

Methodischer Aufbau von Entwicklungsumgebungen nach dem Generative Parametric Design Approach

Kevin Herrmann^{1*}, Osman Altun¹, Philipp Wolniak¹, Iryna Mozgova¹, Roland Lachmayer¹

¹ Leibniz University Hanover

* Korrespondierender Autor:

Kevin Herrmann

Leibniz University Hannover - Institute for Product Development

An der Universität 1

30823 Garbsen

Phone : +49 (0)511 762 2552

Mail: herrmann@ipeg.uni-hannover.de

Abstract

Computer-aided development environments have increasingly found their way into product development over the last 20 years. Particularly in the development of different product variants for different requirements and target groups, they enable the partially or fully automated creation of component designs. The Generative Parametric Design Approach (GPDA) enables the generation of highly dynamic product models for complex components and assemblies by means of structural optimization and thus combines the advantages of parametric, knowledge-based and generative modeling methods. This paper generalizes the GPDA approach and demonstrates its applicability to functional assemblies and in the context of a manufacturing stage-based design model.

Keywords

Generative Parametric Design Approach, Knowledge Based Engineering, Manufacturing Stage-based Design Model, Functional Modeling

1. Einleitung

Mit ansteigender Produktkomplexität werden stärkere zielgruppenspezifische Produkte bei immer kürzer werdenden Entwicklungszyklen gefordert [1]. Dadurch entsteht eine hohe Variantenvielfalt, die den Produktentwicklungsaufwand erhöht. Im Produktentwicklungsprozess kommen hierbei Anpassungs- und Variantenkonstruktionen zum Einsatz, bei denen bekannte Lösungsprinzipien und Produktstrukturen beibehalten und die Komponenten an neue Anforderungen angepasst werden [2, 3]. Um den Entwicklungsaufwand zu reduzieren, werden vermehrt rechnerunterstützte Entwicklungswerkzeuge und virtuell Produktmodelle eingesetzt. Diese ermöglichen mittels Inferenzmechanismen eine teil- oder vollautomatisierte Produktentwicklung. Dabei besteht die Herausforderung, die komplexen Produktmodelle sowie Inferenzmechanismen robust aufzubauen, um Instabilitätsprobleme bei parametrischen und topologischen Änderungen am geometrischen Modell zu vermeiden [4, 5]. Es wurden bereits mehrere Methoden vorgeschlagen dieser Herausforderung zu begegnen [6, 7, 8, 9]. Es wird ein Modellierungsansatz benötigt, der die flexible Generierung von Produktvarianten ohne Fehler und mit minimalem Aufwand ermöglicht. Li et al. beschreiben mit dem Generative Parametric Design Approach (GPDA) ein Ansatz zur Generierung hochdynamischer Produktmodelle für Strukturbauteile mit Hilfe der Kombination der Vorteile parametrischer, wissensbasierter und generativer Modellierungsmethoden. Der GPDA verfolgt die Optimierung von Strukturbauteilen auf unterschiedliche Anforderungen, wie Lastfällen oder Fertigungsrestriktionen, und ermöglicht so das Erstellen von anforderungsspezifischen Produktvarianten [2]. Dieser Beitrag verallgemeinert den Ansatz des GPDA und thematisiert seine Übertragbarkeit auf neue Anwendungsbereiche, wie die der funktionalen Baugruppen und fertigungsstufenbasierten Gestaltungsmodellen.

2. Stand der Technik

2.1. Parametrische Modellierung

Bei der parametrischen Modellierung werden Eigenschaften und Abhängigkeiten von Produktmodellen mit Hilfe von Parametern beschrieben. Die Änderungen von Parametern führen zu direkten Modelländerungen. Dabei muss das CAD-System das Modell aktualisieren, seine Konsistenz prüfen und sicherstellen [10]. Parameter können Wertebereiche oder Beziehungen untereinander aufweisen, die nach arithmetischen, logischen oder geometrischen Regeln festgelegt werden. Diese Regeln hinterlegen wissensbasiert die Gestaltungslogik im Modell [11]. Eine Produktvariante entsteht im Nachhinein durch das Festlegen der Eingangsparameter. Die parametrische Modellierung bietet Potentiale bei der schnellen Generierung von Konstruktionslösungen in Bezug auf sich ändernde Anforderungen, da im Vergleich zu einer Variantenkonstruktion kein statisches Geometriemodell erzeugt wird [11, 12]. Die Flexibilität eines parametrischen Modells hängt von den vordefinierten Abhängigkeiten zwischen den Parametern sowie der gesetzten Randbedingungen ab. Sind diese einmal festgelegt, verliert das Modell an Flexibilität, da der Lösungsraum für die Bauteilgestalt nicht mehr erweitert werden kann. Daher muss die Konstruktionsaufgabe zur Definition der Variantendarstellung und Operationsreihnfolge im Vorfeld untersucht werden [2].

2.2. Wissensbasierte Konstruktionssysteme

Wissenbasierte Konstruktionssysteme sind als Sammelbegriff für computergestützte Problemlösungswerkzeuge zu verstehen [13]. Sie unterstützen den Entwurfsprozess durch vordefinierten Methoden, Algorithmen oder Ergebnissen und integrieren konstruktionsspezifische Aufgaben oder Arbeitsabläufe, um anhand von Anforderungen und Restriktionen eine (teil-)automatisierte Produktmodellerstellung zu erreichen [8, 14]. Dabei kommt meist die parametrische Modellierung als einzelnes Werkzeug zum Einsatz. Um eine Automatisierung des Entwurfsprozesses mittels wissensbasierten Systemen durchzuführen, müssen diese über die Fähigkeit besitzen Schlussfolgerungen aus bestehenden Problembeschreibungen zu ziehen und somit die Freiheitsgrade in einem variablen Produktmodell durch Setzen von Parameterwerten zu eliminieren [15, 16]. Wissensbasierte Systeme benötigen dafür einerseits Wissen über den Lösungsraum und andererseits Steuerungswissen, wie dieser Lösungsraum durch z. B. Parameter-Beschränkungen, Formeln oder Gestaltungsregeln erkundet wird [17]. Dies kann mittels regelbasierten, fallbasierten oder modellbasierten Schlussfolgerungstechniken erfolgen.

2.3. Generative Modellierung

Die generative Modellierung ist eine Methode, bei der eine Bauteilgestalt iterativ anhand von früheren Ausgangsdaten angepasst wird. Dabei werden Regeln aus vorhandenem Ingenieurwissen verwendet, um autonom eine hohe Anzahl von Lösungsalternativen zu generieren, die den spezifischen Anforderungen und Restriktionen entsprechen [2, 18]. Zu Beginn wird bei der generativen Modellierung die Entwurfsidee über einen Satz von algorithmischen Regeln formuliert. Anschließend erlauben generative Inferenzmechanismen Gestaltlösungen automatisch zu generieren und als Output zu visualisieren [19]. Bei den meisten Methoden, die die generative Modellierung aufgreifen, werden zur Erstellung von CAD-Modellen die Graphgrammatik und die räumliche Grammatik verwendet. Diese besitzen das Potential, strukturelle Komponenten zu erstellen, die direkt in einer CAD-Umgebung verwendet werden können [20]. Die Graphgrammatik besitzt ein Vokabular von Graphen und einen Satz an Regeln, um ein CAD-Modell durch ein Netzwerk von Graphen zu bilden. Der dafür verwendete Regelsatz verfügt über Mechanismen, die Strukturen innerhalb des Graphen-Netzwerks identifizieren und durch eine andere ersetzen können [2, 21]. Die räumliche Grammatik verwendet anstelle des Graphennetzwerks eine Geometriedarstellung und ein Vokabular, bestehend aus geometrischen Parametern [22]. Derzeit ist es noch eine Herausforderung, das Ingenieurwissen über Regeln zu formulieren. Grund dafür sind die Vielzahl an geometrischen Abhängigkeiten und die geometrische Komplexität vieler Bauteile [2]. CAD-Modelle lassen sich zwar durch Konstruktionsprachen repräsentieren, jedoch sind die Fragen, wie viele Regeln benötigt werden und wie diese Regeln und das Vokabular auf Basis des vorhandenen Wissens entwickelt werden können, noch offen [23].

3. Problemanalyse

Entwicklungsumgebungen zur automatisierten Produktentwicklung sind Forschungsthema vieler wissenschaftlicher Arbeiten [24, 25, 26]. Generative Entwicklungsumgebungen bieten Potentiale hinsichtlich der Möglichkeit der virtuellen und automatisierten Erprobung unterschiedlicher Varianten für gegebene Anforderungen. Die Integration von rechnerunterstützten Werkzeugen im Aufbau derartiger Entwicklungsumgebungen erfordert eine Abbildung von Analyse- und Synthesemethoden. Insbesondere die Abbildung von Synthesemethoden stellt eine Herausforderung in der Formalisierung dar. In diesem Beitrag wird ein generativer Ansatz (GPDA) für Entwicklungsumgebungen, mittels des Aufbaus von

parametrisch generativen CAD-Modellen und deren Kopplung an Simulationsumgebungen zur Realisierung des Analyseschritts sowie numerische Optimierungsmethoden zur Realisierung des Syntheseschritts diskutiert. Vorteile von GPDA-Modellen gegenüber rein parametrischen Modellen liegen darin eine automatisierte Vergleichbarkeit von konzeptionellen Änderungen durchzuführen sowie einzelne Gestaltelemente, gespeichert in Bibliotheken, für neue Nutzungsszenarien der Produkte wiederverwenden zu können. Gestaltelemente sind dabei selbst parametrisch aufgebaut und können baustrukturunabhängig entwickelt werden (vgl. Abschnitt 4). So ist die Erweiterbarkeit von Modellen durch Hinzufügen neuer Gestaltelemente in die Bibliothek ein weiterer Vorteil in Gegensatz zu rein parametrischen Modellen. Während der Aufwand für den Aufbau der Entwicklungsumgebung gegenüber manueller oder wissensbasierter Konstruktion zwar größer ist, ist der Entwicklungsaufwand für eine neue Variante im Vergleich geringer (siehe Bild 1). Beim manuellen Konstruieren werden einige wenige Varianten zur Lösungsfindung evaluiert. Beim wissensbasierten Konstruieren können mittels Schlussfolgerungsmechanismen eine höhere Anzahl an Varianten teilautomatisiert generiert werden. Die Kopplung von generativ parametrischen Modellen an numerische Optimierer führt zu einer weiteren Lösungsraumeinschränkung, bietet aber das Potential Parameter eines geometrischen Modells als Designvariablen in kleineren Schritten zu variieren und damit mehr Varianten und Konzepte zu evaluieren.

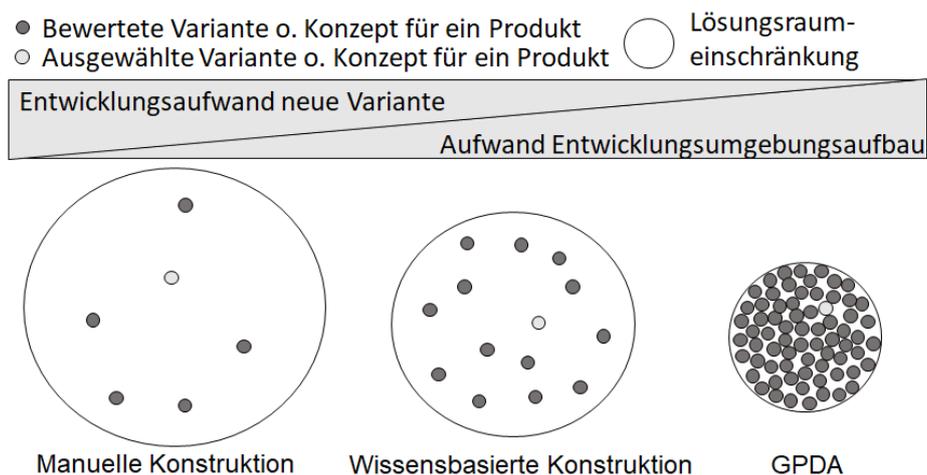


Bild 1: Lösungsraumexplorationen GPDA

Der vorliegende Artikel handelt von Erstellung von generativen Entwicklungsumgebungen nach dem GPDA. Folgende Fragestellungen werden zur Entwicklung eines Rahmenwerks für GPDA-Entwicklungsumgebungen über den Anwendungsbereich von Strukturbauteilen hinaus diskutiert:

- Was sind die wesentlichen Methoden und Werkzeuge zur Entwicklung einer Entwicklungsumgebung nach dem GPDA?
- Wie kann der GPDA über den Anwendungsbereich von Strukturbauteilen hinaus erweitert und generalisiert werden?

4. Idee und Aufbau des Generative Parametric Design Approach

Der Grundgedanke des GPDA ist es, die globalen Abhängigkeiten in CAD-Modellen zu reduzieren, da diese die Flexibilität einschränken. Dies wird dadurch erreicht, dass die Modellstruktur von der Produktstruktur eines Bauteils getrennt wird. Das bedeutet, dass einzelne Komponenten nicht ein einzelnes CAD-Modell sein müssen, sondern eine Kombination aus mehreren sein können. Bei Baugruppen sind die Schnittstellen der einzelnen Komponenten nicht mehr dort, wo sie sich in einem physikalisch existierenden Produkt befinden, sondern dort, wo die kleinsten Abhängigkeiten für das Gesamtmodell entstehen [2]. Die Bereiche zwischen den Schnittstellen werden als Gestaltungszonen bezeichnet. Diese dienen als Platzhalter für Gestaltelemente, die die Lösungsvarianten der Gestaltungszone darstellen und in Bibliotheken gespeichert werden. Entscheidend in der Gestaltungszone ist die Definition der Schnittstellen, wobei eine Referenzgeometrie in Form von Ebenen oder gekrümmten Flächen mithilfe eines Skeletts innerhalb des CAD-Modells festgelegt wird (siehe Bild 2). Die Schnittstellen verbinden zwei Gestaltungszonen miteinander und gewährleisten einen kontinuierlichen Geometrieübergang. Neben den Schnittstellen werden Abmaße und Anordnung der Gestaltungszonen über globale Parameter festgelegt. Das Skelett stellt somit die höchste Ebene in der Abhängigkeitskette dar, weshalb seine geometrische Stabilität gewährleistet sein muss. Innerhalb einer Gestaltungszone lassen sich die Gestaltelemente austauschen. Dadurch kann der GPDA im Vergleich zu einer rein parametrischen Modellierung einen größeren Lösungsraum abdecken, da neben Skalierungsanpassungen auch topologische Änderungen an der Gestalt des Produkts vorgenommen werden können.

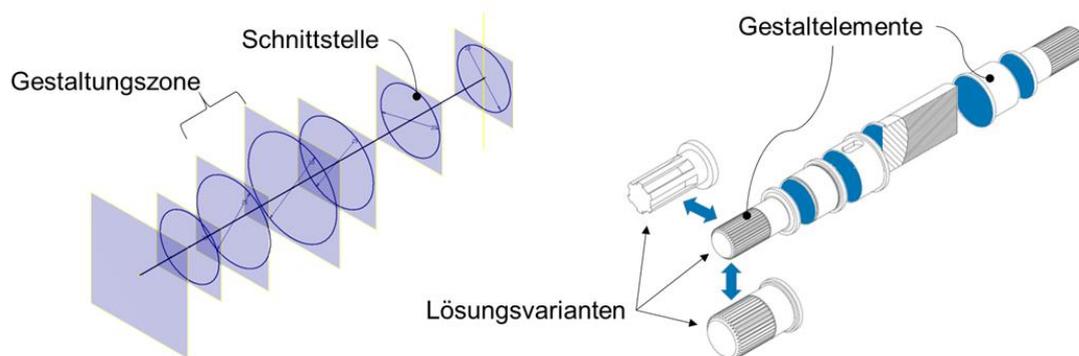


Bild 2: Schematischer Aufbau eines GPDA-Modells

Damit durch diese Modellstruktur automatisch eine optimierte Geometrie generiert werden kann, wird eine computergestützte Entwicklungsumgebung benötigt. Diese passt die Bauteilgestalt durch abwechselnde Synthese- und Analyseschritte iterativ an, z. B. mit einem evolutionären Algorithmus, bis die Anforderungen erfüllt sind [2]. In [2] wird die Entwicklungsumgebung für den GPDA als Optimierungswerkzeug für Strukturkomponenten folgendermaßen umgesetzt. Das zentrale Element ist das Optimierungsmodul, welches auf VB.NET basiert und die gesamte Entwicklungsumgebung steuert. Im Optimierungsmodul definiert der Anwender die Zielfunktion für die Optimierung. Anschließend konfiguriert der Algorithmus das GPDA Modell im CAD-System. Zu Beginn werden verschiedene Varianten mit unterschiedlichen Parameterkonfigurationen erstellt. Die als GPDA Modell vorliegenden Varianten werden dann mittels des STEP-Formats an das FE-System zur Analyse weitergegeben. Alle Randbedingungen und Berechnungsinformationen werden über eine XML-Datei von dem Optimierungsmodul für das FE-System bereitgestellt. Nach der Analyse werden die Ergebnisse der Simulation von dem Optimierungsmodul interpretiert und geprüft,

ob eine der Varianten der ersten Generation die Anforderungen erfüllt. Werden die Anforderungen nicht erfüllt, beginnt das Optimierungsmodul die nächste Iterationsschleife.

5. Erweiterung des GPDA

Im vorherigen Kapitel wurde der GPDA als Entwicklungsumgebung für Strukturbauteile beschrieben. Im Folgenden wird aufgezeigt, wie der Ansatz des GPDA auf andere Anwendungsgebiete übertragen werden kann. Die Entwicklungsumgebung des GPDA lässt sich auf drei wesentliche Komponenten abstrahieren (siehe Bild 4). Im Vordergrund steht ein generativ parametrisches Geometriemodell. Zur Realisierung des parametrischen Aspektes und der Befähigung des Modells robuste Lösungen zu erstellen, sind relative Parametrisierungen der Gestaltungszonen in sich und der Gestaltelemente sowohl in sich als auch zu den Gestaltungszonen notwendig. Kontinuierliche Gestaltelemente sind Elemente, die lediglich geometrisch skaliert werden können. Diskrete Gestaltelemente beschreiben Standardteile bzw. Einkaufsteile, die in Bibliotheken gespeichert und in der GPDA-Entwicklungsumgebung zur Lösungsfindung iterativ ausgetauscht werden. Ein Analysewerkzeug dient dazu Funktionalitäten, Verhalten, Eigenschaften usw. der Produkte bei gegebenen Restriktionen und Nutzerszenarien zu simulieren. In einem Synthesewerkzeug wird die Entscheidung getroffen, welche Variante das Optimum für gegebene Anforderungen und/oder Zielfunktionen ist und was als nächstes im geometrischen Modell dafür anzupassen ist. Mögliche Elemente eines Synthesewerkzeugs und Analysewerkzeugs sind in Bild 3 dargestellt. Informationen über den Modellzustand der aktuellen Variante werden je nach Anwendungsfall entweder über das Synthesewerkzeug an das Analysewerkzeug gegeben oder das Geometriemodell ist Bestandteil des Analysewerkzeugs, wie in kommerziellen FE-Werkzeugen.

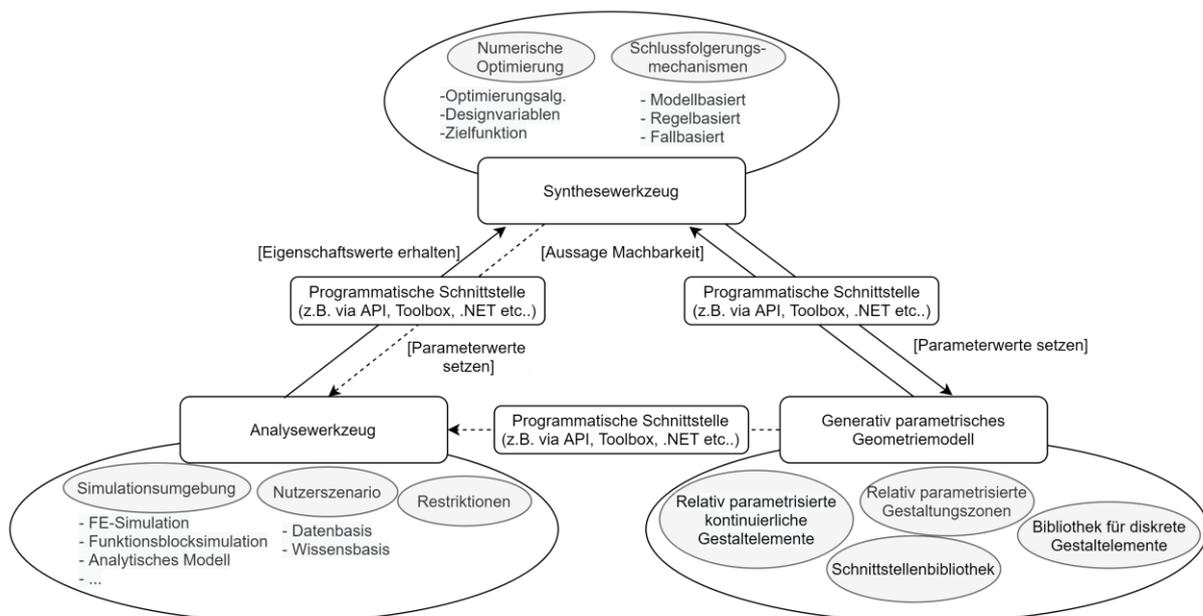


Bild 3: Rahmenwerk GPDA-Entwicklungsumgebungen

5.1. Übertragung auf eine funktionale Baugruppe - Beispiel Kaffeemaschine

Zur Übertragung des GPDA auf funktionale Baugruppen sind die Systemkomponenten aus Bild 3 zu adaptieren. Mit funktionalen Baugruppen werden hier technische Systeme

bezeichnet, bei denen der Stoff- und Energiefluss im Vordergrund steht. Informationsflüsse werden in der GPDA-Entwicklungsumgebung vernachlässigt. Aufbauend auf der Funktionsstruktur eines Produkts werden Gestaltungszonen identifiziert, die Grenzen des Lösungsraumes definiert, sowie eine relative Parametrisierung der Gestaltungszonen durchgeführt. Die Gestaltungszonen für die beispielhaft aufgeführte Kaffeemaschine sind in Bild 4a dargestellt. Innerhalb jeder Gestaltungszone befinden sich mögliche Teillösungen für die Funktionen. Beispielhaft ist hier die Gestaltungszone 2 „Schublade“ aufgezeigt. Die Schublade ist ein kontinuierliches Gestaltelement und wird sowohl in sich relativ parametrisiert als auch bezogen auf die zugehörige Gestaltungszone. Hier befinden sich auch diskrete Gestaltelemente (siehe Bild 4b) wie Pumpen und Heizelemente unterschiedlicher Varianten und Leistungsklassen. Wie bei Wolniak et al. beschrieben wird eine Funktionsblocksimulation in Matlab/Simulink aufgebaut [27]. Basierend auf den zielfunktionsbeeinflussenden geometrischen Größen werden Varianten der Kaffeemaschine analysiert und mittels der Kopplung an einen Optimierungsalgorithmus die Synthese-Operation durchgeführt. Im Fall der Kaffeemaschine können als Zielfunktionen der Wasserverbrauch und die Zeit bis zur Kaffeeausgabe verwendet werden. Als Designvariablen kommen die Parameter der Gestaltungszonen sowie die numerische Auswahl von diskreten Gestaltelementen zum Einsatz. Da das geometrische Modell durchgängig relativ parametrisiert aufgebaut wird, werden alle anderen Größen mit variiert. Regelbasiert sind im geometrischen Modell Varianten eines diskreten Gestaltelementes abgespeichert. Je nach Zustand einer definierten Variablen in der Optimierungsumgebung werden Varianten im geometrischen Modell erzeugt. Nach demselben Prinzip wird mit Schnittstellen zwischen Gestaltungszonen verfahren. Ändert sich ein Gestaltelement, so muss sich die Schnittstelle zu benachbarten Gestaltungszonen adaptieren. Bei diskreten Gestaltelementen erfolgt dies regelbasiert und bei kontinuierlichen über die Parametrisierung.

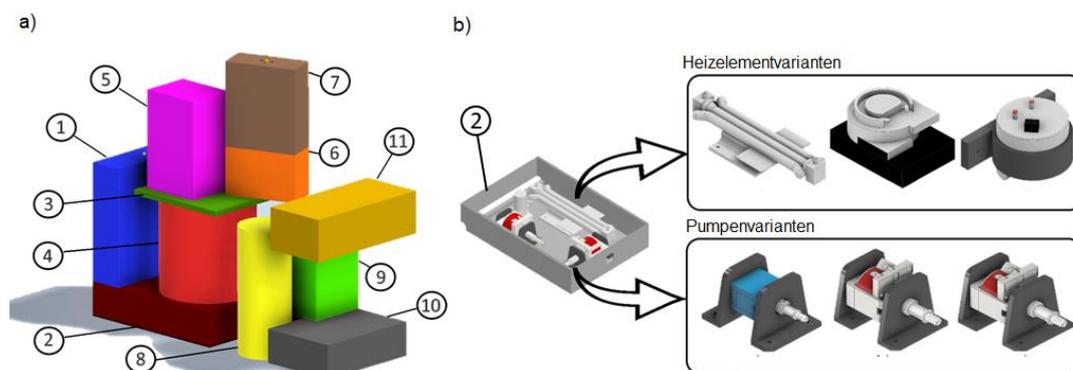


Bild 4: Ausschnitt des GPDA-Modells der Kaffeemaschine

5.2. Übertragung auf ein fertigungsstufenbasiertes Gestaltungsmodell - Beispiel Welle

Der Ansatz eines fertigungsstufenbasierten Gestaltungsmodells wird in [28] und [29] vorgestellt. Er zielt darauf ab, die Herstellbarkeit von Bauteilentwürfen über das Ableiten der zur Herstellung notwendigen Fertigungsstufen zu bewerten. Die inverse Betrachtung der Fertigungskette ermöglicht neben der Berücksichtigung von Fertigungsrestriktionen einzelner Fertigungsverfahren auch die Einbeziehung von prozessübergreifenden Fertigungsrestriktionen [29]. Das Ableiten der Fertigungsstufen wird über mehrere Analyse- und Synthese-Operationen umgesetzt. Die Analyse erfolgt mittels Fertigungssimulationen, z. B. Umformsimulationen und die Synthese übernimmt ein mathematischer Optimierer. Im

Analyse-Synthese-Prozess passt der Optimierer die Gestalt der vorgelagerten Fertigungsstufe so lange an, bis sie die Herstellung der nachgelagerten Fertigungsstufe mit den zur Verfügung stehenden Fertigungsressourcen und Prozessen ermöglicht. Dafür werden die Fertigungsstufen als GPDA Modelle erstellt. Diese enthalten durch die Modellstruktur eine sehr hohe Flexibilität, da der Optimierer beim Ableiten der Fertigungsstufen nicht nur parametrische sondern auch topologische Änderungen vornehmen kann, indem andere Gestaltelemente in das Skelett des Modells gehängt werden. So lässt sich ein größerer Lösungsraum für die Fertigungsstufen darstellen. Dabei nimmt die Anzahl an möglichen Gestaltelementen und deren Detailierungsgrad in Richtung der finalen Bauteilgeometrie zu. Folglich lassen sich viele Gestaltelemente aus den späteren Fertigungsstufen auf weniger Gestaltelemente vorheriger Fertigungsstufen beziehen bzw. können zu diesen reduziert werden [29]. Bild 5 zeigt das GPDA Modell der Fertigungsstufen am Beispiel einer multimateriellen Welle mit einer Auswahl an Gestaltelementen. Der Herstellungsprozess wird im Rahmen des Sonderforschungsbereiches 1153 erforscht und ist hier vereinfacht, bestehend aus einem Reibschweiß-, Fließpressprozess und einer finalen spanenden Bearbeitung, dargestellt.

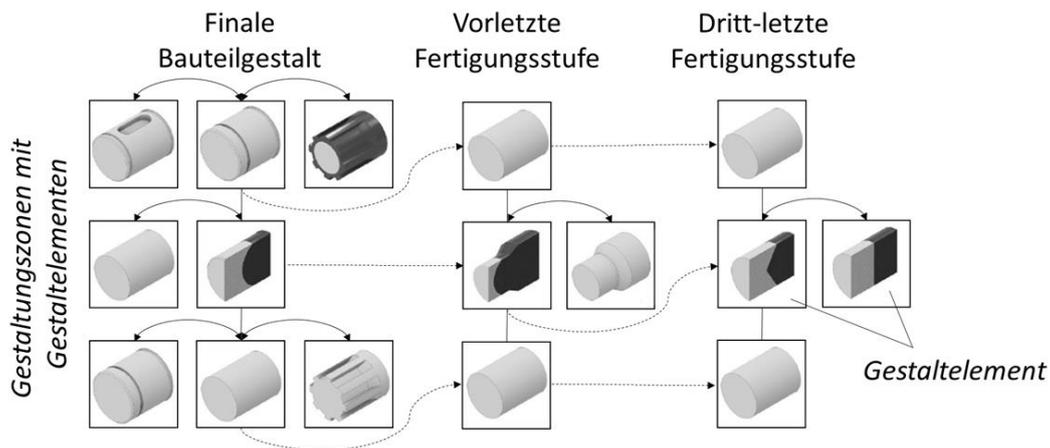


Bild 5: GPDA Modell der Fertigungsstufen am Beispiel einer multimateriellen Welle [nach 29]

6. Zusammenfassung und Ausblick

Nach einer Motivation von generativen Ansätzen in der Produktentwicklung untersucht diese Arbeit den GPDA für Strukturbauteile und dessen Übertragbarkeit auf andere Anwendungsfelder. Ausgehend von der Idee des GPDA ist ein Rahmenwerk für den Aufbau von Entwicklungsumgebungen nach dem GPDA über den Anwendungsfall der Strukturbauteile hinaus entwickelt worden. Eine GPDA Entwicklungsumgebung gliedert sich in drei Systemkomponenten, in ein Synthesewerkzeug, ein Analysewerkzeug und einem generativen parametrischen Geometriemodell. Die Verknüpfungen zwischen einem Geometriemodell, einer Optimierungs- und Simulationsumgebung ermöglicht die automatische Suche innerhalb eines eingeschränkten Lösungsraums nach optimalen Varianten. Zur weiteren Untersuchung der Übertragbarkeit des GPDA auf Anwendungsfälle, wie funktionale Baugruppen und fertigungsstufenbasierte Gestaltungsmodelle, werden die im Rahmen dieser Arbeit diskutierten Ansätze mit weiteren Beispielen erweitert und systematisiert. Insbesondere die Modellierung von generischen Schnittstellen stellt eine Herausforderung für zukünftige Forschungen mit dem GPDA. Darüber hinaus Bedarf es an weiteren spezifischen Methoden zur Unterstützung des Aufbaus von Entwicklungsumgebungen nach dem GPDA.

Danksagung

Die Ergebnisse dieser Publikation sind im Rahmen des Sonderforschungsbereichs 1153 „Prozesskette zur Herstellung hybrider Hochleistungsbauteile durch Tailored Forming“ innerhalb des Teilprojekts C2 entstanden. Die Autoren danken der Deutschen Forschungsgemeinschaft (DFG) für die finanzielle und organisatorische Unterstützung des Projektes (Projektnummer: 252662854).

Literaturverzeichnis

- [1] Eigner, Martin; Roubanov, Daniil; Zafirov, Radoslav: Modellbasierte virtuelle Produktentwicklung. 1. Aufl. Berlin, Heidelberg: Springer-Verlag, 2014.
- [2] Li, Habing; Brockmüller, Tim; Gembarski, Paul Christoph; Lachmayer, Roland: An Investigation of a Generative Parametric Design Approach for a Robust Solution Development. International Design Conference – design 2020. 2020. <https://doi.org/10.1017/dsd.2020.273>
- [3] Pahl, Gerhard et al.: Engineering design: a systematic approach. London: Springer-Verlag, 2007.
- [4] Li, Habing; Lachmayer, Roland: Automated Exploration of Design Solution Space Applying the Generative Design Approach”. Proceedings of the Design Society: International Conference on Engineering Design, Cambridge University Press, Vol. 1 (2019), No. 1, pp. 1085-1094. <https://doi.org/10.1017/dsi.2019.114>
- [5] Amadori, Kristian et al.: Flexible and robust CAD models for design automation. In Advanced Engineering Informatics, Vol. 26 (2012) No. 1, pp. 180-195. <https://doi.org/10.1016/j.aei.2012.01.004>
- [6] Kim, Byung; Han, Soonhung: Integration of history-based parametric translators using the automation APIs. In: International Journal of Product Lifecycle Management, Vol. 2 (2007), No. 1, pp. 18-29. <https://doi.org/10.1504/IJPLM.2007.012872>
- [7] Yin, Chenggang; Ma, Yongsheng: Parametric feature constraint modeling and mapping in product development. In: Advanced Engineering Informatics, Vol. 26 (2012), No. 3, pp. 539-552. <https://doi.org/10.1016/j.aei.2012.02.010>
- [8] Hirz, Mario et al.: Integrated computer-aided design in automotive development. Graz: Springer-Verlag, 2013.
- [9] Boyle, Iain; Rong, Yiming; Brown, David: A review and analysis of current computer aided fixture design approaches. In Robotics and Computer-Integrated Manufacturing, Vol. 27 (2011), No. 1, pp. 1-12. <https://doi.org/10.1016/j.rcim.2010.05.008>
- [10] VDI 2209:2009, 3D-Produktmodellierung Technische und organisatorische Voraussetzungen, Verfahren, Werkzeuge und Anwendungen, Wirtschaftlicher Einsatz in der Praxis.
- [11] Vajna, Sandor et al.: CAX für Ingenieure - Eine praxisbezogene Einführung. 3. Aufl. Berlin: Springer-Verlag, 2018.
- [12] Hoffmann, Christoph M.: Constraint-based computer-aided design. In: Journal of Computing and Information Science in Engineering, Vol. 5 (2005), No. 3, pp. 182-187. <https://doi.org/10.1115/1.1979508>
- [13] Milton, Nick. R.: Knowledge technologies. Mailand: Monza Polimetrica sas, 2008.
- [14] Verhagen, Wim J.C., Bermell-Garcia, Pablo, van Dijk, Reinier E.C., and Richard Curran: A critical review of Knowledge-Based Engineering: An identification of research challenges. In: Advanced Engineering Informatics 26(1): 5-15, 2012. <https://doi.org/10.1016/j.aei.2011.06.004>
- [15] Gembarski, Paul Christoph: On the conception of a Multi-Agent Analysis and Optimization Tool for Mechanical Engineering Parts. In: G. Jezic, J. Chen-Burger, M. Kusek, R. Sperka, R. Howlett, and L. Jain (Hrsg.): Agents and Multi-Agent Systems: Technologies and Applications, Smart Innovation, Systems and Technologies, vol. 186. (2020), pp. 93-102, 2020. https://doi.org/10.1007/978-981-15-5764-4_9
- [16] Sabin, Daniel; Rainer Weigel: Product configuration frameworks - a survey. In: IEEE intelligent systems 13(4): 42-49, 1998.
- [17] Gembarski, Paul Christoph et al.: Knowledge Repositories for Knowledge-Based-Engineering Applications. In: Marjanović, Dorian, Štorga, Mario, Pavković, Neven, Bojčetić, Nenad and Stanko Škec (eds) Proceedings of the DESIGN 2016 14th International Design Conference, Glasgow, The Design Society, 2016.
- [18] Cui, Jia; Tang, Ming Xi: Towards generative systems for supporting product design. In: International Journal of Design Engineering, Vol. 7 (2017) No. 1, pp. 1-16. <https://doi.org/10.1504/IJDE.2017.085639>
- [19] Krish, Sivam: A practical generative design method. In: Computer Aided Design, Vol. 43 (2011) No. 1, pp. 88-100. <https://doi.org/10.1016/j.cad.2010.09.009>
- [20] Chakrabarti, Amaresh et al.: Computer-based design synthesis research: an overview. In Journal of Computing and Information Science in Engineering, Vol. 11 (2011) No. 2, p. 021003. <https://doi.org/10.1115/1.3593409>
- [21] Helms, Bergen: Object-oriented graph grammars for computational design synthesis. München: TU München, Dissertation, 2013.

-
- [22] Hoisl, Frank Rainer: Visual, interactive 3D spatial grammars in CAD for computational design synthesis. München: TU München, Dissertation, 2012.
- [23] Gips, James: Computer implementation of shape grammars. NSF/MIT workshop on shape computation, Massachusetts Institute of Technology, Cambridge, MA, 1999.
- [24] Holder, Kevin; Rudolph, Stephan; Stetter, Ralf; et al: Automated requirements-driven design synthesis of gearboxes with graph-based design languages using state of the art tools. In: Forsch Ingenieurwes 83, 655–668, 2019. <https://doi.org/10.1007/s10010-019-00322-z>
- [25] Bursać, Nikola: Model Based Systems Engineering as a support for the Modular Design in the Context of the Early Stages of Product Generation Engineering. Karlsruhe, Dissertation, 2016.
- [26] Schleich, Benjamin; Wartzack, Sandro: A generic approach to sensitivity analysis in geometric variations management. DS 80-4 Proceedings of the 20th International Conference on Engineering Design, ICED 15 Vol 4 (2015): Design for X, Design to X, Milan, Italy, 27-30.07.
- [27] Wolniak, Philipp.; Klooschreiber, Daniel; Sauthoff, Bastian; Lachmayer, Roland: Integrating Architectural Design Changes in Computer-Aided Design Optimization. Internation Conference on Mass Customization and Personalization - Community of Europe, MCP-CE, 2020.
- [28] Brockmüller, Tim; Siqueira, Renan; Mozgova, Iryna, Lachmayer, Roland: Rechnergestützte Entwicklungsumgebung zur Konstruktion von Tailored-Forming-Bauteilen. DS 98: Proceedings of the 30th Symposium Design for X, 18 – 19 September 2019, Jesteburg, Deutschland, 2019 <https://doi.org/10.35199/dfx2019.17>
- [29] Herrmann, Kevin; Brockmüller, Tim; Gembarski, Paul Christoph; Mozgova, Iryna; Lachmayer, Roland: Fertigungsstufen-basiertes Gestaltungsmodell für mechanische Bauteile. Proceedings of the of the Stuttgarter Symposium für Produktentwicklung SSP 2021, S. 247 – 258, Stuttgart, 2021. <http://dx.doi.org/10.18419/opus-11478>