



Citation for published version:

Bonvoisin, J, Buchert, T & Stark, R 2018, Formal system for the expression of target-oriented design heuristics. in N Bojcetic, N Pavkovic, D Marjanovic, M Storga & S Skec (eds), 15th International Design Conference, DESIGN 2018. vol. 1, The Design Society, pp. 45 - 56, International conference on DESIGN 2018, Dubrovnik, Croatia, 21/05/18. <https://doi.org/10.21278/idc.2018.0120>

DOI:

[10.21278/idc.2018.0120](https://doi.org/10.21278/idc.2018.0120)

Publication date:

2018

Document Version

Peer reviewed version

[Link to publication](#)

Publisher Rights

Unspecified

University of Bath

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.

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



FORMAL SYSTEM FOR THE EXPRESSION OF TARGET-ORIENTED DESIGN HEURISTICS

J. Bonvoisin, T. Buchert and R. Stark

Abstract

In design, it is generally neither possible to list all solutions to a given problem nor to prove whether a solution is optimal. In this context, guidance can be provided to designers in the form of heuristics, i.e. solutions validated by practice which have proven to lead towards satisfactory solutions but which are not guaranteed to be optimal or perfect. This article contributes to the development of supportive design tools by providing a first formal system for the expression of target-oriented design heuristics and their collection in consistent sets.

Keywords: design guidelines, design principles, design heuristics, design for x (DfX), design knowledge

1. Introduction

The essence of design creativity is to develop solutions for open, complex and ill-defined problems (Gonçalves et al., 2016). For those problems, there is generally no systematic method designers can be provided with in order to derive optimal solutions. In other words, it is neither possible to list all solutions to a given problem nor to prove the absolute superiority of a given solution. In the absence of systematic methods, guidance can be given to designers in the form of heuristics, i.e. solutions which have proven to lead towards satisfactory solutions but which are not guaranteed to be optimal.

The relevance of design heuristics as a tool to face complex design tasks is particularly illustrated in the context of sustainable product development. As a result of the pressure to comply with the challenges of sustainability, product design teams face raising numbers of partly competing design requirements (as for example discussed in Buchert et al., 2017). This leads to more complex and risky decision-making, as trade-offs and unforeseen side effects of decisions are more likely to occur. This requires an increased capacity to explore the potential solution space allowed by a given set of product requirements. Great efforts have been made by the international research community in order to identify design strategies which are known from experience or supposed to lead to sustainable products. For example, extensive sets of design for Environment (DfE) heuristics have been provided by Wimmer and Züst (2001) and more recently by Go et al. (2015) as well as Telenko et al. (2016). Heuristics have also been compiled for other purposes such as product flexibility (Bischof and Blessing, 2008), ease of assembly (Barbosa and Carvalho, 2013) and usability (Gong and Tarasewich, 2004).

While many design heuristics have been published throughout scientific literature, less effort has been invested to evaluate their formal quality and usability. Published heuristics sets are generally weakly structured, difficult to navigate in, partly competing and overlapping, and offer different levels of detail. As a result, they do not offer sufficient clarity and usability to be integrated in daily engineering practice. The main claim of this article is that this insufficient usability is the result of incomplete theoretical

foundations and formal definition of design heuristics. Addressing this issue, this article introduces a formal system for the expression of design heuristics. It is aimed at supporting the future development of intuitive applications for industrial designers and product development engineers.

The next section summarizes relevant elements of theory provided by the literature. It introduces the concept of target-oriented design heuristics as a mean to signpost the focus of the article and to fill missing elements of theory. It also highlights the limitations in the structure of existing heuristics sets and derives requirements for the development of new structuration approaches. As an answer to this, Section 3 introduces a new formal system for the expression of target-oriented design heuristics. Section 4 illustrates this formal system through the reformulation of 27 Design for Recycling (DfR) taken from literature. Finally, Section 5 discusses the practical value of the introduced formal system with the help of three use cases.

2. Theory of design heuristics

In the field of psychology, a cognitive heuristic is a “rule of thumb” used to make a decision or judgment (Nisbett and Ross, 1980, cited in Kramer et al., 2014). In the specific domain of product development, design heuristics—alternatively called design principles or design guidelines—have been defined as “context-dependent directives, based on intuition, tacit knowledge, or experiential understanding, which provide design process direction to increase the chance of reaching a satisfactory but not necessarily optimal solution” (Fu et al., 2016). In other words, design heuristics are strategies which have shown to be advantageous in problem solving, which help to quickly identify possible solutions to a new problem, but which are not guaranteed to produce any successful one. In the absence of systematic methods and deterministic rules for solving design problems, design heuristics are a predominant form to capture, accumulate, and reuse design knowledge (Kruse and Seidel, 2017). In this sense, they implement the concept of case-based reasoning (CBR), defined as use of “specific knowledge of previously experienced, concrete problem situations” as a basis to solve a new problems (Aamodt and Plaza, 1994).

Heuristics are generally expressed as short instructive statements (Sarnes and Kloberdanz, 2015) formulated as an imperative phrase (Fu et al., 2016) which contains at least a verb and an object (Bonvoisin et al., 2010). Two examples are “make incompatible materials easily separable” or “change contact material” (Yilmaz et al., 2013, Telenko et al., 2016; respectively). Beyond this mere formal aspect, some authors suggested heuristics need to be illustrated by contextual information specifying the meaning and the applicability of design heuristics. Some documentation formats have been introduced in the literature recommending capturing information such as: level of supporting evidence, time stamp, additional textual description, good and bad examples, underlying generic intent, goal conflicts and applicability in the product development process (Bauer, 2003; Bonvoisin et al., 2010; Sarnes and Kloberdanz, 2015; Fu et al., 2016). The importance of visual illustrations has been underlined by some authors as well, as illustrations have been proved to support information processing and be preferred as inspiration source by designers (Lofthouse, 2006; Bischof and Blessing, 2008; Gonçalves et al., 2016).

2.1. Target-oriented heuristics

Although this has not been stated explicitly so far, design heuristics referred to in scientific literature may be delineated along two dimensions.

First, design heuristics may be either *implicit* or *explicit*. Previous research shows that industrial designers and mechanical engineers unconsciously make use of implicit design heuristics in the cognitive process of generating new designs (Yilmaz et al., 2015). Implicit design heuristics are the product of experience and are neither formalized nor shared. On the opposite, explicit heuristics are formalized—that is, written—information. They may be used by designers in order to support inspiration and to come up with new solutions for a design problem. The most famous of these explicit heuristics are certainly the 39 “principles of invention” used in TRIZ (Altshuller, 1984).

Second, design heuristics may be either what we chose to call here *undirected* or *target-oriented*. The 39 “principles of invention” and the heuristics published by Kramer et al. (2014) are of the first type. They help “jumping in a new subspace of possible solutions” (Yilmaz and Seifert, 2011) without judging

about the compliance of these solutions to any requirement. On the contrary, target-oriented design heuristics are intended to ensure that all solutions of this subspace also satisfy specific requirements. In this sense, target-oriented design heuristics are procedures to orient product design towards a strategic objective (Vezzoli and Sciamia, 2006). They play the role of signposts which point the way to the realisation of desired product properties (Bauer, 2003). Hence, while the usage value of undirected design heuristics is only to inspire new designs, those of target-oriented heuristics is also ensure convergence towards desired requirements. Target-oriented heuristics are typically used in Design for X (DfX) approaches and have been reported in domains such as flexible products design (Bischof and Blessing, 2008), design for environment (Wimmer and Züst, 2001; Go et al., 2015; Telenko et al., 2016) and design for assembly (Barbosa and Carvalho, 2013).

The rest of this article focuses exclusively on *explicit* and *target-oriented* design heuristics which will be simply referred to as *heuristics* in order to simplify readability.

Table 1. Five examples of undirected and target-oriented design heuristics

	<i>Undirected-heuristics from Kramer et al. (2014)</i>	<i>Target oriented (design for recycling) heuristics from Go et al. (2015)</i>
1	Add levels	Minimise the number of different types of materials
2	Adjust function through movement	Make subassemblies and inseparably connected parts from the same or a compatible material
3	Add natural features	Avoid the mixing of materials in assemblies
4	Add to existing product	Mark all plastic and similar parts for ease of identification
5	Add motion	Use materials which can be recycled

2.2. Structure of target-oriented heuristics sets

Heuristics are generally not used separately from each other but rather bundled in sets. Sets of target-oriented heuristics particularly provide pools of possible solution approaches corresponding to a given design objective—what Dahlström (1999) calls an “advice bank”. Those sets are mostly documented as unordered (as in Bischof and Blessing, 2008; Bonvoisin et al., 2010) or categorized lists (as in Go et al., 2015) and can be alternatively presented as cards (as in Kramer et al., 2014) or mind maps (as in Telenko et al., 2016, to some extent). In some fewer examples, heuristics are stored in databases offering more complex structures and filtering possibilities (Bauer, 2003; Russo et al., 2017).

The way heuristics are documented generally overlook that they are virtually connected with each other in a network of *concretisation* and *influence* relations. Concretisation relation occurs when a heuristic is translated into a more generic or specific one. An illustration of this relation in the context of object-oriented programming is the concept of class inheritance. This is the type of relation which is used by Dahlström (1999) for defining company-specific heuristics out of more generic ones. Influence relations occur when heuristics mutually reinforce or exclude each other, i.e. lead to a design trade-offs. For example, modularity may be defended as a design strategy to extend product lifetime, since it eases disassembly and hence product maintenance, upgrade and remanufacturing. At the same time, it implies the embedding in the product a larger amount of reversible mechanical connections, hence opening to more mechanical tolerances, instabilities and breakages. This strategy hence leads to a design trade-off inside the field of design strategies supposed to lead to longer product service life.

The definition of the concretisation and influence relations is summarized in the diagram in Figure 1 next page. As in object-oriented programming, concretisation relations generate tree structures, meaning a parent can have different children but a child can have only one parent. Since design parameters can be interrelated in many ways, influence relation builds network structures, meaning a parent can have more than one child and a child have more than one parent.

While the existence of these relations between design heuristics is implicitly acknowledged in the scientific community, there is so far no formal system allowing for recording and representing them.

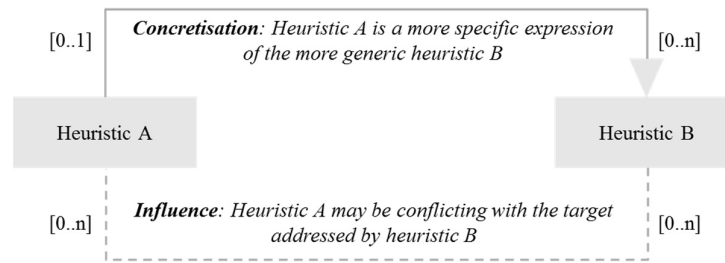


Figure 1. Two types of relations between target-oriented heuristics

These relations become important when it comes about generating consistent heuristic sets. Whatever the method used for gathering heuristics¹, the formalisation of heuristics sets implies processing routines ensuring homogeneity and non-redundancy (Bonvoisin et al., 2010). Homogeneity of a heuristic set is ensured by translating heuristics into a desired level of specificity. That is, to a same level in the tree structure formed by the concretisation relations. Redundancy is avoided by merging overlapping heuristics which may be generated by the translation into a desired level of specificity. However, no specific methodological approach systematically describing these routines has been developed so far. They are therefore processed ad hoc and subjectively, leading to biases and low reproducibility.

Little attention has been generally paid to date to the formalization of the criteria defining the quality of a heuristic set, such as homogeneity, non-redundancy or exhaustiveness. Defining such criteria would require developing a comprehensive and consistent formal system for expressing heuristics and sets. Some suggestions have been delivered by the scientific community in order to extended already existing formalization schemes. Fu et al. (2016) suggested developing a formalization scheme based on formal logic in order to describe heuristics and their articulations, however without providing any solution for it. Bauer (2007) proposed an analytical expression of heuristics as well as a specification of the genericity of heuristics using Weber's characteristics-properties-model (Weber, 2007). Partial aspects have been covered in different pieces of literature but further work is required in order to deliver a comprehensive formal system able to support the recording of interlinked heuristic in consistent sets.

3. Proposed formal system

This section introduces the main original propositions of this article, that is, 1) a formal definition of well-formed target-oriented heuristics and 2) two ways of representing heuristics sets through graphs (the second building on the first, being more powerful but requiring a supplementary hypothesis).

3.1. Syntactic definition of design heuristics

Bauer (2007) proposes that heuristics are made of four (implicit or explicit) elements: an instruction, a feature addressed by this instruction, the higher-level property which is to be influenced and the requirement set to this property. In the example "avoid sharp edges", the instruction is "avoid", the feature is "sharp edges", the property is the security of use (here implicit) and the requirement is to improve it (implicit as well).

Bringing further the abstraction process started by this author, we can identify three semantic units and a semantic connector at work in design heuristics, as illustrated in Figure 2 next page. The semantic units are 1) the artefacts the designer shapes, e.g. "edge", 2) the properties of these artefacts, e.g. "sharpness" and "safety" and 3) the changes in these properties, e.g. "improve." The semantic connector is an assumed causal relation between both sets of artefacts, changes and properties, e.g. which can be expressed by "in order to."

¹ Different methods have been reported to be used for the discovery of design heuristics and their collection in sets. This includes for example observation of a designer's "think-aloud" activity (Yilmaz et al., 2015), collection through long practice (Russo et al., 2017), analysis of existing products (Bonvoisin et al., 2010), extrapolation of the results of analytic methods (Telenko et al., 2009), literature review (Telenko et al., 2016) and even machine learning based on coded design data (Matthews et al., 2002).

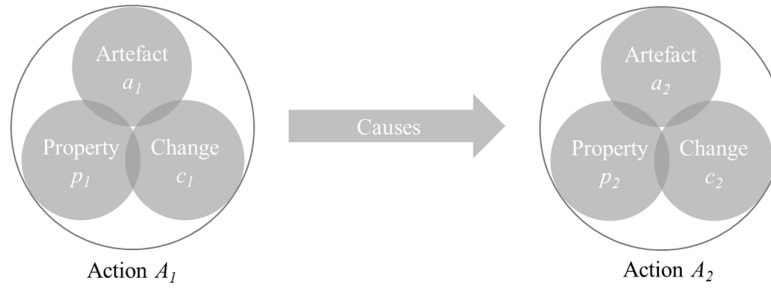


Figure 2. Semantic units and connectors at work in target-oriented design heuristics

Expressed in EBNF (extended Backus-Naur Form, ISO, 2004)², the syntactic rules for generating a well-formed target-oriented design heuristic are:

- *Heuristic = Action "causes" Action*
- *Action = Change Property Artefact*

In the following, we adopt also the short form $A1 \rightarrow A2$, where $A1$ and $A2$ are actions and the sign " \rightarrow " indicates a causal relation. We also refer to $A1$ as the "instruction" and $A2$ as the "targeted consequence". A well-formed heuristic is a heuristic which complies with the syntactic rules expressed here.

3.2. Heuristics sets as directed graphs

The syntactic definition of design heuristics defined above allows the formulation of design heuristics sets as directed graphs where nodes represent actions and directed edges causal relations.

The consistency of a heuristic sets is ensured when each heuristic of this set supports a given desired design objective. This is equivalent to say that each heuristic of this set shares the same targeted consequence. This means in turn that a consistent set of well-formed heuristics can be depicted by a star-shaped graph where the centre node is the shared targeted consequence and the eccentric nodes are the instructions of all different heuristics in the set. This is illustrated in Figure 3a, where all heuristics share the same targeted consequence A .

Following the same syntactic formalism, a more generic heuristic is concretised by a more specific one when the instruction of the former is the targeted consequence of the latter. In other words a heuristic $x1 \rightarrow y1$ can be concretised by any other heuristic $x2 \rightarrow y2$ where $x1 = y2$. Assuming transitivity of causal relations, a series of two realizing heuristics $x \rightarrow y$ and $y \rightarrow z$ automatically imply the existence of a third heuristic $x \rightarrow z$. Adding more specific heuristics to a star-structured heuristic set results in adding levels in the graph and hence forming a tree structure. This is illustrated in Figure 3b by heuristics $G \rightarrow F$ and $H \rightarrow D$ which are concretisations of the heuristics $F \rightarrow A$ and $D \rightarrow A$ respectively.

Following this same formalism again, the heuristic $x \rightarrow y$ has side effects when it exists an action z so $x \rightarrow z$. Adding such side effects to the graph requires nodes having more than one outgoing edge, hence forming a directed graph. This is exemplified in Figure 3c by heuristics $H \rightarrow D$ and $H \rightarrow E$ as well as $B \rightarrow A$ and $B \rightarrow F$.

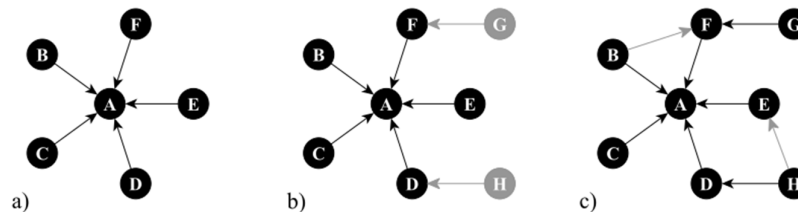


Figure 3. Star, hierarchic and directed graphs of design heuristics
(Legend: Each node represents an action, i.e. a change in the property of an artefact)

² The Extended Backus-Naur form is a meta-syntax, that is, a language to define allowable syntactic structures of simple languages such as programming languages.

3.3. Heuristics sets as directed and weighed graphs

The formal system defined so far still does not allow for recording design trade-offs. This would require the ability to detect when two actions are mutually exclusive, that is, that a change on an artefact property is incompatible with another change on that same artefact property. In order to do so, we suggest formulating the following axiom: any change of an artefact property is either an increase or a decrease in the value of this property. Using the same example as earlier, in order to fit with this axiom, "avoid sharp edges" needs to be reformulated into "decrease the sharpness of the edges in order to increase product safety." Using this axiom implies assuming that each action can be reformulated into another action where the change is either "increase" or "decrease". Taking up the new axiom, the syntactic rules for generating a well-formed target-oriented design heuristic expressed in EBNF are:

- *Heuristic = Action "causes" Action*
- *Action = Change Property Artefact*
- *Change = Increase | Decrease*

This supplementary axiom also allows simplifying this grammar into one single rule. Indeed, the instruction and the targeted consequence can now only be combined in four ways: increase/increase, increase/decrease, decrease/increase and decrease/decrease. Consequently, there are only two types of causal relations (see Table 2): either the properties change in the same direction (increase/increase and decrease/decrease) or in the opposite direction (decrease/increase and increase/decrease). Knowing this, it is possible to express heuristics according to the simple following rule expressed in EBNF:

- *Heuristic = Property Artefact ("positively" | "negatively") "influences" Property Artefact*

Table 2. Combinations of changes in instructions and targeted consequences

	Change in instruction	Change in targeted consequence	Combinations
Case 1	↗	↗	Positive influence
Case 2	↗	↘	Negative influence
Case 3	↘	↗	Negative influence
Case 4	↘	↘	Positive influence

This new format allows representing heuristics as graphs where each node represents an artefact property and each edge either represents a positive or a negative influence between these properties. Figure 4 reproduces the example given in Figure 3c in this new format. It not only allows depicting concretisation relations between heuristics but also trade-offs as in the case of a change in property *B*. An increase in *B* results in a decrease in *A*. But it also results in a decrease in *F*, which leads to an increase in *A*. As a result, *A* is both decreased and increased by *B*.

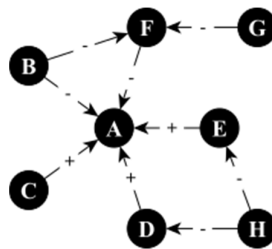


Figure 4. Heuristics sets as directed and weighed graphs (Legend: Each node represents a property of an artefact)

4. Validation of the formal system on an existing heuristic set

This section aims to demonstrate the applicability of the formal system introduced in Section 3. To do so, a set of 27 Design for Recycling (DfR) heuristics published by Go et al. (2015, Table 3, page 21, reproduced in Table 1 and fully in Appendix 1) has been translated into the new formal system.

This translation process highlighted four weaknesses in the original set of DfR heuristics listed hereafter:

1. Interestingly, only one heuristic among the 27 made clear reference to a targeted consequence: "Mark all plastic and similar parts *for ease of identification*." This heuristic clearly indicates an instruction (to mark plastics) and a pursued consequence of this instruction (to make plastics easily identifiable). The 26 other heuristics simply stated an instruction, letting the targeted consequence implicit. An example is "Avoid composite materials" which only bears an instruction and does not state what this instruction is supposed to produce.
2. In those cases the targeted consequence had to be guessed by the authors. As these heuristics have been collected in a set entitled "design for recyclability", the targeted consequence of all these heuristics could have been set to "recyclability of the product." This however does not bear further useful information. Instead, the authors strived to identify the generic principle targeted by the heuristic and contributing to recyclability. For example, "Minimise the number of different types of materials" makes reference to the ability to sort materials for different end-of-life processes, which in turn has a positive influence on recyclability. This allows the identification of two interconnected heuristics: "diversity of materials" positively influences "ability to sort materials" which positively influences "product recyclability." This process allows for a more staged structuration of heuristics.
3. Some heuristics had to be split into two because they bore different propositions. For example, "Hazardous parts should be clearly marked and easily removed." While this sentence refers to one single artefact (hazardous parts), it refers to two different properties of this artefact (the presence of marking and the easiness to remove).
4. Some pairs of heuristics found to be redundant, that is, different formulations of the same idea. For example, "Ensure compatibility of ink where printing is required on plastic parts" and "Eliminate incompatible labels on plastic parts." In both cases, the addressed artefacts are sets of materials (labels and plastics) and the addressed property is their compatibility for recycling. The difference between both sentences lies in the formulation: the first is positive (presence of compatibility) and the second uses a double negation (absence of incompatibility).

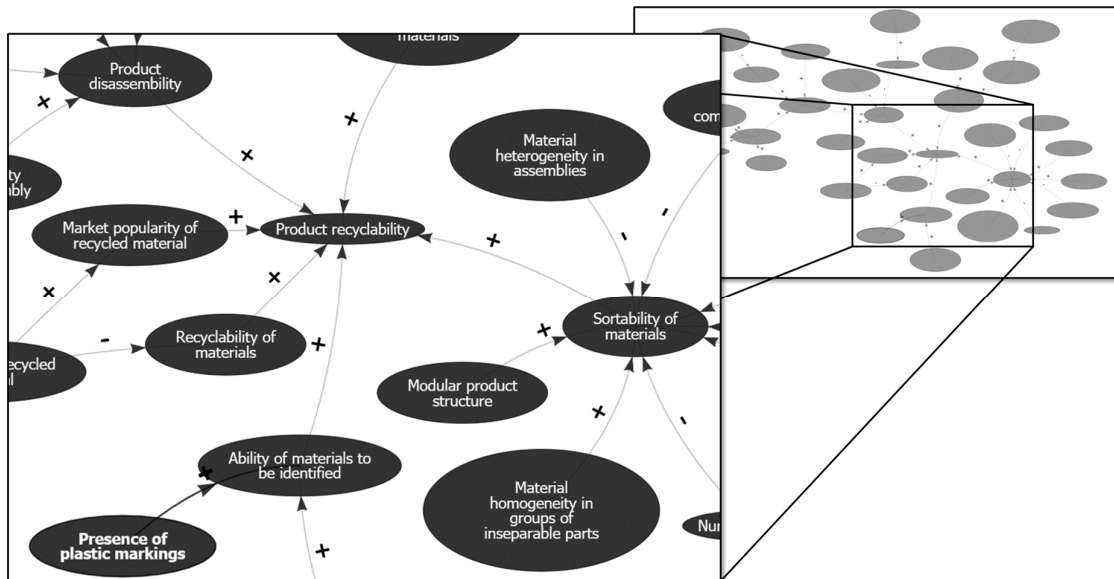


Figure 5. Set of design heuristics translated in the introduced formal system

At the end, the translation process produced a graph made of 34 edges—all directly or indirectly connected towards the node "product recyclability". This graph is partially displayed in Figure 5 and fully reproduced in the Appendix 2. 34 heuristics can be directly read out of this graph. Thanks to the staging in up to six hierarchical levels, additional indirect heuristics can be read using the transitiveness of causal relations. The graph additionally shows one trade-off situation regarding usage of recycled

material. On the one hand, the usage of recycled content contributes to stimulate the market for this material, participates to the building of treatment facilities and therefore positively influences the product recyclability. On the other hand, some material such as plastics cannot be recycled infinitely so the use of recycled content can also influence recyclability negatively.

5. Use cases for a formal description of design heuristics

The following sections present three interlinked cases in which the introduced formal system could find useful applications: introduction of new heuristic sets, assistance to product design, and development of computational applications based on heuristics.

5.1. Introduction of new heuristic sets

The example reported in Section 4 illustrates that representing a heuristic set as a causal graph allows deriving quality measures based on graph properties. For example, a high connectivity of the graph (that is, a high average number of edges per node) is an indicator of high coherence in the heuristics set. On the contrary, the presence of unreachable nodes in the graph indicates the presence of heuristics which are disconnected with the rest of the set. The variance in the number of edges per node, the connectivity (the minimum number of elements that need to be removed to generate disconnected subgraphs), as well as the clustering coefficient can be used as indicators for the homogeneity of the heuristics set as well. A heuristic set represented in a causal graph is per definition non-redundant, as long as only one node has been recorded for each artefact property.

As a consequence, adopting a more formal representation helps increasing the quality of heuristics sets, in particular in terms of consistency, specificity and non-redundancy. Hence, the introduced formal system can provide benefits for engineering research when presenting new sets of heuristics or by practitioners to better adopt existing ones.

5.2. Product development support

The introduced formal system can be used as a basis for knowledge management and decision support in product development as well. By making heuristics available in the form of a network it becomes possible to identify possible side effects or contradictions of heuristics. For example the heuristic: "Make designs as modular as possible, with separation of functions" presented by Go et al. (2015) for DfR may contradict the heuristic "Specify lightweight materials and components" from the perspective of DfE (Telenko et al., 2016). Lightweight design aims at saving energy in product usage and/or production. For realizing lightweight design, several functions are often integrated in one component and prevent the idea of modularization aiming at an increase of material and component service time.

The formal system can also be used to create heuristics sets for a specific product or industrial branch. While most available heuristics presented in research are rather generic formulations they can be less useful in concrete decision situations. For example in Design for Sustainability (DfS) different product types cause varying environmental impacts in different phases of the product lifecycle. Heuristics for reducing energy demand in the usage phase are for example not useful for products having most of their impacts in production phase. Furthermore a heuristic such as: "Minimize energy and material loss" (Telenko et al., 2016) requires an in-depth understanding of effects related to changes of product characteristics (material, geometry, etc.) and technical product properties (weight, efficiency, recyclability etc.). Currently, engineers primarily use classical desk research or the expertise of colleagues to educate themselves about cause and effects of engineering decisions related to their product. A network consisting of generic and product-specific heuristics can be used to get a better understanding of a product and may rationalise engineering education.

5.3. Implementation in computational applications

Another advantage of the introduced formal system is to support computer aided manipulation of design heuristics. The introduced formal system can be implemented in an IT-tool able to create selective views on a network of heuristics, to identify direct or indirect relations between artefact properties, to break down design targets or to analyse engineering change scenarios including possible side effects and trade-

offs when following heuristics. In order to inquire these potentials, the authors developed the Design Dependency Visualisation Tool, an in-browser application based on the JavaScript visualization library Vis.js (www.visjs.org). The source code of this software is available online (www.github.com/jbon/design-dependencies) and is released under an Apache 2.0 license, granting any interested person with freedom to use and to contribute to further development.

The basic functionality of this tool is to enable engineers to declare, sort and manipulate causal relations between artefact properties and to allow an ergonomic and intuitive interaction with causal graphs. The graph presented in Figure 5 has been produced with this application. Further additional functions are aimed to support product development as well as design education. Two use cases are particularly addressed:

- Definition of design targets: Decomposition of overarching design performance indicators (e.g. cost, environmental impact, or flexibility) to more product-specific operational parameters. Design engineers get an overview of the factors influencing the targets which have been set and can better judge how targets can be achieved.
- Evaluation of engineering change impact: Show the consequences of a parameter change on other parameters and particularly on overarching performance indicators. The user is supported in decision making by identifying all possible consequences of a change through propagation along different product parameters. The application allows checking whether a change in a parameter contributes to targets or inhibits target achievement.

6. Outlook

The new elements of theory presented in this article support the formulation of well-formed heuristics and higher quality in the report of heuristics sets. It also allows for the implementation of design heuristics manipulation software which could support industrial designers and product development engineers in their daily activities. These new elements are however partly based on some hypotheses which should be verified through further research.

The first one has been formulated in Section 3.3: any change of an artefact property can be formulated either as an increase or as a decrease in the value of this property. Although this hypothesis seems reasonable at this point, it may need to be confirmed. Particularly of interest is not only whether this statement is valid, but also whether forcing the formulation of changes in terms of increase or decrease lead to information losses. The second hypothesis is the underlying motivation for this article: presenting design heuristics in a strict formal system offers more usability than existing formats. The introduced formal system allows investigating the role of display formats on heuristics appropriation and therefore delivers the opportunity to test this hypothesis. It also allows investigating the potential impact of design heuristics on the design activity and on the produced designs.

Further research approaches would be to validate the applicability of the introduced formal model to larger heuristics sets. Sufficient material has been published, as Bauer (2007) and Fu et al. (2016) reported to have gathered respectively 300 and 858 heuristics from existing literature. This process would be the opportunity to investigate the level of subjectivity of the translation process, for example by assessing the inter-coder reliability of two people translating existing heuristics to this formal system. This would be as well the opportunity to make existing design heuristics publicly available throughout the in-browser application developed by the authors.

References

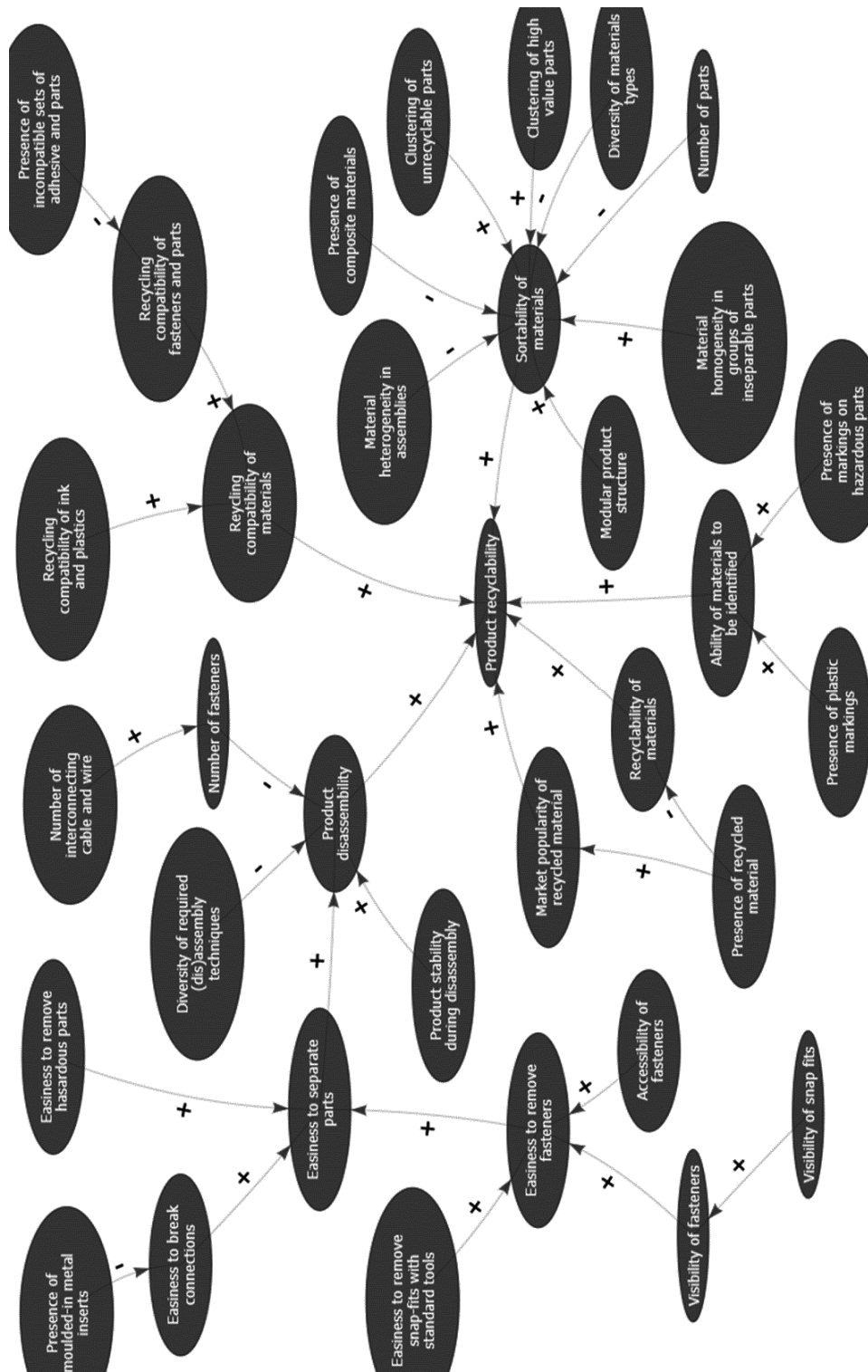
- Aamodt, A. and Plaza, E. (1994), "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches", *AI Communications*, Vol. 7 No. 1, pp. 39–59. <https://doi.org/10.3233/AIC-1994-7104>
- Altshuller, G.S. (1984), *Creativity as an exact science*, Gordon and Breach.
- Barbosa, G.F. and Carvalho, J. (2013), "Design for Manufacturing and Assembly methodology applied to aircrafts design and manufacturing", *IFAC Proceedings Volume*, Vol. 46 No. 7, pp. 116–121. <https://doi.org/10.3182/20130522-3-BR-4036.00044>
- Bauer, S. (2003), "Design for X – Ansätze zur Definition und Strukturierung", *Proceedings of the 14th Symposium on Design for X - DFX 2003, Neukirchen/Erlangen, Germany, October 13-14, 2003*, The Design Society.

- Bischof, A. and Blessing, L. (2008), "Guidelines for the development of flexible products", *Proceedings of the DESIGN 2008 / 10th International Design Conference, Dubrovnik, Croatia, May 19-22, 2008*, The Design Society, Glasgow.
- Bonvoisin, J., Mathieux, F., Domingo, L. and Brissaud, D. (2010), "Design for energy efficiency : proposition of a guidelines-based tool", *Proceedings of the DESIGN 2010 / 11th International Design Conference, Dubrovnik, Croatia, May 17-20, 2010*, The Design Society, Glasgow, pp. 629–638.
- Buchert, T., Halstenberg, F.A., Bonvoisin, J., Lindow, K. and Stark, R. (2017), "Target-driven selection and scheduling of methods for sustainable product development", *Journal of Cleaner Production*, Vol. 161, pp. 403-421. <https://doi.org/10.1016/j.jclepro.2017.05.067>
- Dahlström, H. (1999), "Company-specific guidelines", *Journal of Sustainable Product Design*, Vol. 8, pp. 18–24.
- Fu, K.K., Yang, M.C. and Wood, K.L. (2016), "Design Principles: Literature Review, Analysis, and Future Directions", *Journal of Mechanical Design*, Vol. 138 No. 10, p. 101103. <https://doi.org/10.1115/1.4034105>
- Go, T.F., Wahab, D.A. and Hishamuddin, H. (2015), "Multiple generation life-cycles for product sustainability: the way forward", *Journal of Cleaner Production*, Vol. 95, pp. 16–29. <https://doi.org/10.1016/j.jclepro.2015.02.065>
- Gonçalves, M., Cardoso, C. and Badke-Schaub, P. (2016), "Inspiration choices that matter: the selection of external stimuli during ideation", *Design Science*, Vol. 2, p. e10. <https://doi.org/10.1017/dsj.2016.10>
- ISO (2004), *ISO/IEC 14977:1996 -- Information technology -- Syntactic metalanguage -- Extended BNF*. [online] Available at: <http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html>.
- Kramer, J., Daly, S., Yilmaz, S. and Seifert, C. (2014), "A Case-Study Analysis of Design Heuristics in an Upper-Level Cross-Disciplinary Design Course", *Proceedings of the 2014 ASEE Annual Conference, Indianapolis, USA, June 15-18, 2014*, American Society for Engineering Education.
- Kruse, L.C. and Seidel, S. (2017), "Tensions in Design Principle Formulation and Reuse", *Proceedings of the 12th International Conference on Design Science Research in Information Systems and Technology (DESRIST) 2017, Karlsruhe, Germany, May 30 – June 1, 2017*, KIT, Karlsruhe, pp. 180-188.
- Lofthouse, V. (2006), "Ecodesign tools for designers: defining the requirements", *Journal of Cleaner Production*, Vol. 14 No. 15-16, pp. 1386–1395. <https://doi.org/10.1016/j.jclepro.2005.11.013>
- Matthews, P.C., Blessing, L.T.M. and Wallace, K.M. (2002), "The introduction of a design heuristics extraction method", *Advanced Engineering Informatics*, Vol. 16 No. 1, pp. 3–19. [https://doi.org/10.1016/S1474-0346\(02\)00002-2](https://doi.org/10.1016/S1474-0346(02)00002-2)
- Russo, D., Rizzi, C. and Spreafico, C. (2017), "How to Build Guidelines for Eco-Improvement", *Proceedings of the International Conference on Sustainable Design and Manufacturing / Sustainable Design and Manufacturing 2017*, Springer, Cham, pp. 879–887. https://doi.org/10.1007/978-3-319-57078-5_82
- Sarnes, J. and Kloberdanz, H. (2015), "Heuristics guidelines in eco-design", *Proceedings of the ICED 15 / 20th International Conference on Engineering Design, Vol. 1, Milan, Italy, July 27-30, 2015*, The Design Society, Glasgow, pp. 171-180.
- Telenko, C., O'Rourke, J.M., Conner Seepersad, C. and Webber, M.E. (2016), "A Compilation of Design for Environment Guidelines", *Journal of Mechanical Design*, Vol. 138 No. 3, p. 031102. <https://doi.org/10.1115/1.4032095>
- Telenko, C., Seepersad, C.C. and Webber, M.E. (2009), "A Method for Developing Design for Environment Guidelines for Future Product Design", *Proceedings of the ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE2009), San Diego, USA, August 30 – September 2, 2009*, ASME, pp. 291–302. <https://doi.org/10.1115/DETC2009-87389>
- Vezzoli, C. and Sciamia, D. (2006), "Life Cycle Design: from general methods to product type specific guidelines and checklists: a method adopted to develop a set of guidelines/checklist handbook for the eco-efficient design of NECTA vending machines", *Journal of Cleaner Production*, Vol. 14 No. 15-16, pp. 1319–1325. <https://doi.org/10.1016/j.jclepro.2005.11.011>
- Wimmer, W. and Züst, R. (2001), *ECODESIGN Pilot: Product Investigation, Learning and Optimization Tool for Sustainable Product Development with CD-ROM*, Kluwer Academic Publishers. <https://doi.org/10.1007/0-306-48393-9>
- Yilmaz, S. and Seifert, C.M. (2011), "Creativity through design heuristics: A case study of expert product design", *Design Studies*, Vol. 32 No. 4, pp. 384–415. <https://doi.org/10.1016/j.destud.2011.01.003>
- Yilmaz, S., Daly, S.R., Christian, J.L., Seifert, C.M. and Gonzalez, R. (2013), "Can experienced designers learn from new tools? A case study of idea generation in a professional engineering team", *International Journal of Design Creativity and Innovation*, Vol. 2 No. 2, pp. 82-96. <https://doi.org/10.1080/21650349.2013.832016>
- Yilmaz, S., Daly, S.R., Seifert, C.M. and Gonzalez, R. (2015), "How do designers generate new ideas? Design heuristics across two disciplines", *Design Science*, Vol. 1, p. e4. <https://doi.org/10.1017/dsj.2015.4>

Appendix 1: 27 DfR heuristics from Go et al. (2015, p. 21)

Materials	Minimise the number of different types of materials
	Make subassemblies and inseparably connected parts from the same or a compatible material
	Avoid the mixing of materials in assemblies
	Mark all plastic and similar parts for ease of identification
	Use materials which can be recycled
	Use recycled materials
	Ensure compatibility of ink where printing is required on plastic parts (duplicate)
	Avoid composite materials
	Eliminate incompatible labels on plastic parts (duplicate)
Fasteners and connection	Hazardous parts should be clearly marked and easily removed (composite)
	Minimise the number of fasteners
	Minimise the number of fastener removal tools needed
	Fasteners should be easy to remove
	Fastening points should be easy to access
	Snap-fits should be obviously located and able to be disassembled using standard tools (composite)
	Try to use fasteners of material compatible with the parts connected
	If two parts cannot be compatible make them easy to separate (redundant)
	Eliminate adhesives unless compatible with both joined parts
Product Structure	Minimise the number and length of interconnecting wires or cables used
	Connections can be designed to break as an alternative to removing fasteners
	Minimise the number of parts
	Make designs as modular as possible, with separation of functions
	Locate unrecyclable parts in one area which can be quickly removed and discarded
	Locate parts with the highest value in easily accessible places
	Design parts for stability during disassembly
Avoid moulded-in metal inserts or reinforcements in plastic parts	
Access and break points should be made obvious	

Appendix 2: Set of design heuristics translated into the introduced formal system



Dr. Jérémy Bonvoisin, Lecturer (Assistant Professor)
 University of Bath, Department of Mechanical Engineering
 4 East - University of Bath, BA2 7AY Bath, United Kingdom
 Email: j.bonvoisin@bath.ac.uk