected systems. First, the selection of an appropriate algorithm, including an abstract modularity metric (as an objective function), is a challenging task. Second, dense matrices bear an increased likelihood of finding pareto optimal clusterings. At different stages in the clustering procedure information were used to improve results. It was found, that a binarisation is rather suitable for structural explorations, whereas numerical thresholds can reduce ambiguity in detecting structures. However, for interpreting modelled information early in the clustering process, knowledge on the system, its context or the modelling is required. The higher the connectivity of a system is, the greater is the need for a manual interpretation of a resulting clustering. This post-processing can be enhanced by introduced guidelines and clear DSM visualizations. Generally, a holistic view and careful use of analytical tools within a clustering procedure is recommended. By this means clustering results are obtained faster, and the potentials of using DSMs are further exploited to better manage complexity in highly connected systems.

References

Baldwin, C. Y. and Clark, K. B. (2006), "Modularity in the design of complex engineering systems", In: Braha, D., Minai, A.A. and Bar-Yam, Y., *Complex Engineered Systems. Understanding Complex Systems*, Springer, Berlin, Heidelberg, Germany, pp. 175–205. https://doi.org/10.1007/3-540-32834-3_9

Behncke, F., Maurer, D., Schrenk, L., Schmidt, D., and Lindemann, U. (2015), "Clustering technique for DSMs", *Risk and change management in complex systems. Proceedings of the 16th International DSM Conference Paris, France, 2–4 July* 2014, Carl Hanser Verlag, Munich, Germany, pp. 177-186. https://doi.org/10.3139/9781569904923.018

Browning, T. R. (2015), "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

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press

Chiriac, N., Hölttä-Otto, K., Lysy, D., and Suh, E. S. (2011), "Level of modularity at different levels of system granularity", *Journal of Mechanical Design*, Vol. 133 No. 10, pp. 329–339. https://doi.org/10.1115/1.4005069

Eppinger, S. D. and Browning, T. R. (2012), *Design structure matrix methods and applications*, MIT press, Cambridge, MA, USA. https://doi.org/10.7551/mitpress/8896.003.0003

Goldberg, M. K., Hayvanovych, M., and Magdon-Ismail, M. (2010), "Measuring similarity between sets of overlapping clusters", *SocialCom '10: Proceedings of the 2010 IEEE Second International Conference on Social Computing*, IEEE Computer Society, Minneapolis, MN, USA, pp. 303–308. https://doi.org/10.1109/SocialCom.2010.50

Heisig, P., Clarkson, P. J., Hemphälä, J., Wadell, C., Bergendahl, M. N., et al. (2009), Challenges and future fields of research for modelling and management of engineering processes, *MMEP White Paper - Report from Workshops with Industry and Academia*, Design Society, Cambridge, Munich, Stockholm.

Helmer, R., Yassine, A., and Meier, C. (2010), "Systematic module and interface definition using component design structure matrix", *Journal of Engineering Design*, Vol. 21 Nr. 6, pp. 647–675. https://doi.org/10.1080/09544820802563226

Hölttä-Otto, K., Chiriac, N. A., Lysy, D., and Suk Suh, E. (2012), "Comparative analysis of coupling modularity metrics", *Journal of Engineering Design*, Vol. 23 No. 10-11, pp. 790–806. https://doi.org/10.1080/09544828.2012.701728

Jung, S. and Simpson, T. W. (2014), "A clustering method using new modularity indices and genetic algorithm with extended chromosomes", *DSM '14: Risk and change management in complex systems. Proceedings of the 16th International DSM Conference Paris, France, 2–4 July* 2014, Carl Hanser Verlag, Munich, Germany, pp. 167–176. https://doi.org/10.3139/9781569904923.017

Keller, R., Eckert, C. M., and Clarkson, P. J. (2006), "Matrices or node-link diagrams: Which visual representation is better for visualising connectivity models?", *Information Visualization*, Vol. 5 Nr. 1, pp. 62–76. https://doi.org/10.1057/palgrave.ivs.9500116

Kreimeyer, M. F. (2009), *A Structural Measurement System for Engineering Design Processes*, Technical University of Munich.

Lancichinetti, A., Fortunato, S., and Kertész, J. (2009), "Detecting the overlapping and hierarchical community structure in complex networks", *New Journal of Physics*, Vol. 11 Nr. 3, pp. 033015. https://doi.org/10.1088/1367-2630/11/3/033015

Little, J. D. C. (1970), "Models and managers: The concept of a decision calculus", *Management Science*, Vol. 16 Nr. 8, pp.466-485. https://doi.org/10.1287/mnsc.16.8.b466

Maier, J. F., Eckert, C. M. and Clarkson, P. J. (2017), "Model granularity in engineering design–concepts and framework", *Design Science*, Vol. 3 No. e1. https://doi.org/10.1017/dsj.2016.16

Newman, M. E. J. (2010), *Networks: An introduction*, second edition, Oxford University Press, Oxford, UK. https://doi.org/10.1093/oso/9780198805090.001.0001

Pimmler, T. U. and Eppinger, S. D. (1994), "Integration analysis of product decompositions", *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, Minneapolis, MN, USA, pp. 343-351. https://doi.org/10.1115/detc1994-0034

Sarkar, S., Dong, A., Henderson, J. A., and Robinson, P. A. (2014), "Spectral characterization of hierarchical modularity in product architectures", *Journal of mechanical design*, Vol. 136 Nr. 1, pp. 0110061–01100612. https://doi.org/10.1115/1.4025490

Sarkar, S., Henderson, J. A., and Robinson, P. A. (2013), "Spectral characterization of hierarchical network modularity and limits of modularity detection", *PLOS ONE*, Vol. 8 No.1, pp. e54383. https://doi.org/10.1371/journal.pone.0054383

Schöttl, F. and Lindemann, U. (2015), "Quantifying the complexity of sociotechnical systems – a generic, interdisciplinary approach", *Procedia Computer Science*, Vol. 44, pp. 1–10. https://doi.org/10.1016/j.procs.2015.03.019

Sharman, D. M. and Yassine, A. A. (2004), "Characterizing complex product architectures", *Systems Engineering*, Vol. 7 No. 1, pp. 35–60. https://doi.org/10.1002/sys.10056

Thebeau, R. E. (2001), *Knowledge management of system interfaces and interactions from product development processes*, Massachusetts Institute of Technology.

Wynn, D. C. and Clarkson, P. J. (2018), "Process models in design and development", *Research in Engineering Design*, Vol. 29 Nr. 2, pp. 161–202. https://doi.org/10.1007/s00163-017-0262-7

Yu, T.-L., Yassine, A. A., and Goldberg, D. E. (2007), "An information theoretic method for developing modular architectures using genetic algorithms", *Research in Engineering Design*, Vol. 18 Nr. 2, pp. 91-109. https://doi.org/10.1007/s00163-007-0030-1

https://doi.org/10.1017/pds.2022.72 Published online by Cambridge University Press