

Modellierung und Analyse funktionaler Varianz komplexer technischer Systeme

Modeling and analysis of functional variance of complex technical systems

Bastian Menninger^{1,*}, Dominik Wiechel², Sascha Rackow³, Gregor Höpfner¹, Christian Oleff², Joerg Berroth¹, Iris Gräßler², Georg Jacobs¹

¹ Institute for Machine Elements and Systems Engineering, RWTH Aachen University

² Heinz Nixdorf Institute, Chair for Product Creation, Paderborn University

³ CLAAS Industrietechnik GmbH

* Korrespondierender Autor:

Bastian Menninger

Rheinisch-Westfälische Technische Hochschule Aachen

Institut für Maschinenelemente und Systementwicklung

Eilfschronsteinstraße 18, 52062 Aachen

☎ +49 241 80 95639

✉ bastian.menninger@imse.rwth-aachen.de

Abstract

Complexity of variances in systems leads to challenges in structuring and in making development decisions. Models can collect system information to make them available and transparent. Currently, no approach to model the variance of systems along relevant engineering artifacts exists.

This research aims to develop a consistent variant modeling approach: In a literature review (1), relevant approaches in model-based variant management are identified to derive requirements for variant modelling (2). Based on the requirements, approaches are evaluated in a matrix (3). Since no approach satisfies all requirements, one approach is extended (4). For evaluation purpose (5), applicability of the approach is demonstrated and assessed by experts. The result is a modeling method for variant modeling.

Keywords

Model-Based Systems Engineering, Variants, function-oriented Complexity

1. Motivation

Die Beherrschung externer und interner Komplexität gilt als zentrale Aufgabe der Systementwicklung, um die Wettbewerbsfähigkeit eines Unternehmens sicherzustellen [1]. Komplexität drückt sich unter anderem in einer hohen Varianz an Entwicklungsartefakten (Anforderungen, Funktionen, Lösungsprinzipien, physische Elemente und Abhängigkeiten [2]) aus. Probleme bei der Beherrschung von Varianz wirken sich durch die Zunahme der Vielfalt der Teile, der Entwicklungsaufwände und einen ineffizienten Umgang mit technischen Änderungen negativ auf die Wettbewerbsfähigkeit des Unternehmens aus [4]. Daher werden bereits in der Produktentwicklung geeignete Ansätze benötigt, um die Varianz zu beherrschen und somit die Komplexität zu bewältigen. Dabei unterstützen Ansätze der modellbasierten Produktentwicklung das Variantenmanagement, indem Struktur- und Verhaltensinformationen und die Merkmale von Varianten sowie deren Abhängigkeiten und Wechselwirkungen systematisch in Systemmodellen abgebildet werden [5]. Durch eine explizite und durchgängige Verknüpfung aller Modellelemente können Inkonsistenzen in Entwicklungsdaten reduziert, Varianten gezielt konfiguriert und Ingenieure bei Entwicklungsentscheidungen der Systemarchitektur unterstützt [6, 7] werden. Des Weiteren können Auswirkungen technischer Änderungen auf die Varianten [8] abgeschätzt, redundante Informationen identifiziert [6] und Systemarchitekturen für folgende Systemgenerationen wiederverwendet werden [9, 10]. Funktionsorientierung bekommt eine wachsende Relevanz in der Entwicklung cyberphysischer Systeme und wird in der Betrachtung von Varianz nicht ausreichend berücksichtigt. Aktuell existiert keine Methode, um Ingenieure bei der frühzeitigen Abbildung von Varianz von Funktionen und Lösungsprinzipien zu unterstützen [11]. Aktuelle Ansätze des modellbasierten Variantenmanagements bilden die Varianz vor allem auf physischer Ebene ab, basierend auf Stücklisten für explizite Varianten [4, 12]. Im Vergleich zur Abbildung der Varianz auf physischer Ebene gilt die frühzeitige Abbildung von Varianz innerhalb der funktionalen und logischen Ebene der Systemarchitektur als Herausforderung [13]. Die Vernachlässigung der Varianz auf funktionaler und logischer Ebene führt zu einer unvollständigen Informationsgrundlage der Entwickler, sodass keine optimalen Entscheidungen getroffen werden können (Abbildung 1).

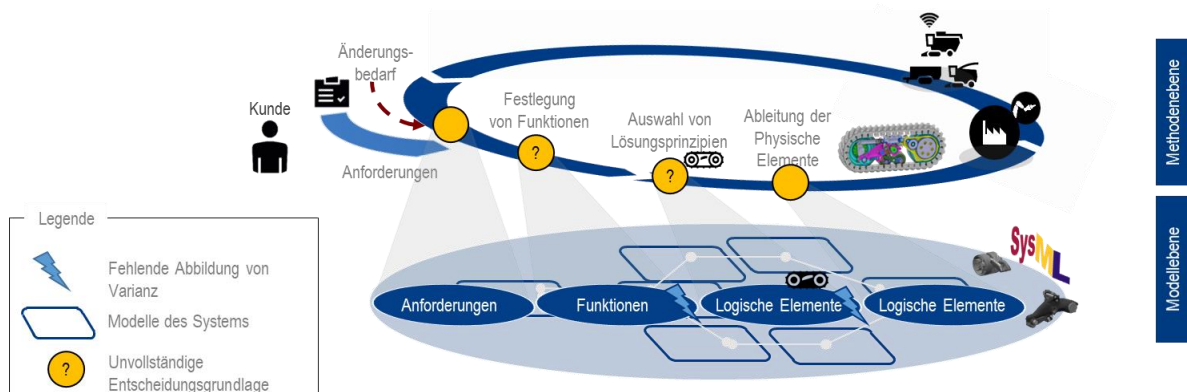


Abbildung 1: Herausforderungen in der modellbasierten Abbildung von Varianz

2. Stand der Technik

Varianten- und Konfigurationsmanagement

Das Variantenmanagement unterstützt Entwickler bei der Entwicklung, Gestaltung und Strukturierung von variantenreichen Systemen im Unternehmen [1]. Gründe für die Vielzahl der Varianten sind unter anderem individuelle Kundenbedürfnisse sowie branchen- oder länderspezifische Anforderungen [4]. Eine Herausforderung im Umgang mit Varianten ist die entstehende Inkonsistenz, wenn Daten unterschiedlicher Module nicht miteinander verknüpft

sind und nicht konsistent nachverfolgt und eingepflegt werden können [19]. Ansätze des modellbasierten Variantenmanagements eignen sich, um die Interdependenzen zwischen den betrachteten Entwicklungsartefakten abzubilden [14, 15].

Ansätze des modellbasierten Variantenmanagements

Die Ansätze des modellbasierten Variantenmanagements beruhen auf Modellierungssprachen und Notationen, die für die Abbildung von Varianten entwickelt oder angepasst wurden [16]. Kang et al. [17] entwickelte den Ansatz des Feature-Oriented Domain Analysis (FODA), bestehend aus einer Notation, mit der sich Funktionen (Features) sowie die dazugehörigen Variationen beschreiben und analysieren lassen. Basierend auf dem FODA Ansatz wurden weitere Ansätze wie die Feature-Oriented Reuse Method (FORM) hergeleitet. Pohl et al. [18] definieren über das Orthogonal Variability Model (OVM) ein Konzept zur Abbildung von Varianteninformationen aus dem Softwarebereich innerhalb eines separaten orthogonalen Modells. Beide Ansätze sind durch die disziplinspezifische Fokussierung auf Software beschränkt. Eine existierende disziplinübergreifende Modellierungssprache ist die Common Variability Language (CVL) der Object Management Group, welche die spezifische Abbildung von Varianten ermöglicht, ohne aber die Entwickler bei der Ausarbeitung der Varianten zu unterstützen [19]. Zur Abbildung komplexer technischer Systeme eignet sich die Modellierungssprache SysML, welche durch angepasste Modellelemente in einem SysML-Profil erweitert werden und so Varianz abbilden kann. Ein Beispiel ist das SysML-Profil Model-Based Product Line Engineering (MBPLE), welches genutzt werden kann, um verhaltensorientierte Varianten zu beschreiben und Konfigurationen abzuleiten [20]. In einem weiteren Ansatz definiert Weikiens [21] ein SysML Profil (VAMOS), welches Stereotypen enthält, die für die modellbasierte Beschreibung von Varianten genutzt werden können. Bei beiden SysML-Profilen wird die Varianz erst auf physischer Ebene modelliert. Die Varianz der vorgelagerten Funktionen und logischen Systemelemente wird vernachlässigt, obwohl diese einen weitreichenden Einfluss auf Kosten oder Machbarkeit haben [24]. Zusätzlich existieren alleinstehende Softwareprodukte wie Pure::variants [13], welche , mit Systemmodellierungs-Werkzeugen kombiniert angewendet werden können, aber zu zusätzlichen Lizenzkosten führen.

motego

Das motego Profil ist ein SysML-Profil zur modellbasierten Systementwicklung des Instituts für Maschinenelemente und Systementwicklung an der RWTH Aachen University [38] und verfügt bislang nicht über die Möglichkeit der Variantenmodellierung. Das Profil ermöglicht die durchgängige Modellierung von Systemen anhand der vier Artefakte Anforderungen, Funktionen, Lösungen und physische Elemente. Zur differenzierten Betrachtung der Artefakte wurden Stereotypen in dem Profil definiert, die eine detaillierte Modellierung von Systeminformationen in einer hohen Granularität zulassen. Zudem definiert die Methode eindeutige Zusammenhänge, die zwischen den Informationen modelliert werden [39, 40].

Modellierungsmuster im Variantenmanagement

Die Ansätze des modellbasierten Variantenmanagements lassen sich anhand von Modellierungsmustern unterscheiden, welche die Struktur der Modelle beschreiben. Durch das 150% Modellierungsmuster werden alle Modellelemente von den existierenden Variaten eines Systems innerhalb eines Systemmodells abgebildet. Durch das Modellierungsmuster wird die Anzahl von abzubildenden Entwicklungsartefakten reduziert und Konsistenz innerhalb des Modells sichergestellt [22]. Nach dem Kapselungsmuster werden alle Varianten in einem separaten Modell dargestellt. Das 150%-Modellierungsmuster eignet sich um Varianten mit einem hohen Grad an Gemeinsamkeit abzubilden, während heterogene Varianten mit dem Kapselungsmuster modelliert werden [20].

3. Forschungsproblem und Forschungsziel

Die explizite und durchgängige Modellierung der Varianz aller Entwicklungsartefakte wird in aktuellen Ansätzen der modellbasierten Systementwicklung sowie in Modellierungssprachen wie der Systems Modeling Language (SysML) nicht hinreichend unterstützt [23]. Zum einen fehlt eine dokumentierte und systematische Beschreibung der notwendigen Vorgehensweise, um eine durchgängige Modellierung entlang der unterschiedlichen Entwicklungsartefakte und deren Varianten in SysML zu ermöglichen. Zum anderen fehlen aufgrund der unvollständigen Betrachtung in den SysML-Ansätzen die notwendigen Modellelemente und Abhängigkeiten, um Varianz frühzeitig und durchgängig in einem Profil abbilden zu können und so Entwickler bei Entscheidungen zu unterstützen. Folgende Forschungsfragen (FF) ergeben sich aus dieser Problemstellung:

FF 1: Welche Vorgehensweise eignet sich für die durchgängige modellbasierte Abbildung von Varianz?

FF 2: Welche Modellelemente und Abhängigkeiten werden für eine durchgängige modellbasierte Variantenabbildung benötigt?

FF 3: Um welche Modellelementtypen und Abhängigkeitstypen muss ein Profil eines Modellierungsansatzes für eine durchgängige modellbasierte Variantenabbildung erweitert werden?

4. Verwendete Methoden bzw. Vorgehensweisen

Zur Beantwortung der Forschungsfragen wird ein sechsschrittiges Vorgehen gewählt: Für die Identifikation existierender Ansätze wird eine systematische Literaturrecherche in wissenschaftlichen Datenbanken durchgeführt (1). Die Ansätze werden in Hinblick auf eine durchgängige Modellierung von Varianten analysiert, um darin enthaltene Modellelement und Abhängigkeiten gegenüberzustellen. Aus der Gesamtheit der Ansätze werden Anforderungen an eine variantengerechte, modellbasierte Produktentwicklung abgeleitet und durch Experten vervollständigt (2). Diese literaturbasierten Anforderungen werden in einem Workshop mit Fachexperten aus unterschiedlichen Anwendungsbranchen erweitert. Anhand der Anforderungen werden die in (1) identifizierten Ansätze in einer Abgrenzungsmatrix bewertet (3). Der vielversprechendste Ansatz wird ausgewählt und um einen durchgängigen Modellierungsansatz erweitert (4). Die resultierende Methode und das dazugehörige Profil der Modellierungssprache wird auf ein Raupenlaufwerk für Landmaschinen der CLAAS Industrietechnik GmbH angewendet (5). In der Evaluation werden die Ergebnisse mit dem für das Raupenlaufwerk verantwortlichen Ingenieur analysiert und hinsichtlich Anwendbarkeit und Nutzerfreundlichkeit bewertet (6).

Systematische Literaturrecherche 1	Ableitung von Anforderungen 2	Bewertung der Ansätze 3	Erweiterung eines Ansatzes 4	Anwendung des erweiterten Ansatzes 5	Evaluierung des erweiterten Ansatzes 6
Identifikation existierender Ansätze	Analyse bzgl. durchgängiger Modellierung	Erstellung einer Abgrenzungsmatrix	Ausgleich von Defiziten bzgl. abgeleiteter und vervollständigter Anforderungen	Abbildung von Varianten und Beziehungen zwischen Varianten eines Systems	Vorstellung System und Methode
Analyse/Auswertung wiss. Datenbanken	Vervollständigung durch Experten	Auswahl durch Bewertungszahlen			Bewertung entsprechend der Anforderungen

Abbildung 2: Darstellung der Vorgehensweise

5. Ergebnisse und Diskussion

5.1. Systematische Literaturrecherche

Basierend auf den Forschungsfragen (siehe Abschnitt 3) und den Erkenntnissen aus dem Stand der Technik, werden deutsche und englische Schlüsselwörter sowie deren Synonyme für die Literaturrecherche abgeleitet. Die Schlüsselwörter werden durch Boolesche Operatoren zu Suchstrings zusammengesetzt, um wissenschaftliche Datenbanken zu untersuchen. Hierzu zählen die Library von IEEE, die Library der Design Society, Google Scholar, Web of Knowledge und Science Direct. Die Anzahl der identifizierten Dokumente und die Anzahl der ausgewählten Dokumente sind in Tabelle 1 dargestellt.

Tabelle 1: Darstellung der systematischen Literaturrecherche

	IEEE Library		Design Society		Google Scholar		Web of Knowledge		Science Direct	
	ID	AD	ID	AD	ID	AD	ID	AD	ID	AD
"variant modeling" AND ("model based systems engineering" OR "Mbse")	16	1	10	0	72	14	5	1	9	2
"variant management" and "sysml"	0	0	12	0	213	13	2	1	15	3
"modellbasiertes variantenmanagement"	-	-	5	0	267	12	-	-	-	-
„variantenmodellierung" AND ("model based systems engineering" OR "Mbse")	-	-	40	2	15	2	-	-	-	-
Selektierte Dokumente	1		2		41		2		5	

ID: identifizierte Dokumente AD: ausgewählte Dokumente

5.2. Anforderungen an eine variantengerechte modellbasierte Produktentwicklung

Die ausgewählten 51 Dokumente werden analysiert, um Anforderungen an die durchgängige Variantenmodellierung zu identifizieren. Hierzu werden textuelle Beschreibungen von Anforderungen aus den Dokumenten zusammengetragen. Anschließend werden die identifizierten Anforderungen analysiert und konsolidiert. Die daraus resultierenden Anforderungen an das modellbasierte Variantenmanagement sind in Tabelle 2 abgebildet.

Tabelle 2: Anforderungen an das modellbasierte Variantenmanagement

Anforderungen an das modellbasierte Variantenmanagement

Rückverfolgbarkeit [15, 22, 25–28]: Zwischen Modellartefakten müssen rückverfolgbare und rechnerinterpretierbare Abhängigkeiten modelliert sein

Abbildung der Varianz [13, 15, 27, 29]: Die Varianz von unterschiedlichen Modellartefakten muss explizit abgebildet werden.

Formalisierung [13, 15, 22, 27, 30]: Die Variantenmodellierung muss durch eine verfügbare formalisierte Modellierungssprache unterstützt werden.

Sichten [23, 28, 31]: Die Informationen müssen aus verschiedenen Sichten (funktional, logisch, physisch) darstellbar sein

Restriktionen [15, 28]: Restriktionen bei der Zusammenstellung von Varianten müssen explizit abgebildet werden.

Toolunterstützung [13]: Die Modellierung muss durch ein Softwarewerkzeug unterstützt werden.

Interdisziplinarität [23, 31]: Bei der Modellierung müssen interdisziplinäre Entwicklungsartefakte abgebildet werden können.

Tiefe & Granularität [13, 27, 28, 32]: Die Modellierung von Modellartefakten muss in unterschiedlicher Tiefe und Granularität entlang der Modellartefakte (Funktionen, logische Elemente, physische Elemente) möglich sein.

5.3. Abgrenzungsmatrix

Die Anforderungen an das modellbasierte Variantenmanagement werden in die Abgrenzungsmatrix übertragen, um die existierenden Ansätze zu bewerten. Aus den ausgewählten 51 Dokumenten wurden durch KO-Kriterien (z.B. fehlende methodische Beschreibung, fehlende Visualisierung, nicht interdisziplinär) die relevantesten Ansätze des modellbasierten Variantenmanagements ausgewählt. Die relevantesten 11 Ansätze werden anhand der hergeleiteten Anforderungen (vgl. Abschnitt 5.2) in einer Abgrenzungsmatrix bewertet und gegenübergestellt.

Tabelle 3: Abgrenzungsmatrix

Ansätze \ Anforderung	Rückverfolgbarkeit	Abbildung der Varianz	Formalisierung	Sichten	Restriktionen	Toolunterstützung	Interdisziplinarität	Tiefe & Granularität			Bewertungszahl
								F	L	P	
Hanna, M.; Schwede, L.N.; Krause, D., 2018 [6, 33]	●	◐	◐	●	◐	●	●	◐	◐	●	7,5
Zou et al., 2019 [34]	●	◐	◐	◐	◐	●	●	◐	◐	●	7
Colletti et al., 2020 [20]	●	◐	◐	◐	●	●	◐	○	◐	●	6,5
Melzer et al., 2016 [32]	●	●	◐	◐	○	●	◐	◐	◐	●	6,5
Weilkiens, T., 2016 [21]	●	●	●	●	◐	●	●	○	◐	●	8
White et al., 2021 [11]	◐	◐	◐	◐	○	○	●	○	○	●	4
Bilic, D., 2020 [27]	●	◐	◐	◐	○	●	●	○	◐	●	6
Bursac [35]	●	●	◐	◐	◐	●	●	◐	○	●	7
Vogel-Heuser et al., 2015 [36]	◐	●	◐	◐	◐	●	●	○	◐	●	6
Schulte, Dickopf, Standke, 2017 [15, 25]	●	◐	◐	◐	○	●	●	◐	◐	◐	6
Manhart et al., 2013 [37]	◐	◐	◐	●	○	○	◐	◐	◐	◐	4,5

Legende: F: Funktionen L: Logische Elemente P: Physische Elemente
 ○ Nicht erfüllt ◐ teilweise erfüllt ● vollständig erfüllt

Anhand eines Bewertungsschemas (0 = nicht erfüllt, 0,5 = teilweise erfüllt, 1 = vollständig erfüllt) wird die Erfüllung der Anforderungen durch die Ansätze bewertet. Bei neun Ansätzen wird die Entwicklung bei der Anwendung durch Softwaretools unterstützt. Als auszuführende Modellierungssprache nutzen zehn Ansätze SysML. Von den SysML Ansätzen ist lediglich das formalisierte SysML-Profil von Weilkiens [21] öffentlich zugänglich, was die Voraussetzung für eine Anpassbarkeit ist. Der Abgleich mit den hergeleiteten Anforderungen verdeutlicht, dass bisher kein Ansatz die Varianz durchgängig und explizit auf allen Entwicklungsartefakt-Ebene abbildet. 6 Ansätze bilden die Varianz von Funktionen, 9 Ansätze die Varianz logischer Elemente ab. Es existiert jedoch kein durchgängiger Lösungsansatz, der Abhängigkeiten entlang aller Ebenen von Entwicklungsartefakten herstellt.

5.4. Durchgängiger Ansatz zur Variantenmodellierung

Basierend auf den Bewertungen in der Abgrenzungsmatrix wurde der Ansatz VAMOS [21] als vielversprechendster Ansatz ausgewählt. Sieben der zehn definierten Anforderungen

werden vollständig erfüllt. Defizite des VAMOS Ansatzes sind die unvollständige Abbildung von benötigten Restriktionen und die Tiefe der Granularität in den Bereichen Funktionen und Lösungen. Die Unvollständigkeit der Restriktionen ist auf die geringe Anzahl an möglichen Zusammenhängen zwischen Varianten zurückzuführen, die in VAMOS enthalten sind. Diese lauten REQUIRES, welche die gleichzeitige Verwendung der beiden Varianten innerhalb eines Systems erfordert und XOR, welche die gleichzeitige Verwendung von zwei Varianten innerhalb eines Systems verbietet. Es wurden zwei neue Restriktionen ergänzt. REQUIRED BY und IOR. REQUIRED BY erfordert das Nicht-Vorhandensein einer Variante beim Nicht-Vorhandensein einer anderen Variante. IOR erfordert das Vorhandensein einer Variante sofern eine andere Variante nicht vorhanden ist (vgl. Abbildung 2)

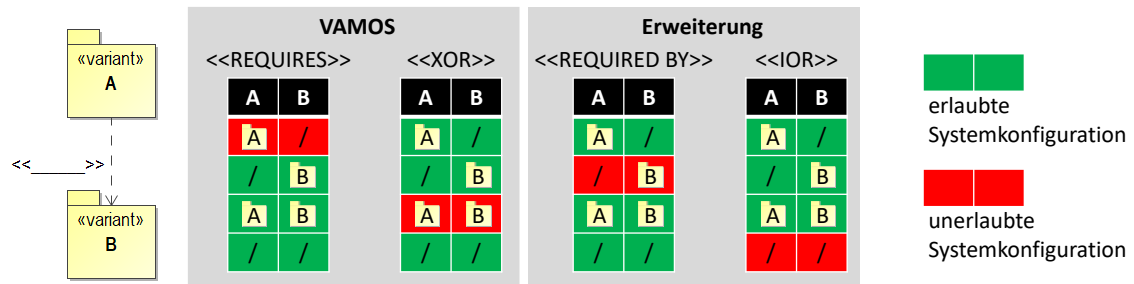


Abbildung 2: Restriktionen als Modellelemente des durchgängigen modellbasierten Variantenmanagements

Die Defizite in der Tiefe der Granularität werden durch die Kombination mit dem motego-Profil zur durchgängigen Modellierung behoben. Die Kombination mit dem VAMOS-Profil erlaubt abgebildeten Systeminformationen zusätzlich die Zuordnung zu einem Stereotypen aus dem Variantenmanagement. Somit kann Varianz auf jeder der vier Bereiche abgebildet werden. Insgesamt wurde das motego-Profil um die 4 Stereotypen zur Klassifizierung von Systemelementen bezüglich des Variantenmanagements ergänzt (Variant, Variation, Variation Point und Variant Configuration), und um vier Stereotypen zur Abbildung von Variantenrestriktionen (REQUIRES, XOR, REQUIRED BY und IOR) (vgl. Abbildung 3).

5.5. Evaluation

Die Evaluation der Methode erfolgt anhand eines Fallbeispiels. Für das System Raupenlaufwerk wurden 15 Funktionen variiert. Anschließend wurden 27 Varianten der funktionserfüllenden Lösungsprinzipien abgeleitet und modelliert. Auszüge der Modellierung sind in Abbildung 3 dargestellt. Die Evaluierung erfolgte mit vier Experten aus den Bereichen Produkt- und Softwareentwicklung. Zunächst wurde die Anwendbarkeit anhand der Anforderungen evaluiert, anschließend die Nutzerfreundlichkeit während der Anwendung. Die Ergebnisse der Diskussion zur Anwendbarkeit sind:

Rückverfolgbarkeit: Aufgrund der in der Methode definierten Relationen und Restriktionen zwischen den Systemelementen (beispielsweise *Generalisierungsbeziehung* zwischen *ElementaryFunction* und *Solution*, Abbildung 3) ist die Rückverfolgbarkeit zwischen den Modellelementen entlang der Entwicklungsartefakte vollständig erfüllt.

Abbildung der Varianz: VAMOS bietet die grundlegenden Stereotypen für eine eindeutige Abbildung und Zuordnung von Varianten. Durch die Kombination mit dem motego SysML-Profil ist die durchgängige Abbildung der Varianz vollständig *Variation Lösung Kraft_Leiten* mit *Variante Reifen* und *Variante Midroller*, Abbildung 3)

Formalisierung: Die Kombination der Ansätze wurde in der semiformalen Modellierungssprache SysML umgesetzt. Diese definiert die Semantik der Modellierung, der Syntax wird durch die Methode definiert. Damit ist die Modellierung formal.

Sichten: Das Profil fokussiert sich auf interdisziplinäre Entwicklungsartefakte (Anforderungen, Funktionen, Lösungen, Physische Elemente, siehe Abbildung 3). In der vorliegenden Evaluation wurde das Profil für einen mechatronisches technisches System angewendet, wodurch die Anforderung als vollständig erfüllt bewertet wird.

Restriktionen: Durch die Erweiterung um die zusätzlichen Restriktionen ist es den Entwicklern möglich, relevante Restriktionen zwischen zwei Varianten abzubilden. (Funktion Kraft leiten benötigt das Vorhandensein einer Lösung Kraft leiten, *Funktion REQUIRES Lösung*. Ein Midroller ist nur notwendig falls die Funktionsvariante Kraft gummigefedert leiten existiert, *Midroller REQUIRED BY gummigefedert* siehe Abbildung 3). Die Anforderung Restriktionen ist somit vollständig erfüllt.

Toolunterstützung: Durch die Nutzung der SysML als Modellierungssprache lässt sich das kombinierte Profil mit allen Tools nutzen, die diese Sprache unterstützen, beispielsweise dem Cameo Systems Modeler von Dassault Systèmes.

Interdisziplinarität: Die Allgemeingültigkeit des RFLP-Ansatzes, der in der motego-Methode zum Einsatz kommt, erlaubt die Anwendung des Profils für jeden Fachbereich der Entwicklung. (Bsp: Entwicklungs-Sicht: Es muss eine Variante von Kraft leiten existieren, *gummigefedert IOR luftgefedert*. Vertriebsicht: Midroller und Rad dürfen aus Design-Gründen nicht am selben System verbaut sein: *Midroller XOR Rad*, Abbildung 3) Somit kann Varianz für unterschiedlichste Disziplinen der Entwicklung eines Systems modelliert werden. Die Anforderung Interdisziplinarität ist somit erfüllt.

Tiefe & Granularität: In dem kombinierten Profil wurde eine gegenseitige Zuordnung der Modellelemente vorgenommen, sodass die Kompatibilität sichergestellt ist und die geforderte Tiefe und Granularität zur Abbildung der Varianten erreicht werden kann. Die Anforderung wird als vollständig erfüllt bewertet.

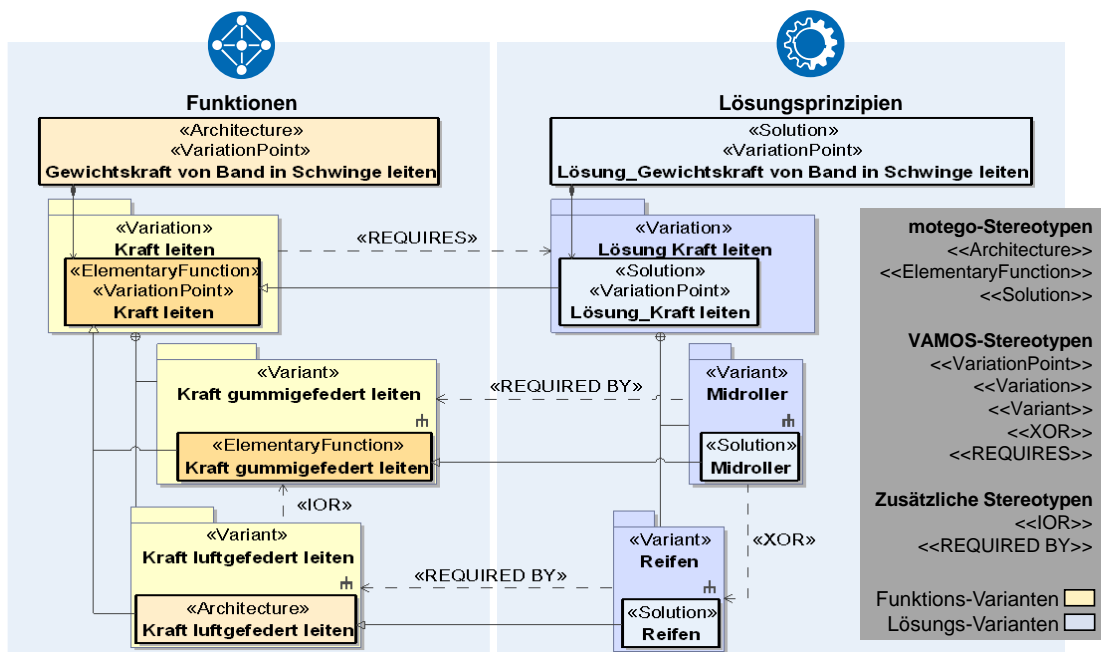


Abbildung 3: Modellelemente und Vorgehensweise für durchgängiges modellbasiertes Variantenmanagement in der Funktions- und Lösungsebene

Alle Anforderungen sind vollständig erfüllt, wodurch das Forschungsziel erreicht wurde und die identifizierten Defizite des VAMOS Ansatzes durch entsprechende Erweiterung und Kombination mit dem motego Ansatz reduziert wurden.

In der abschließenden Evaluation der Nutzerfreundlichkeit durch die Befragung der verantwortlichen Experten wurden folgende Bewertungen abgegeben: Die Entwickler werden durch das kombinierte Profil bei Entwicklungsentscheidungen auf funktionaler und logischer

Ebene unterstützt. Anhand der durchgängigen Abhängigkeiten und den ergänzten Restriktionen, können Entwicklungsentscheidungen nachvollziehbar und rückverfolgbar modelliert werden. Die Methode hilft bei der Beherrschung der Komplexität und damit einer wettbewerbsfähigen modellbasierten Entwicklung variantenreicher Systeme. Die konsistente Abbildung aller Abhängigkeiten bleibt aufgrund unvollständiger Informationsgrundlagen eine Herausforderung. Zur Steigerung der Akzeptanz im Unternehmen, soll der Aufwand der Modellierung durch teilweise Automatisierung der Modellierung reduziert werden.

6. Zusammenfassung und Ausblick

In dem vorliegenden Beitrag werden Ansätze des modellbasierten Variantenmanagements systematisch identifiziert, bewertet und erweitert. Defizite in der durchgängigen Modellierung von Varianten durch fehlende Modellelemente werden ausgeglichen. Der resultierende Modellierungsansatz bildet benötigte Modellelemente entlang der Entwicklungsartefakte Anforderungen, Funktionen, Lösungsprinzipien und physischen Elementen ab. Zusätzlich verfügt der Ansatz über notwendige Beziehungen zur Abbildung von hierarchischen Abhängigkeiten und Abhängigkeiten zur Sicherstellung von Durchgängigkeit. Der Ansatz wurde für die Modellierung eines variantenreichen Fallbeispiels angewendet und mit Hilfe von Experten evaluiert. Der Ansatz und die exemplarische Darstellung der Vorgehensweise unterstützt Entwickler bei der modellbasierten Abbildung von Varianz und bei der Festlegung von Entwicklungsentscheidungen in Bezug auf geeignete Lösungsalternativen. Potenziale existieren bei der Nutzung des Ansatzes zur Analyse technischer Änderungen [41, 42] sowie beim Einsatz von Künstlicher Intelligenz zur Sicherstellung der Konsistenz und zur Reduktion des Anwendungsaufwands für die Anwender des modellbasierten Ansatzes [43].

Danksagung

Diese Forschungsarbeit ist Teil des Projekts "ImPaKT" und wird vom Bundesministerium für Bildung und Forschung (BMBF) gefördert.

Literaturverzeichnis

- [1] Schuh, Günther; Riesener, Michael: Produktkomplexität managen. Strategien - Methoden - Tools. München: Hanser Verlag, 2018.
- [2] Eigner, Martin: System Lifecycle Management. Berlin, Springer Berlin Heidelberg, 2021.
- [3] Ponn, Josef; Lindemann, Udo (Hrsg.): Konzeptentwicklung und Gestaltung technischer Produkte. Berlin, Heidelberg, 2011.
- [4] Zigel, Matthias: Übergreifendes Konzept zur Strukturierung variantenreicher Produkte und Vorgehensweise zur iterativen Produktstruktur-Optimierung. Kaiserslautern, Techn. Univ., Diss., 2006.
- [5] Wyrwich, Christian et al.: Integration von Produktportfolios in die funktionsorientierte modellbasierte Systementwicklung. In: Binz et al. Stuttgarter Symposium für Produktentwicklung. Stuttgart, 2021.
- [6] Seiler, Florian M. et al.: MBSE zur Unterstützung der Produktentwicklung von modularen Produktarchitekturen. In: Krause, Dieter; Hartwich, Tobias S.; Rennpferdt, Christoph (Hrsg.) Produktentwicklung und Konstruktionstechnik. Berlin, Springer Verlag, 2020, S. 111–134.
- [7] Gräßler, Iris; Pöhler, Alexander: Produktentstehung im Zeitalter von Industrie 4.0. In: Maier, Günther; Engels, Gregor; Eckhard, Steffe: Handbuch Gestaltung digitaler und vernetzter Arbeitswelten, 2020.
- [8] Gräßler, Iris; Wiechel, Dominik: Systematische Bewertung von Auswirkungsanalysen des Engineering Change Managements. In: DS 111: Proceedings of the 32nd Symposium Design for X, 27, 2021.
- [9] Tschirner, Christian: Tag des Systems Engineering. Herzogenaurach, 25.-27.10 2016. Hanser, München, 2017.
- [10] Gräßler, Iris: Umsetzungsorientierte Synthese mechatronischer Referenzmodelle, 2015.
- [11] White, Dustin et al.: Environment modeling for evaluating system variants in model-based systems engineering. In: Procedia CIRP 104, 2021, S. 750–755.
- [12] Eigner, Martin; Koch, Walter; Muggeo, Christian (Hrsg.): Modellbasierter Entwicklungsprozess cybertronischer Systeme. Berlin, Springer Verlag, 2017.
- [13] Schäfer, Andreas et al.: Variability realization in model-based system engineering using software product line techniques: an industrial perspective, 2021.

- [14] Jacobs, Georg et al.: Function-Oriented Model-Based Product Development. In: Design Methodology for Future Products, 2022, S. 243–263.
- [15] Schulte, Tim et al.: Erweiterung des integrierten Konzeptes aus Prozessrahmenwerk und Beschreibungssystematik von mecPro² um ein modellbasiertes Variantenmanagement. In: Tag des Systems Engineering 2016.
- [16] Inkermann, David et al.: Die Produktarchitektur als zentrales Konzept in der Produktentwicklung. In: DS 98: Proceedings of the 30th Symposium Design for X (DFX 2019), 2019, S. 263–278.
- [17] Kang, Kyo C. et al.: Feature-Oriented Domain Analysis (FODA) Feasibility Study. 1990.
- [18] Pohl, Klaus; Böckle, Günter; Linden, Frank: Software product line engineering. Foundations, principles, and techniques ; with 10 tables. Berlin, Springer Verlag, 2005.
- [19] Haugen, Øystein; Wąsowski, Andrzej; Czarnecki, Krzysztof: CVL: common variability language. In: Proceedings of the 16th International Software Product Line Conference. New York, USA, 2012
- [20] Colletti, Ryan A. et al.: Best Practice Patterns for Variant Modeling of Activities in Model-Based Systems Engineering. In: IEEE Systems Journal 14, 2020, S. 4165–4175.
- [21] Weilkens, Tim: Variant modeling with SysML. MBSE4U, Fredesdorf, 2016.
- [22] Bilic, Damir et al.: An Integrated Model-Based Tool Chain for Managing Variability in Complex System Design. In: 2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C), 2019.
- [23] Albers, Albert et al.: Model-based systems engineering in modular design. In: Design Science 5 (2019).
- [24] Verein Deutscher Ingenieure: VDI 2235. Wirtschaftliche Entscheidungen beim Konstruieren.
- [25] Schulte, Tim; Dickopf, Thomas; Standke, Antje: Integration & CTP-Spezialisierung. In: Eigner, Martin; Koch, Walter; Muggeo, Christian (Hrsg.) Modellbasierter Entwicklungsprozess cybertronischer Systeme. Berlin, Springer Verlag, 2017, S. 87–92.
- [26] Wagner, Stefan; Lichter, Horst (Hrsg.): Software Engineering 2013 - Workshopband, Gesellschaft für Informatik e.V. Bonn, 2013.
- [27] Bilic, Damir: Managing Variability in SysML Models of Automotive Systems, Västerås, 2020.
- [28] Dumitrescu, Cosmin et al.: Capturing Variability in Model Based Systems Engineering. In: Aiguier et al. (Hrsg.) Complex Systems Design & Management. Cham, 2014, S. 125–139.
- [29] Wu, Dazhong et al.: SysML-based design chain information modeling for variety management in production reconfiguration. In: Journal of Intelligent Manufacturing 24, 2013.
- [30] Bartelt, Christian; Herold, Sebastian: Modellorientiertes Variantenmanagement. In: Mayr, Heinrich C. & Brey, Ruth (Hrsg.) Modellierung. Bonn, 2006, S. 173-182.
- [31] Bursac, Nikola; Albers, Albert; Schmitt, Thomas: Model Based Systems Engineering in Modular Design – A Potential Analysis using Portal Type Scraper Reclaimers as an Example. In: Procedia CIRP 50 2016.
- [32] Melzer, Sylvia et al.: Physische Architekturen variantengerecht aus Funktionalen Architekturen für Systeme (FAS) spezifizieren. In: Schulze et al. (Hrsg.) Tag des Systems Engineering. München, 2016, S. 429–438.
- [33] Hanna, Michael; Schwede, Lea-Nadine; Krause, Dieter: Model-based consistency for design for variety and modularization. In: The 20th International Dependency and Structure Modeling Conference. 2018.
- [34] Li, Huaxia et al.: Model-based Systems Engineering Process for Supporting Variant Selection in the Early Product Development Phase. In: IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 2019.
- [35] Bursac, Nikola: Model Based Systems Engineering zur Unterstützung der Baukastenentwicklung im Kontext der Frühen Phase der Produktgenerationsentwicklung. Karlsruhe, 2016.
- [36] Vogel-Heuser, Birgit et al.: Interdisziplinärer Produktlinienansatz zur Steigerung der Wiederverwendung. In: at - Automatisierungstechnik 63, 2015, S. 99–110.
- [37] Manhart, Peter et al.: Konzepte zur Erweiterung des SPES Meta-Modells um Aspekte der Variabilitäts- und Deltamodellierung. In: Wagner, Stefan; Lichter, Horst (Hrsg.) Software Engineering 2013 - Workshopband. Bonn, 2013, S. 283–292.
- [38] Zerwas, Thilo et al.: Mechanical concept development using principle solution models. RWTH Aachen, 2021.
- [39] Spütz, Kathrin et al.: Integration of Production and Cost Models in Model-Based Product Development. In: Open Journal of Social Sciences 09, 2021, S. 53–64.
- [40] Drave, Imke et al.: Modeling mechanical functional architectures in SysML. In: Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems. New York, NY, USA, 2020, S. 79–89.
- [41] Gräßler, Iris et al.: Model-Based Effect-Chain Analysis for Complex Systems. In: Proceedings of the Design Society 2, 2022, S. 1885–1894.
- [42] Graessler, Iris; Oleff, Christian; Scholle, Philipp: Method for Systematic Assessment of Requirement Change Risk in Industrial Practice. In: Applied Sciences 10, 2020, S. 86-97.
- [43] Gräßler, Iris.; Wiechel, Dominik; Pottebaum, Jens: Role model of model-based systems engineering application. In: IOP Conference Series: Materials Science and Engineering 1097, 2021.