

Guidelines for Application of the Constituent Roadmap of Product Design Based on Axiomatic Design

Erik Puik^{1,*}, Joost van Duijn¹, and Darek Ceglarek²

¹HU University of Applied Sciences Utrecht, Padualaan 99, 3584 CH Utrecht, the Netherlands

²International Digital Laboratory, WMG, University of Warwick, Coventry, CV4 7AL, UK

Abstract. The ‘Constituent Roadmap of Product Design’ (CRPD) is a framework that is applied to track the designers’ knowledge of a product design as it progresses during development. The Constituent Roadmap of Product Design was introduced in late 2015 and relies heavily on Axiomatic Design; it uses Axioms 1 and 2 to model understanding of the product- and manufacturing-design by the designer. A central theme when applying the Constituent Roadmap is a procedure of regularly updating the ‘Check Matrix’. The Check Matrix is a table that continuously monitors the designer’s knowledge position. Correct application of the check matrix is bound to a number of behavioural characteristics that need to be taken into consideration, e.g.: (i) in a decomposed system, it is not possible that a knowledge score at a certain hierarchical level exceeds that of one of its sublevels. A lack of knowledge is, similar to information in design, the sum of its subordinate knowledge deficits. (ii) on the other hand, the score in the check matrix indeed may exceed that of its lower levels during the process of zigzagging, as zigzagging is a process that advances top-down through the hierarchy of the product design.

This paper explains how to apply the check matrix under various circumstances without violating the hierarchical dependencies of good practice in design.

1 Introduction

The ‘Constituent Roadmap of Product Design’ (CRPD) is a systems-engineering tool (or an engineering design tool) to support the process of product development. The CRPD is applied to track a designer’s knowledge, and understanding of a product design, as it evolves during the product design process. The CRPD was introduced in late 2015 and it is based on the complexity theory of AD [1]. It tries to combine ease of use while maintaining scientific substantiation. The CRPD is applicable by designers in the full range from ‘novice’ to ‘experienced’. Because of this, the method is also suitable for educational purposes.

‘Knowledge of the Designer’ is the central theme of the CRPD. According to good AD-practice, satisfaction of Axioms 1 and 2 indicate that a designer has good understanding of his design. This is only possible if the designer is in possession of the necessary knowledge of that design, as the quality of a design will not exceed the knowledge of the designer [2]. The CRPD applies a number of four hierarchical levels to help the designer in the process of decomposition of the product design and its related production processes. It also distinguishes a number of three sequential development phases of the design. During the design process, the development status of the four hierarchical levels and three phases of the CRPD are tracked using a table that is called the ‘Check Matrix’. The check matrix is a leading tool in the

application process of the CRPD. This paper discusses a number of behavioural guidelines to enable simple and correct application of the check matrix.

The paper is organised as follows. Section 2 explains the background of the CRPD and how it is applied. Section 3 analyses the application of the CRPD and determines the issues that need to be well understood for proper application of the CRPD. Section 4 explains the methodology of investigation, and Section 5 shows a case in which this is applied. Finally, Section 6 discusses the findings and draws conclusions.

2 Background on the constituent roadmap of product design

The CRPD builds further on the design matrix of AD. The design matrix forces the designer to organise the conceptual capability of the design. This is being done by organisation of the following relationships: (i) Customer Attributes and Functional Requirements (CAs & FRs), (ii) Functional Requirements and Design Parameters (FRs & DPs), and (iii) Design Parameters and Process Variables (DPs & PVs). These relations (i, ii, & iii) must be properly substantiated during the product development process. AD provides solutions for substantiation, the CRPD provides overview of the development process.

* Corresponding author: erik.puik@hu.nl

2.1 Constituent roadmap of product design

The CRPD is a merged matrix of 7x4 cells that embeds product hierarchy on the vertical axis. The hierarchy that is applied in the CRPD is borrowed from the widely applied V-Model, and in particular its German variant; the ‘V-Modell XT’ [3]. The V-Modell XT defines a hierarchical division in four levels: (i) Project, (ii) Product, (iii) Systems, and (iv) Parts. This hierarchy has proven its use in industrial environments.

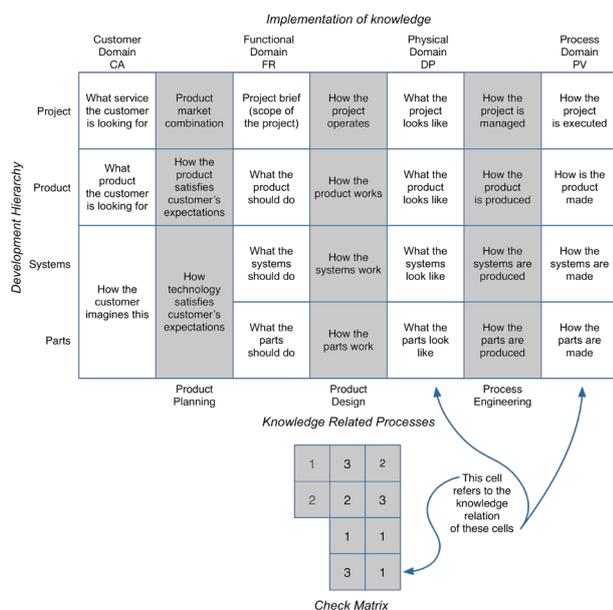


Fig. 1. The CRPD and its check matrix

The domains of AD are represented by the odd columns in Figure 1. The domains of AD contain factual descriptions of respectively: (indicates column numbers) (1) what the customer expects, (3) what the system should do, (5) how this is realised, and (7) how this can be achieved. The designer will apply his knowledge to the domains and as this happens ‘the designers’ knowledge’ turns into ‘applied knowledge’. Applied knowledge is directly related to the product design. Therefore, applied knowledge is static. It consists of lifeless descriptions of which knowledge was once the foundation. As a result, the applied knowledge in the columns will not change unless the designer executes his design activities again and updates the applied knowledge.

The even columns store ‘descriptive knowledge’ that was necessary for the designer to generate the static descriptions in the odd columns. As its name indicates, descriptive knowledge describes the knowledge that underlies: (again column numbers) (2) the process of product planning, (4) the product design, and (6) the process engineering processes.

2.2 The check matrix

Descriptive knowledge, as exerted by the designer, is the source of current and future understanding of the design. It is stored in the form of high-level knowledge

structures like design matrices that define solid design relations. This is where the check matrix proves its value; it keeps track of descriptive knowledge that is directly related to understanding of the design. If enough descriptive knowledge is present with the designer, the design may be considered to be well-understood; it may be considered to be a ‘good design’ [4]. Using the check matrix, only a number of ten relations that are represented by numerical values, are sufficient to assess the total product design, including customer interactions and processes.

Two restrictions apply; the CRPD applies two general assumptions that (i) designers are idealistic people with no double agenda that simply pursue a dependable and reliable product design, and (ii) they are granted enough time and means to apply their descriptive knowledge to the design. As a result, adequate knowledge of the designer will lead to a good design and all FRs will be satisfied [1]. Under these presumptions, the only binding condition to guarantee a good design is to make sure that the knowledge of the designer, represented in the even rows, is adequate to guarantee a correct relation between the cells left and right of the even rows (lower right, Figure 1). This relation is weighted and scored with a value in the check matrix.

2.3 Three sequential development phases

The CRPD distinguishes a number of three phases in the design process [1]. The phases are respectively: (i) ‘Exploration’, (ii) ‘Conceptual’, and (iii) ‘Robustness’:

- The Exploration phase leads to an initial product/process definition and exposition. It creates coherence in the process of product design by definition and organisation of temporary CAs, FRs, DPs, and PVs of new and adolescent elements in the design;
- The Conceptual phase will lead to product/process specification and decomposition. Conceptualisation verifies the concept by completion and decoupling of the product and process design matrices. The result is that product and processes become completely decomposed and specified (i.e. proof of concept). A completed conceptual phase is equivalent to satisfaction of the Independence Axiom;
- Finally, the Robustness phase is applied to enforce a statistically sound relation between PVs, DPs, and FRs. A completed robustness phase is equivalent to a satisfied Information Axiom.

The statuses of the design relations in the CRPD are represented in the check matrix with a number from 0–3, directly referencing to the best-completed development phase of the particular relation, this is shown in Figure 2:

0. Status of Project, Product, System, or Part unknown;
1. Exploration phase was successfully completed;
2. Conceptual phase was successfully completed;
3. The robustness phase was successfully completed.

Using these four statuses, the check matrix can be applied as a measure to track progression of the product development process.

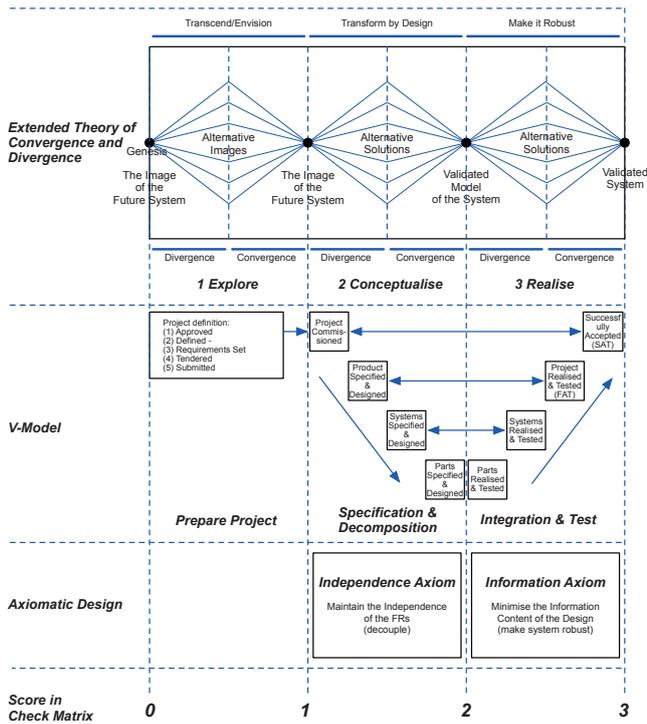


Fig. 2. The three phases of the design process that form the basis of the CRPD

3 Approach

3.1 Analysis

The representation with the check matrix was introduced late 2015 by Puik et al. [1] and its application is bound to a number of guidelines. The guidelines are the main topic of investigation in this paper.

Problem statement

Early attempts to apply the CRPD sometimes showed quite erratic check matrices with numbers 0–3 scattered all over the matrix, especially in the beginning of the project. The question is what would be good practice for application of the check matrix during the process of design:

What are the rules of conduct when rating the design relations in the check matrix of the CRPD (horizontally and vertically).

Current situation

Firstly, the CRPD is inspired by the V-Model XT. The V-Model has a preferred path in phases 2 and 3 that are also applicable and even preferred for the CRPD.

Secondly, AD uses the procedure of zigzagging to address the design relations top-down during the conceptual phase.

Thirdly, Puik et al. [5] have introduced the concept of ‘reverse zigzagging’ to be applied during the robustness phase. Both methods, zigzagging and reverse zigzagging are adopted as a temporary guideline for determination of the values in the check matrix.

Key limitations

At this point, there are no proven procedures for good practice to apply for the check matrix when applying the CRPD. A substantiated guideline could improve application of the CRPD from the perspective of ease as well qualitative.

3.2 Methodology

Project flow of the V-Model

The project flow of the V-Modell XT was accurately described by Höhn et al. [3]. Figure 2 shows this flow in relations to the three phases of the CRPD. The project definition corresponds with the Explorative phase. It can also be seen that the ‘left leg of the V-Model’ is applied for the conceptual phase and ‘the right leg’ is applied for the robustness phase. The transition from the conceptual to the robustness phase takes place when the product is fully decomposed and specified, which is the case at the tip of the ‘V’.

The project flow in the V-Model during conceptual phase is top down, which means that higher hierarchical levels are specified first followed by the lower levels. During the robustness phase, this is reversed and the flow is bottom up.

Zigzagging

Zigzagging is standard procedure of AD that was described by Suh [4]. Zigzagging in AD adds an important aspect to the project flow in comparison to the V-Model; it connects the different domains. Analogue to the V-model, the flow is top-down but zigzagging also structurally verifies the design relations (CA)-FR-DP-PV increasing the score from 1 → 2. Zigzagging cannot be executed in the V-Model because that model does not differentiate between the customer, functional, physical, or process domains.

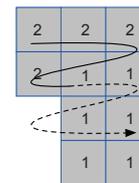


Fig. 3. Project flow through the check matrix when zigzagging

Reverse zigzagging

Reverse zigzagging is applied during the robustness phase. The basic principle is comparable to the standard zigzagging process but the hierarchical direction is reversed from top-down to bottom-up and from left-right to right-left. After successful values of the check matrix are increased from 2 → 3.

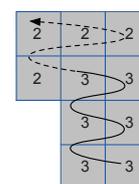


Fig. 4. Project flow through the check matrix when reverse zigzagging is applied

The project flow of the V-Model as well as basic and reversed zigzagging seem fully compatible; though the hierarchical direction during exploration is not really set, this indeed is the case during the conceptual phase (top-down) and the robustness phase (bottom-up). Zigzagging adds the project flow through the domains, where the V-Model has no capability because the domains are not applied for the V-Model.

How are the key limitations addressed?

To substantiate the procedures for application of the check matrix, the constituent roadmap will be applied to a product development case. The flow of the V-Model, zigzagging and reverse zigzagging will be applied. The project, and how the check matrix develops, will be accurately monitored during this process.

4 Case study: Development of a biglide-manipulator for 3D printers

The applied case concerns a novel manipulator for a newly developed 3D ‘Fused Deposition Modelling’ (FDM) printer (Figure 5). This manipulator is based on a ‘Biglide’ servo system [6].

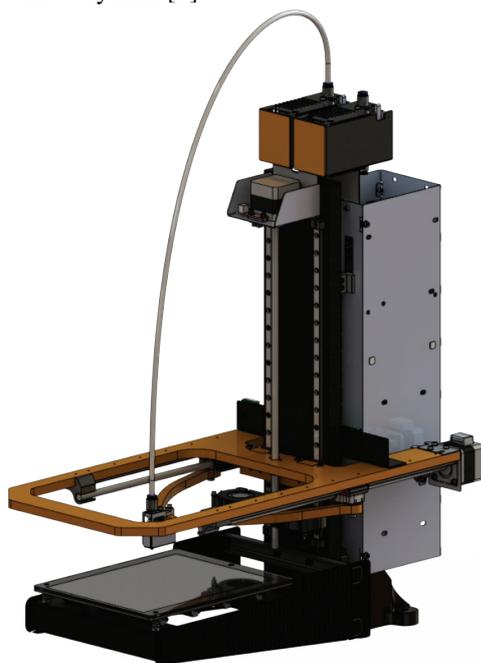


Fig. 5. Biglide-manipulator applied for 3D printing

Figure 6a shows a top-view of the manipulator, Figure 6b a schematic representation of the kinematics. The biglide-system uses a double linear guideway that is converted to an X-Y movement by a double arm. The arm uses a flexible joint near the tool centre point (TCP) to compensate for the distance variations between the slides A and B. In Z-direction, a similar linear guideway is applied for an independent perpendicular manipulation of the bi-glide system.

The accuracy of the manipulator is subject of investigation.

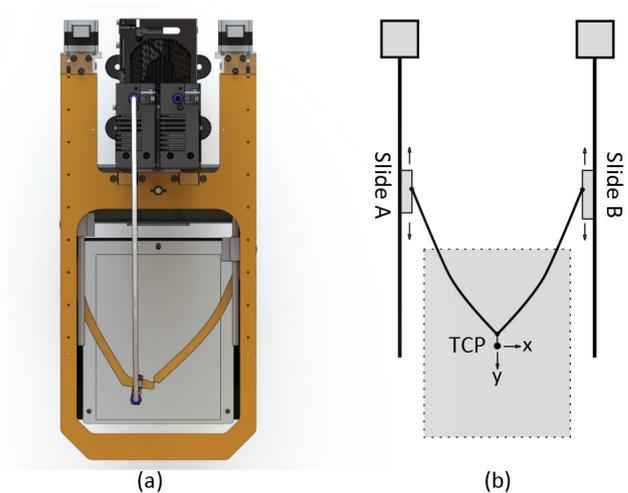


Fig. 6. (a) Top view of biglide-system, (b) kinematic model

4.1 Exploration phase

The exploration phase aims to define the ‘Image of the Future System’ (figure 2). It addresses a lot of uncertainties in the project, not only from a technological perspective but also from the marketing perspective. Analyses were executed for:

- (i) Customer interactions (CA-FR);
- (ii) Product development (FR-DP);
- (iii) Process development (DP-PV).

(i) In this case, the manipulator was developed to replace an existing servo system. However, the change of manipulator was not really perceptible from the customer point of view. This left the CA-FR relation unaffected, leaving the product-market combination unaffected. The product complies when the new manipulator is able to satisfy the FRs.

(ii & iii) System engineering tools were applied to gain knowledge of the desired manipulator and its interfaces with the rest of the system design; it was functionally decomposed using QFD and IDEF-0 for respectively the product- and the process-designs.

The manipulator was realised as pre-prototype (test setup for functionality testing). Initial tests were performed and the system had the capability of producing reasonable quality 3D prints. Materials for realisation were determined from the perspective of functionality, manufacturing and cost price. Preliminary relations were determined for the mapping of FRs-DPs and DPs-PVs.

A number of alternative options were kept open and were tested in parallel (different options for spindles and motors and many options for the flexible hinge).

3	1	1
3	1	1
	1	1
	1	1

Fig. 7. Check matrix after completion of the explorative phase

Figure 7 shows the check matrix after the explorative stage. At this point, decoupling and robustness of the design relations cannot yet be guaranteed.

4.2 Conceptual phase

The design should be decoupled to acquire conceptual rigour of the design. This means that the Independence Axiom is met, guaranteeing that the FRs indeed can be satisfied with the chosen DPs (same for DPs & PVs). To enable this, the system is accurately modelled by CAD; kinematics and tolerances are simulated. The process of zigzagging got stuck due to coupling at systems level. This was caused by: (i) double-interfaces of the mechanical lay-out, and (ii) conflicts of the electrical-connections crossing the modular framework, both problems leading to coupling of the design. To address the problems, the modular capability of the manipulator was improved by extending the modular decomposition of the 3D-printer by use of an 'Entity Relationship' model, unravelling complicated interrelations in the modular structure. Mechanical and electrical components were reordered for an independent modular layout. This process was implemented in a prototype. To close the conceptual phase, zigzagging was resumed to verify the independence of the product- and process-design matrices. This time the process was successful down to the parts level of the design. The check matrix was upgraded to '2's comparably to the '1's in Figure 6.

4.3 Robustness phase

The third phase, revealed a weakness in the mode of implementation of the biglide-manipulator. Though X- and Y-positioning were fast and accurate, the system had an adverse effect on the Z-positioning. When applying the process of reversed zigzagging, it appeared that the Z-accuracy could not satisfy the aimed values of the FRs. Analysis learned that the inaccuracy of some of the parts in the slides (Figure 5) were amplified by the length of the biglide-arm. A lack of knowledge on the implementation of the biglide-mechanism, led to reduced accuracy of the realised prototype. The initial construction was gradually improved by a tuning process with a number of iterations. After this, the reverse zigzagging process was restarted and completed all the way up. The acquired knowledge on the initial prototype has enabled modification of the geometry of the biglide-mechanism. It realised a major improvement on robustness of the system. The re-engineered biglide-servo-system eliminated all the conflicts between FRs, DPs, and PVs.

5 Discussion

In this case study, the check matrix was regularly updated during the ongoing engineering process of the manipulator. It allows the engineers to keep track of the development process as it progresses. By doing so, the matrix supports the go/no-go assessments that determine if the next phase can be started.

During this exercise, the expectations for application of the check matrix were largely confirmed. Note that from elemental perspective, values at higher hierarchical levels in the check matrix cannot have higher scores in the check matrix, because errors escalate hierarchically upwards through the decomposition tree (and therefore also upwards through the check matrix). The exception is the process of zigzagging. Zigging moves top-down through the matrix, as shown in Figure 3, and increases the values in the check matrix from '1' to '2'. By doing this, it follows the same direction as the V-Model where 'Specification and Decomposition' is applied in the left leg and also moves from top to bottom. However, when the upper cells in the check matrix are advanced to '2', this is under the assumption that the lower cells will follow. If this is not the case, the process of zigzagging will stop and the higher values need to be set back to their old value ('1').

Strengths of this approach

The CRPD is based on AD and because of this it shares its rigour. The check matrix adds optimal overview of the project for the designer. This is done by tracking knowledge of the designer that is needed to complete the design.

Application of the check matrix enforces a gradual reduction of project risks as the project evolves. It prevents different risk levels in the project from drifting apart. When larger risks remain present in the project, it may lead to engineering efforts being lost; finished and well-engineered parts of the design may need to be altered to address the risks, and as a result the engineering process needs to be redone.

Weaknesses of this approach

Systems engineering methods typically cause overhead for the designer. These efforts are only valuable if it saves the designer from executing repair actions for the design. The method indeed is able to save the designer from rework as errors are found in an early stage and do not aggravate for longer times.

The decomposition tree of a product design gets a significant width at the bottom of the tree. The score in check matrix always represents the lowest value of any one of the decomposed elements. If there is only a single low scoring element it gives the same score as when there are many low scoring elements. This muddies the view of the designer.

Other Considerations

The check matrix uses a limited amount of numbers from 0 to 3. The check matrix could be tailored to focus on systems representing the sub-developments, enabling more detailed tracking.

References

- [1] E. C. N. Puik and D. Ceglarek, "Axiomatic Product Design in Three Stages: A Constituent Roadmap That Visualises the Status of the Design Process by Tracking the Knowledge of the Designer," *ASME 2015 International Mechanical Engineering*

- Congress and Exposition*, pp. V015T19A003–V015T19A003, Nov. 2015.
- [2] E. C. N. Puik and D. Ceglarek, “The Quality of a Design Will Not Exceed the Knowledge of its Designer; an Analysis Based on Axiomatic Information and the Cynefin Framework,” *ICAD2015*, vol. 34, pp. 19–24, 2015.
- [3] R. Höhn, S. Höppner, M. Broy, A. Rausch, R. Petrasch, S. Biffel, R. Wagner, W. Hesse, and K. Bergner, *Das V-Modell XT*, 1st ed. Heidelberg: Springer, 2008.
- [4] N. P. Suh, “Axiomatic Design - Advances and Applications,” in *advances and applications*, Oxford University Press, New York.Oxford, 2001, Chapter 5, pp. 239-298., 2001.
- [5] E. C. N. Puik, E. Smulders, J. Gerritsen, B. V. Huijgevoort, and D. Ceglarek, “A Method for Indexing Axiomatic Independence Applied to Reconfigurable Manufacturing Systems,” *ICAD2013*, vol. 7, pp. 186–194, Jun. 2013.
- [6] S. D. Stan, R. Balan, and V. Maties, *Optimal design of parallel kinematics machines with 2 degrees of freedom*. 2008.