TOWARDS REAL-TIME FEEDBACK ON MANUFACTURABILITY FOR ENGINEERING DESIGNERS DIRECTLY FROM MANUFACTURERS

Weißenbach, Paul; Gerhard, Detlef
TU Wien, Austria

Abstract

An approach to provide automatically generated feedback on manufacturability for engineering designers without breaking their flow of thought is presented. The origin of the feedback should be knowledge bases maintained by manufacturers. Therefore, the feedback provider is from the perspective of the engineering designer behind a network. This “remote design checks” introduces additional aspects regarding latency and intellectual property protection, which are incorporated into a proposed distributed software architecture. Feedback is provided directly in engineers authoring tools (CAD software). A proof of concept implementation was created. Architecture and proof of concept are evaluated and discussed on the basis of a use case based on aluminium extrusion.

Keywords: Digital / Digitised engineering value chains, Computational design methods, Virtual Engineering (VE), Remote design checks, Business models and considerations

Contact:
Paul Christian Weißenbach
TU Wien
Institute of Engineering Design and Logistics Engineering
Austria
paul.weissenbach@tuwien.ac.at

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 5: Design for X, Design to X, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

Engineer's time and capabilities are valuable resources. Learning about production technologies and how a design becomes more suitable to manufacture is time-consuming and introduces iterations. The assumption is that a design, which has been created with low knowledge about a particular production technology, has to be re-designed (iterated over) after engineering designers have acquired more knowledge and capabilities. The presented approach tries to minimise the duration of this kind of iterations to a level where the engineering designer's flow of thought is uninterrupted.

Automation is applied in order to reduce response times for feedback to almost zero. This raises the question of what feedback in engineering design on manufacturability can be answered by algorithms. The kind of feedback and a path to create it automatically is dependent on one's capabilities and state of the art of useable algorithms and implementations. Both, the technology and capabilities change over time. Therefore, the more lasting question is who has the benefit to develop, run and maintain these algorithms. The main assumption in this paper is that it is the manufacturers. They are the ones who have a valid interest to provide engineering services to their (potential) customers as a product service system.

Product engineers who are not experts in all available production technologies have different approaches. One example could be: A product engineer creates a model in an authoring software tool, e.g. Computer Aided Design (CAD). Not sure on how to manufacture the model, the engineer seeks help from production engineers working with the company's suppliers. The product engineer contacts the production engineers with the initial design and gets basic feedback information on how the design could be more suitable to be produced with a specific technology. This is usually time-consuming, since all involved persons are engaged and the feedback quality often is not quite satisfying. The actual time spend on creating the feedback is small in comparison to the total duration of the feedback loop. Assuming that most of the feedback can be automatically generated, the question is how it should be and if it can be provided to product engineers.

We propose a distributed software architecture as an answer to this question in this paper. The goal is to provide feedback within a second. The reason behind a sub second time frame is, that it doesn't interrupt the engineer's flow of thought. A software user's flow stays uninterrupted if the application's response is within a second (Card et al., 1991; Miller, 1968). The assumption is that engineers whose flow of thought is uninterrupted can achieve more and therefore performs better. The overhead of getting into the right mental state to solve a design problem (context switches) are in total reduced.

If feedback is presented almost immediately, engineers should not need to switch software tools to perceive it. The premise is that engineering designers who need feedback on manufacturability mainly use CAD tools for modelling. They use more software tools but they are not in the scope of this article. The feedback should be given in an unintrusive way in order to not unnecessarily interrupt the engineers' flow of thought. A comparable system is spell checking in popular word processing software. Words the system assumes to be written wrong are underlined without stopping the user from typing. Spell checkers can additionally make suggestions to for correction. They can be triggered by the user and run in the background. Taken into the realm of CAD modelling software, a dimension in a model could be underlined if it cannot be manufactured with a configured production technology. If a dimension is driving the model, feedback can suggest changing the dimension to a minimum threshold.

Ulterior goal behind this effort is to speed up whole development endeavours. The assumption is that delays on "small" feedback loops while designing can severely affect engineering design projects in its entirety. Small delays can add up because feedback loops are intervened. When feedback from one source is integrated, feedback on another (or even the same) subject becomes necessary. This repeats itself and is one explanation for the iterative nature of engineering design. If waiting time can be eliminated in commonly occurring cases, the overall development time will be reduced.

The quality of the feedback matters itself. The focus of this paper is on how to provide real-time feedback from a service managed by manufacturers and how to integrate it in the engineers authoring software. How good the quality of specific feedback on a specific technology is unaddressed. Wiggins (2012) lists characteristics good feedback has to incorporate. One that should be mentioned is actionable.

Aim is to have better accessible and continuously improving knowledge base systems for designers. This paper contributes towards this goal with a distributed architecture to make it easier to develop, test and deploy knowledge bases, which supports engineers with hints on better manufacturability.

2 RELATED WORK

The described approach falls into cloud based design and manufacturing even though it does not fulfil all of Wu et al.'s (2015) requirements. An overview of cloud based engineering systems is given in (Wu et al., 2016). Wang et al. (2002) is a good starting point for such systems with a focus on conceptual design. In comparison, the presented approach is unique in its combined focus on immediate feedback (and as a result its level of automation), location where the feedback is presented (authoring tools) and the sources of the feedback (multiple manufacturers).

2.1 Academic implementations

A similar approach has been proposed by Hochmuth et al. (1997). Information on tolerances are collected from a "Konstruktionssystem" and used to calculate the tolerances feasibility. A report is generated and provided to the engineer. The report is rendered as a web page and accessible through the authoring tools user interface. Who or where the computation is executed is not addressed. Marking the differences to the presented approach as the focus on computation "behind" a network.

Zissis et al. (2016) explore a web based "Collaborative CAD/CAE System as a Service" that is provided over the internet to facilitate the benefits of SaaS (Software as a service) often referred to as cloud computing. The presented approach tries to reap the same benefits. Namely, scalability, faster deployment of software changes while benefiting end-users with convenient IT functions lacking hardware installations upgrades and maintenance (Gold et al., 2004). A similar, commercial available system is explained in Junk (2016). The presented approach does not seek to provide a CAD environment. The goal is to integrate with existing CAD systems. Services in the proposed architecture "only" compute feedback on manufacturability. Zissis et al address the issue of integrating simulation but not explicitly feedback. Their stated focus for the future is on increasing the collaborative aspects by enabling simultaneous visualisation. Knowledge transfer will be achieved by engineers working on the same models simultaneously not through automation.

2.2 Design checks in commercial implementations

Commercial CAD systems already implement different design checks and let users create their own. An example is the distance of punched holes to bending lines in sheet metal parts (Parametric Technology Corporation, 2016). If they are too close to each other bending will deform the hole, the system creates a report so that engineers can find critical occurrences in a model. To the authors' knowledge, none of the popular CAD systems natively supports "remote design checks", where the check is computed not on the engineer's workstation, nor do they make it easy to build them. Main benefit of remote design checks is that they can integrate current information from the shop floor (manufacturer). The assumption is that manufacturers can represent their capabilities better and do not want to depend on CAD vendors for updates when their production technology advances.

On a company level, CAD software (e.g., PTC's Creo, Dassault Systemes' Solidworks) supports "design" or "model checkers". These can be explicitly started by an engineer or automatically executed before a model is stored in a PDM system. This tools can check models and drawings for various characteristics, e.g. if all sketches are fully defined, views in drawings overlap or if the configured material is in a predefined list. The checks are mostly about consistence appearance (e.g., font, font-size) and model hygiene (e.g., suppressed features). Complex checks regarding manufacturability are unavailable.

On model level, it is possible to create different kind of measurements and attach alerts to them, e.g., the distance between two points can be measured and an alert configured to trigger if this distance falls below a threshold. Every time the model is recomputed, the condition is checked and if fulfilled a warning is displayed to the engineering. No suggestions to solve problems are given to the engineer, except which conditions are not fulfilled. These design checks are stored in the models themselves, making the product engineer responsible for them. Our assumption is that a product designer is not the expert on the used manufacturing technology and therefore checks would be better in the responsibility of a manufacturer. Advances on the shop floor have to be integrated on a per model basis, resulting in a potential disadvantage that the same work has to be done in multiple locations (models) to reflect the improvements consistently. Special software focusing on checking manufacturability exists (e.g., Geometric's DFMPro). They integrate with CAD system and include more predefined manufacturing specific rules than the options mentioned so far. They are still not as specialized enough to distinguish between different manufacturers employing the same production technology without adaption.

The purpose of CAM software is not to provide feedback, rather to actually allow transforming a model into a representation that can be used on the shop floor, e.g., G-Code instructions for a CNC machine. Information on how good a design will be to manufacture is a by-product of this transformation. We argue that, a feedback system for manufacturability does not necessarily need to work similarly to CAM tools. For fast feedback, it might be enough to have statistical data from previous production runs of similar shapes and it might be unnecessary to compute G-code and run it in a virtual environment to gather actionable feedback. Furthermore, CAM software is usually not fully automated, as manual configuration is often necessary. CAM tools integrate or are part of CAD systems.

Manufacturers have been identified (e.g., plethora.com, emachineshop.com) who provide plugins to CAD systems, which compute manufacturability and provide feedback. They support design engineers by analysing manufacturability, providing quotes and the option to order parts. The analysation is triggered manually. The functionality of these plugins is close to the presented approach, with the distinction that they are single source (single manufacturer) feedback systems not multiple manufacturing systems.

2.3 Engineering design theory

According to a study done by Abramovici and Herzog (2016), real-time decision support is the second most named requirement (after "interdisciplinarity") on engineering methods for smart products and services. Overall 61 percent do rather not and do not agree that today's IT tools are suited for the challenges of what they refer to as engineering 4.0 (in style of "Industrie 4.0"). The study made multiple recommendations. The ones this approach tries to follow are:

- "Greater use of feedback from production and product use for optimizing engineering processes", by a focus on feedback from manufacturing.
- "Development of an intercompany knowledge management", by a distributed service oriented knowledge based engineering system.
- The service-oriented approach facilitates the "adaptable IT architectures, stronger cooperation with external partners and downstream partners (production, sales and product use)" recommendation.

There are scientific design guidelines for manufacturing. A general-purpose example is design for manufacture and assembly (DFMA) introduced by Boothroyd (1994). A production specialized example can be found in (Donati and Tomesani, 2005). Material to support engineering designers is created by suppliers to help (potential) customers. One example in form of a book is (Sapa Extrusion, 2014), which is for engineers looking to design aluminium profiles. Aluminium extrusion serves as an initial use case for the overall architecture introduced.

3 USE CASE: ALUMINIUM EXTRUSION

Different types of extrusion exist. In case of warm aluminium extrusion, a heated aluminium block (blank) is pressed through a tool (die) to achieve the needed form. The result is an extruded aluminium profile. The engineering design use case of an aluminium profile can be simplified to three types of stakeholders. The engineer, who wants to use the profiles for an overall design task; the company that wants to produce and sell the result of the overall design task; and suppliers who want to produce extruded profiles and manufacture the necessary dies.

3.1 Engineer

In our use case, the product engineer is not an expert in aluminium extrusion. The engineer has no deep knowledge about die design and its many peculiarities. The goal is to use an extruded part for an overall product design. The engineer's main interest is to design a product for the companies target market. Proposed designs need to be producible and furthermore, should exploit the production technology as much as economically reasonable. Two basic examples of information the engineer might need are as follows. First, sharp corners create points with selectively high pressure on the die and reduce its lifetime and productivity. If the engineer models sharp edges, the system could suggest rounding them with minimum radius (Sapa Extrusion, 2014). Rounding can potentially cause a material collision with other parts in the overall design. Therefore, a rounded trench is a suitable additional suggestion. Both suggestions are depicted in Figure 1. a).

The second example regards tool sizes. A larger press is costlier to operate than a smaller one. Additionally, smaller dies are usually cheaper to manufacture than larger ones. If the design slightly exceeds the size of a tool, its maximum contour should be shown. Then, the engineer can decide if a smaller profile is a feasible option for the overall design. Depicted in Figure 1. b)

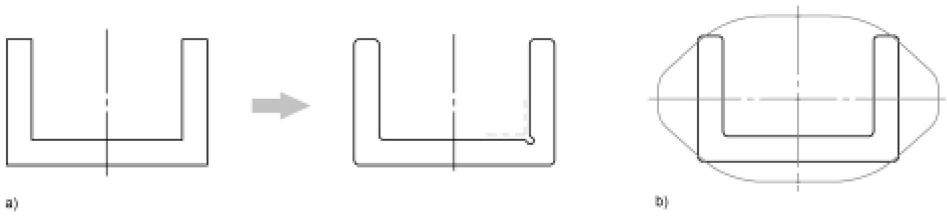

Figure 1. a) Rounding suggestion, b) Maximum profile contour

3.2 Engineer's company

The main issue of the company (besides creating and selling products) is to keep track of its intellectual property. Information about potential future products should only leave the company towards trusted suppliers. The information itself should be reduced to what the supplier needs to create feedback. Other company issues are internal guidelines, which only tangibly touch manufacturability of the profile itself. For instance, screw canals in profiles have to be standardized as the screws the for the company's products are also subject to internal standardisation. For sourcing, the company wants to identify capable suppliers. This has already affected the engineer as a trade-off. Do benefits of a design exploiting a special capability of one supplier outweigh the advantage of multiple suppliers?

3.3 Supplier

Manufactures invest in their production equipment and improve their technology from die design and operations to additional processing steps, e.g., bending, heat treatments, straightening. Their goal is to push these advances to (potential) customers and to differentiate themselves from their competitors. They are considerate about their know-how and do not want their advances leak to the competitors.

4 ARCHITECTURE

The four basic components of the distributed architecture are illustrated in Figure 2. All services are connected through web technology, i.e., HTTP or HTTPS respectively.

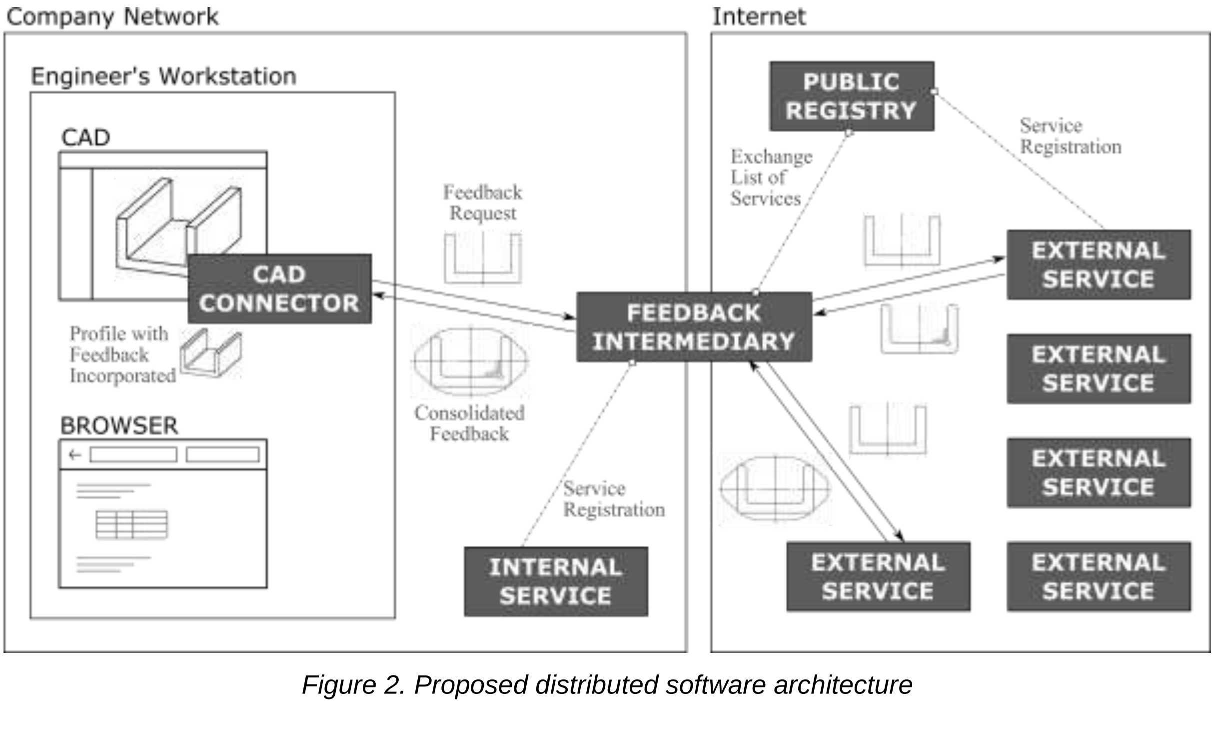

Figure 2. Proposed distributed software architecture

4.1.1 CAD Connector (authoring tool integration)

CAD tools and browsers are commonly used by engineers. Browsers are even integrated into many CAD systems. In order to extract geometry and general information from CAD and visualize feedback, a "CAD connector" is necessary. The CAD connector sends partial models to and receives data from the feedback intermediary. As different services have different informational needs, the CAD Connector is flexible and "extractors" and "visualizers" can be loaded during run time, i.e., without restarting the CAD systems. The browser allows engineers to add additional information, which could not easily be integrated to the CAD model and to configure the CAD connector itself.

4.1.2 Feedback Intermediary (company-wide proxy)

The feedback intermediary's main purpose is to accept requests for feedback, pass it to appropriate services, and then collect the responses to proxy it back to the CAD connector. Filtering (e.g., black- and whitelisting services or drop request containing certain types of information) and adding information (e.g. company accounts for special services) to the request are carried out in this component.
In order to know which services are available, the feedback intermediary can connect to a public service index to receive updated lists of available services. This proxy component can be configured through a web interface.

4.1.3 Registry (public service index)

The registry is a central address for service providers (i.e., manufacturers) to enlist the services they offer. From the perspective of the other stakeholders, it is the location to find needed services. As model extractors and visualizers are dynamically loadable in the CAD connector, the registry provides the necessary infrastructure to upload and download them. The registry is itself available through a user-friendly web interface where services can be reviewed and rated by users.

4.1.4 Services (internal, external)

Services take partial model data and compute feedback on it. External (public) service registers to the public service index (registry) and internal (private) services to a feedback intermediary. How services are implemented is left to the service providers. Two examples are given in the next section, where the proof of concept implementation is discussed. Like (Zissis et al., 2016) approach, services should exploit the benefits of cloud computing.
It should be noted, that while services and extractors/visualizers belong together, extractors and visualizers can be used by multiple different services. An extractor logic should not be executed multiple times if multiple services require the same information.

5 PROOF OF CONCEPT (POC) IMPLEMENTATION

In order to verify the distributed architecture for feasibility and to identify potential problems, a proof of concept implementation has been created using FreeCAD, an open source CAD system further described in (Falck et al., 2012). The POC does not implement all features of the architecture, nor was it optimised for computational performance.
The functionality of the CAD connector is exposed as a button in the CAD system. Clicking it triggers the extraction of two-dimensional contour (a sketch with a specific name) from the three-dimensional model and sends it to the intermediary. The intermediary passes the model further to a service. The service has preloaded a headless FreeCAD instance (i.e., without a graphical user interface) on service start up. On request (i.e., sketch data arrives at the service) two analyses are executed. The analyses are correspondent with the examples described in the use case section.
First, the sketch's wire (i.e., contour) is checked if it is closed. If closed, the software "walks" the wire's basic elements (edges) and calculates the angle between elements or checks the radius, when the element is an arc. If both are below configured thresholds, indicating a critical point, they are pushed to a list, which is then added to the response.
For the determination of a proper tool size, the sketch is rasterized to a (binary) black and white image and pixel by pixel compared to a similar image representing a slightly smaller version of available tool sizes. The overlapping pixels are counted. Is the count between zero and a threshold, a predefined maximum size contour is added to the response. The image manipulation is computed with ImageMagick Studio (2014).

On receiving the feedback, the CAD connector adds sketches with the feedback to the model. The selectablitily of the additions are set to false as they should not interfere with the engineer's further modelling operations. The engineer can optionally hide them in the model tree. Screenshots of the CAD system are depicted in Figure 3. Sharp corners are encircled.

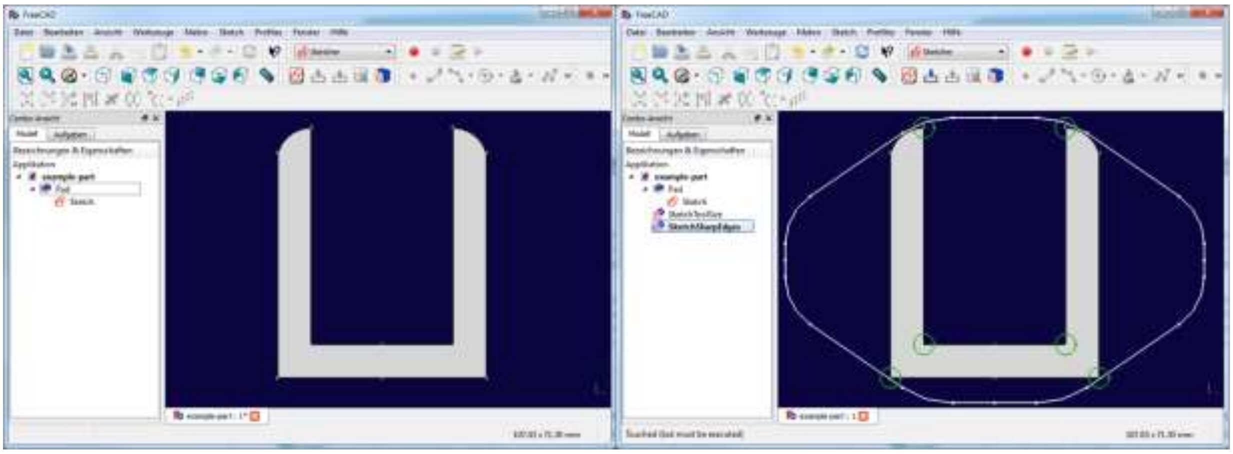

Figure 3. Additions to the CAD model by the POC implementation

6 RESULTS AND DISCUSSION

Goal of the presented approach is to create an architecture that enables fast, automated, distributed computed feedback for engineers and provide it in CAD systems. The given use case serves as a basis for the architecture validation and provides two examples for the proof of concept implementation, which is used to verify the architectures feasibility for simple feedback.

6.1 Architecture validation

Both types of stakeholders, the company and the supplier have a focus on protecting their intellectual property. The supplier achieves this by only proving the feedback not disclosing how it is computed. Partitioning services into extractors/visualizers running on the engineers' workstations and feedback computation "behind" a network allows the manufacturer to share the source code of the extractors/visualizers. This to establish trust without exposing know-how.
Companies have more options and fine-grained control over their intellectual property. They can choose which services they trust on company-wide level, by configuring the feedback intermediary to only allow certain trusted services or block distrusted ones for the engineers. They can also filter certain types of information at this level. As the feedback intermediary allows internal services to register, it can forward model information additionally to an internal service where, e.g., models are checked for internal standardisation violations. The engineer needs for information and the quality of this information is mainly left to the service providers. The sub second response time to not interrupt the engineers' flow of thought is addressed in the POC evaluation.

6.2 Proof of concept evaluation

Extracting information from the model, proxying it to the service through intermediary, calculating basic feedback and displaying it took in total 582 ms (sigma = 47 ms, n = 12). 347 ms (sigma = 24 ms) for the feedback computation and 231 ms (sigma = 33 ms) for visualising the feedback. All components were running on the same virtual machine on Ubuntu Linux with three cores and 8 GiB of memory. A debug build of the FreeCAD version 0.17 was used. Network performance depends on location and internet connection. As everything in the test setup was running on the same host network, latencies were neglected. The result shows that with delay of 417 ms added by the network, feedback would still be displayed within the sub second goal. The code was not performance optimized, i.e., the two analysations were not calculated in parallel, the file system was unnecessarily hit for rasterization and the algorithms were implemented naïvely. From the results, we conclude that it is feasible to compute basic manufacturability feedback as a service and integrate it within a second and therefore does not interrupt the engineer's flow of thought. Assuming that network roundtrip stays within the above bounds. The POC also indicates that naïvely implement services are not suitable for complex feedback.

7 SUMMARY AND OUTLOOK

A distributed software architecture to achieve real-time (sub second) feedback on manufacturability has been proposed and a proof of concept implementation (POC) has been created. The underlying assumption is, that fast feedback is immensely useful in all kinds engineering and that manufacturers are the ones who can provide it best. The architecture is evaluated against a described use case and the POC is used to test if feedback from a remote server can be provided to engineers within a second.

While the implemented examples are simple, future work will concentrate on examples that are more complex utilizing a wider spectrum of available technology. We would like to encourage the scientific community to discuss aspects in engineering design that can be potentially offloaded to computers, as it has the potential to free engineering resources and speed up development endeavours.

REFERENCES

Abramovici, M. and Herzog, O. (2016), *Engineering im Umfeld von Industrie 4.0: Einschätzungen und Handlungsbedarf*, Herbert Utz Verlag.

Boothroyd, G. (1994), "Product design for manufacture and assembly", *Computer-Aided Design*, Vol. 26 No. 7, pp. 505–520. https://doi.org/10.1016/0010-4485(94)90082-5

Card, S.K., Robertson, G.G. and Mackinlay, J.D. (1991), "The information visualizer, an information workspace", *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp. 181–186. https://doi.org/10.1145/108844.108874

Donati, L. and Tomesani, L. (2005), "The effect of die design on the production and seam weld quality of extruded aluminum profiles", *Journal of Materials Processing Technology*, Vol. 164–165, pp. 1025–1031. https://doi.org/10.1016/j.jmatprotec.2005.02.156

Falck, B., Falck, D. and Collette, B. (2012), Freecad [How-To], Packt Publishing Ltd.

Hochmuth, R. and others. (1997), "Konstruktionsbegleitende Analyse auf Fertigbarkeit von Toleranzfestlegungen auf Basis des Konstruktionssystems mfk", *DFX 1997: Proceedings of the 8th Symposium on Design for Manufacturing, Schnaittach/Erlangen*, Germany, 16.-17.10. 1997.

ImageMagick Studio, L.L.C. (2014), "ImageMagick: Convert, Edit, and Compose Images", Version, Vol. 6, pp. 9–1.

Junk, S. (2016), *Onshape - kurz und bündig: Praktischer Einstieg in Freeware-CAD und 3D-Druck*, Springer-Verlag. https://doi.org/10.1007/978-3-658-15380-9

Miller, R.B. (1968), "Response time in man-computer conversational transactions", *Proceedings of the December 9-11, 1968, Fall Joint Computer Conference*, Part I, ACM, pp. 267–277. https://doi.org/10.1145/1476589.1476628

Parametric Technology Corporation. (n.d.). "Creo Parametric Help Center", available at: http://support.ptc.com/help/creo/creo_pma/usascii/#page/sheetmetal%2Fsheetmetaldesign%2FManufacture_Prep%2FAbout_Reports.html%23 (accessed 16 December 2016).

Sapa Extrusion. (2014), *Konstruktionshandbuch Erfolg Mit Aluminiumprofilen*.

Wang, L., Shen, W., Xie, H., Neelamkavil, J. and Pardasani, A. (2002), "Collaborative conceptual design—state of the art and future trends", *Computer-Aided Design*, Vol. 34 No. 13, pp. 981–996. https://doi.org/10.1016/s0010-4485(01)00157-9

Wu, D., Rosen, D.W., Wang, L. and Schaefer, D. (2015), "Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation", *Computer-Aided Design*, Vol. 59, pp. 1–14. https://doi.org/10.1016/j.cad.2014.07.006

Wu, D., Terpenny, J. and Schaefer, D. (2016), "Digital design and manufacturing on the cloud: A review of software and services", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, pp. 1–15. https://doi.org/10.1017/s0890060416000305

Zissis, D., Lekkas, D., Azariadis, P., Papanikos, P. and Xidias, E. (2016), "Collaborative CAD/CAE as a cloud service", *International Journal of Systems Science: Operations & Logistics*, pp. 1–17. https://doi.org/10.1080/23302674.2016.1186237