OPTIMIERUNG FEHLERTOLERANTER FLUGZEUGSYSTEME MIT MEHRFACHEN SICHERHEITS- UND ZUVERLÄSSIGKEITSANFORDERUNGEN

Christian Raksch, **Frank Thielecke**

Institut für Flugzeug-Systemtechnik,
Technische Universität Hamburg-Harburg,
Nesspriel 5, 21129 Hamburg,
Deutschland

KURZFASSUNG

Der gesteigerte Funktionsumfang und die daraus bedingte erhöhte Vernetzung unterschiedlicher Flugzeugsysteme führen zu einer zunehmenden inner- und intersystemischen Komplexität aktueller und zukünftiger Systementwicklungen. Diese Komplexität wirkt sich neben der Realisierung zudem auf den Entwicklungsprozess aus, dessen Vorentwurfsphase aufgrund der bestehenden Variantenvielfalt nicht umfassend überblickt werden kann. Für eine transparente und zielführende Systementwicklung sind daher primär vor der Definition der Systemarchitektur und -technologien unterstützende Werkzeuge notwendig.

Dieser Beitrag widmet sich der Optimierung von komplexen Systemarchitekturen unter Berücksichtigung von mehrfachen Sicherheits- und Zuverlässigkeitsanforderungen sowie zusätzlichen häufig konträren Systemparametern, wie beispielsweise der Systemmasse. Den Hintergrund hierfür bildet ein bewährtes hybrides Systemanalysemodell unter Verwendung von Zuverlässigkeitsblockdiagrammen zur Abbildung der Systemtopologien und Fehlereignisse sowie nebenläufigen endlichen Automaten zur Modellierung eines dynamischen zustandsdiskreten Verhaltens. Auf Basis dieses hybriden Modells kann mit Hilfe eines mehrfach redundanten Systemmodells ein Gleichungssystem mit den modellierten Freiheitsgraden des Optimierungsproblems aufgestellt werden. Weitere Systemparameter, wie z.B. Architekturmassen oder der Leistungsverbrauch, können über frei definierbare Systemgleichungen berücksichtigt werden. Das vollständige Gleichungssystem dient anschließend der Zielwertberechnung eines mehrkriteriellen Optimierungsprozesses. Entsprechend des Problemcharakters und -umfangs kann aus mehreren Optimierungsverfahren ausgewählt werden. Der Beitrag stellt die Auswahlkriterien der Optimierungsverfahren vor und verdeutlicht diese anhand der verfahrenspezifischen Eigenschaften.

Die Anwendung des mehrkriteriellen Optimierungsprozesses und die Interpretation der Ergebnisse werden mit Hilfe eines anschaulichen und vereinfachten fehlertoleranten Beispielsystems demonstriert.

STICHWORTE

Flugzeugsysteme, Vorentwurf, Zuverlässigkeit, Sicherheit, Optimierung, Fehlertoleranz, Redundanz, Systemarchitektur

NOMENKLATUR

BB	**B**ranch and **B**ound
CFSM	**C**oncurrent **F**inite **S**tate **M**achines
GA	**G**enetischer **A**lgorithmus
MMEL	**M**aster **M**inimum **E**quipment **L**ist
MTBF	**M**ean **T**ime **B**etween **F**ailure
MRS	**M**ehrfach **R**edundantes **S**ystemmodell
NSGA	**N**on **S**orting **G**enetic **A**lgorithm
RAP	**R**edundancy **A**llocation **P**roblem
RBD	**R**eliability **B**lock **D**iagram
VE	**V**ollständige **E**numeration
ZÜW	**Z**ustands**ü**bergangs**w**ahrscheinlichkeit

$\mathcal{A}$	[–]	Zielwertmenge
a	[–]	Zustand aktiv
c	[–]	Zustand passiv-kalt
g	[–]	Generation
h	[–]	Zustand aktiv-heiss
i	[–]	Zustand isoliert
m	[kg]	Masse
K	[–]	Ereignisindikatorvariable
$\mathbf{K}$	[–]	Menge der Ereignisse
M	[–]	Minimalpfad
$\mathcal{N}$	[–]	Menge der Architekturbeschränkungen

p	$[-]$	Zustand passiv-warm
P	$[-]$	Wahrscheinlichkeit
$\mathcal{R}$	$[-]$	Menge probabilistischer Strukturfunktionen
$\mathcal{S}$	$[-]$	Menge zusätzlicher Systemgleichungen
VA	$[-]$	Vorgängerargument
VT	$[-]$	Vorgängerterm
$\mathbf{x}$	$[-]$	Architekturvektor
$\mathbf{K}$	$[-]$	Ereignismenge
$\mathbf{KA}$	$[-]$	aktive Ereignismenge
$\mathbf{KW}$	$[-]$	passiv-warme Ereignismenge
ϕ	$[-]$	Systemstrukturfunktion
λ	$[\mathrm{s}^{-1}]$	Fehlerrate
χ	$[-]$	Strukturvektor

1 EINLEITUNG

Der steigende Systemumfang und die steigende Vernetzung von Flugzeugsystemen führen zu einer erhöhten inner- und intersystemischen Komplexität. Im Vorentwurf wird mit der Architektur von Systemen maßgeblich die Sicherheit und Zuverlässigkeit eines Systems bestimmt, das von unterschiedlichen Redundanzstrategien geprägt ist. Neben den Sicherheitsvorschriften gemäß EASA CS-25 werden die Flugzeugsysteme bei aktuellen Flugzeugentwicklungen zudem durch die Anforderungen an die operationelle Zuverlässigkeit beeinflusst und durch die Kosten- und Massenkontingente im Systementwurf beschränkt [1, 2]. Die Auswahl einer Systemarchitektur bedarf der Berücksichtigung zahlreicher qualitativer wie quantitativer Kriterien und ingenieurtechnischer Erfahrung. Vor allem bei der Entwicklung neuer Systemtechnologien bietet sich aufgrund der Lösungsvielfalt ein teilweise nicht überschaubarer Lösungsraum. Für eine Begrenzung des Lösungsraumes hinsichtlich Sicherheit und Zuverlässigkeit und die Abschätzung weiterer Faktoren bietet der hier dargestellte Ansatz eine Lösung an. Weitere Kriterien der Systemarchitekturen können begleitend berechnet und abschließend mit den Sicherheits- und Zuverlässigkeitswerten kombiniert werden.

Der vorliegende Artikel stellt eine Optimierungsumgebung für Systemtopologien unter Berücksichtigung der quantitativen Sicherheit und Zuverlässigkeit vor, beschränkt durch konträre Ziele. Das Ziel des Optimierungsprozesses ist eine transparente Unterstützung der Konzeptfindung und die Einschränkung des vollständigen Lösungsraums. Zur Einschränkung der möglichen Lösungen werden zunächst architekurelle Beschränkungen genutzt und im nächsten Schritt wird die Menge der nichtdominierten Architekturen anhand der ermittelten Zielwerte bestimmt. Abschnitt 2 stellt das hierfür verwendete hybride Systemmodell vor, gefolgt von der Optimierungsumgebung für mehrere probabilistische Anforderungen und Beschränkungen. Das Konzept wird anschließend in Abschnitt 4 mit einem Anwendungsbeispiel eines vereinfachten elektrischen Energieversorgungssystems demonstriert.

2 HYBRIDES SYSTEMMODELL

Zur Optimierung fehlertoleranter Flugzeugsysteme mit unterschiedlichen Topologien und Redundanzstrategien ist ein geeignetes Systemmodell notwendig, das die Möglichkeit zur Einbindung von Freiheitsgraden im Systementwurf besitzt. Der in Abschnitt 3 vorgestellte Optimierungsprozess nutzt hierfür ein hybrides Systemmodell, das im Folgenden vorgestellt wird. Auf einer oberen Modellierungsebene werden Zuverlässigkeitsblockdiagramme zur Abbildung der System- und Ereignisstruktur genutzt. Auf einer unteren Ebene erfolgt die Modellierung von Zustandstransitionen und Abhängigkeiten zwischen einzelnen Ereignissen durch nebenläufige, endliche Zustandsautomaten.

2.1 Zuverlässigkeitsblockdiagramme

Die Grundlage eines Systemmodells für die spätere Optimierung bildet ein Zuverlässigkeitsblockdiagramm, das die Systemtopologie abbildet. Das zu untersuchende Ereignis wird hierbei in positiver Logik formuliert, z.B. „Wahrscheinlichkeit der Energieversorgung einer Sammelschiene“. Mathematisch wird dieses Ereignis durch eine Systemstrukturfunktion $\Phi(\mathbf{K})$ der Komponentenmenge $\mathbf{K}$ mit Hilfe Boolescher Algebra abgebildet. Das Ziel des vorgestellten Verfahrens ist es, diese Funktion für generische Strukturen zu ermitteln und zur weiteren Zielwertanalyse umzuformen. Hierfür wird jeder Komponente eine binäre Indikatorvariable K_i zugewiesen, für die Folgendes gilt [3]:

(1) $K_i = 1$ für eine funktionsfähige Komponente

und

(2) $K_i = 0$ für eine **nicht** funktionsfähige Komponente.

Mit Hilfe der Indikatorvariable K_i kann ein Erwartungswert $E(K_i)$ berechnet werden, der der Komponentenzuverlässigkeit R_i entspricht. Die Fehlerrate λ_i wird im Rahmen der Konzeptuntersuchungen entsprechend des mittleren Abschnitts der Badewannenkurve als konstant angenommen [3, 4]. Dieses ist aufgrund der generellen zeitlichen Unabhängigkeit und der statistisch verteilten Fehlereignisse bei Flugzeugsystemen zulässig [3].

Die Systemstrukturfunktion $\Phi(\mathbf{K})$ wird mit Hilfe von Minimalpfaden aus der Systemtopologie bestimmt. Nachfolgend werden die Minimalpfade mit Hilfe des HEIDTMANN Algorithmus orthogonalisiert und disjunktiv durch ein logisches ODER verknüpft [5]. Die resultierende Systemstrukturfunktion ist analog zur Indikatorvariable der Komponenten definiert [3].

Abbildung 1 verdeutlicht zur Bestimmung der Minimalpfade die Modellierung einer unidirektionalen Brückenschaltung mit Hilfe von Zuverlässigkeitsblockdiagrammen.

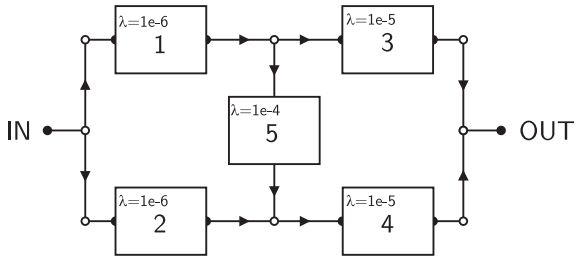

ABBILDUNG 1: Example of an unidirectional bridge structure RBD

Die resultierenden Minimalpfade dieses Beispielsystems sind:

(3) $M_1 = K_1 \wedge K_2$, $M_2 = K_3 \wedge K_4$ und $M_3 = K_1 \wedge K_5 \wedge K_4$.

Somit ergibt sich nach der Orthogonalisierung der Menge der Minimalpfade über die Systemstrukturfunktion die probabilistische Systemfunktion zu:

$$(4) \quad R_S = R_1R_2 + R_3R_4(1-R_1R_2)... \\ ... + R_1R_4R_5(1-R_2)(1-R_3) .$$

Die ermittelte Systemfunktion kann im Folgenden für die Berechnung des modellierten Szenarios genutzt werden.

2.2 Nebenläufige, endliche Zustandsautomaten

Als untere Modellierungsebene werden nebenläufige, endliche Zustandsautomaten (engl. *Concurrent Finite States Machines, CFSM*) zur Abbildung eines zustandsdiskreten dynamischen Systemverhaltens genutzt. Jedem Block des oberen Zuverlässigkeitblockdiagramms wird ein Zustandsautomat zugewiesen, dessen Transitionsbedingungen lokal in Abhängigkeit der weiteren Systemkomponenten und Ereignisse definiert werden.
Abbildung 2 verdeutlicht die möglichen Komponentenzustände und deren Transitionen. Die diskreten Zustände und dazugehörigen konstanten Fehlerraten sind dabei wie folgt definiert [6]:

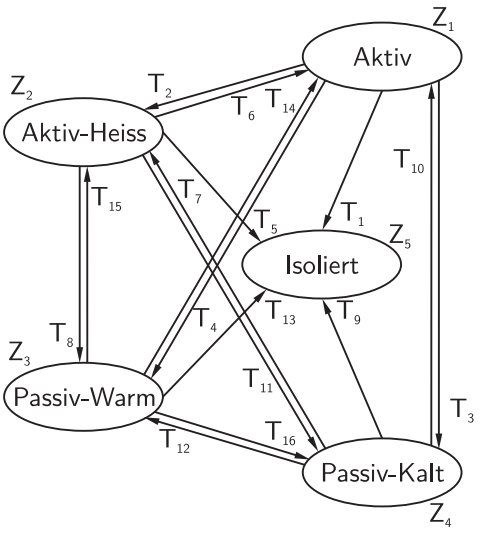

ABBILDUNG 2: Unterschiedliche Komponentenzustände und entsprechende Zustandstransitionen

"aktiv": die Komponente a ist mit Beginn der Flugmission der vollen Belastung ausgesetzt. Die Fehlerrate wird durch λ_a beschrieben.

"aktiv–heiss": mit Beginn der Flugmission ist die redundante Komponente h der gleichen Belastung wie der Arbeitskomponente a ausgesetzt. Für die Fehlerrate der Komponente gilt: $\lambda_h = \lambda_a$.

"passiv–warm": die redundante Komponente w ist einer geringeren Belastung ausgesetzt, sofern die Arbeitskomponente a funktionsfähig ist oder bis die Komponente selbst ausfällt. Die Fehlerrate liegt in dem Intervall $0 < \lambda_w < \lambda_a$.

"passiv–kalt": sofern kein erster Fehler im System vorliegt, ist die redundante Komponente c keiner Belastung ausgesetzt. Daraus folgt für die Fehlerrate: $\lambda_c = 0$.

"isoliert": Ausgefallener oder transitionsbedingter isolierter terminierender Zustand einer Komponente. Die Zustandsendung ist „i".

Die Transitionsbedingungen zwischen diesen Zuständen werden mit Hilfe einer logischen Syntax definiert, die die weiteren Komponenten und deren Zustände adressiert.

2.3 Interaktion der Systemmodellebenen

Die Verbindung der beiden Modellierungsebenen bildet das hybride Systemmodell zur quantitativen Sicherheits- und Zuverlässigkeitsanalyse. Abbildung 3 verdeutlicht diesen Ansatz und die Interaktion der Modellierungsebenen mit Hilfe der aktuellen Komponentenzustände und injizierter Komponentenfehler. Das hybride Systemmodell erlaubt es somit nicht nur unterschiedliche Komponentenzustände, sondern auch zustandsspezifische Fehlerraten zu berücksichtigen [7].

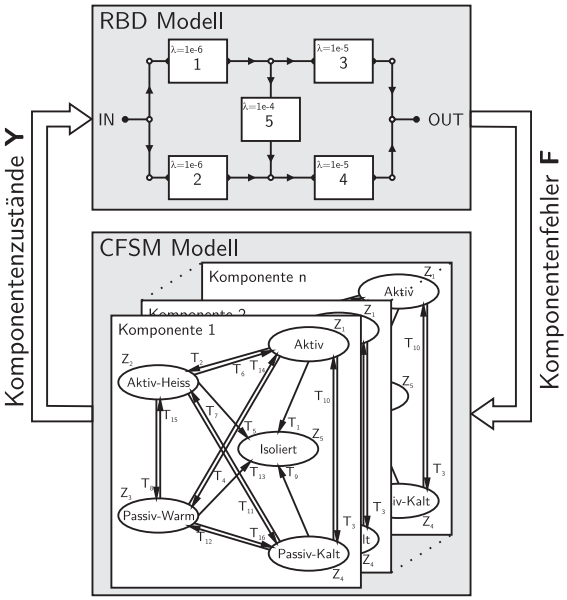

ABBILDUNG 3: Kopplung der Modellierungsebenen des hybriden Systemmodells

Mit Hilfe eines TIEFE-ZUERST Algorithmus kann das hybride Systemmodell genutzt werden, um einen vollständigen Zustandsraum des Systemmodells aufzustellen. Dieser Zustandsraum entspricht dem ermittelten Zustandsraum mit Hilfe von MARKOV-Ketten. Die weitere Nutzung bedingt jedoch keine Lösung von linearen Differentialgleichungen erster Ordnung zur Lösung der Wahrscheinlichkeiten der Zustandstransitionen. Stattdessen kann eine rekursive Formal genutzt werden, die auf Faltungsintegralen basiert. Gleichung 5 verdeutlicht die Nutzung der rekursiven Formal zur Bestimmung der Zustandswahrscheinlichkeiten, dabei werden die Vorgängerargumente VA und Vorgängerterme VT in den unterschiedlichen Degradationsstufen wieder verwendet. Neben den rekursiven Anteilen werden zudem Zustandübergangswahrscheinlichkeiten (ZÜW) berücksichtigt, die den Zustandswechsel einer Komponente, hervorgerufen durch eine Fehlerinjektion, probabilistisch charakterisieren. Die entsprechenden Terme sind für die zuvor vorgestellte Zustandsmenge in Tabelle 1 aufgeführt und werden entsprechend der Zustandsänderung in Gleichung 5 eingesetzt [6].

$$
(5)\qquad P_x^{\mathrm{p}}[\phi(\mathbf{K})](t) = \begin{cases} \underbrace{\dfrac{\lambda_x^{\{\mathrm{a,w}\}}}{\lambda[VA_1^1]}}_{VT_1^1} \cdot (1 - \underbrace{R_x^{\{\mathrm{a,w}\}} \cdot \mathrm{Z\ddot{U}W}_1}_{VA_1^1}) \cdot E[\phi(\mathbf{KA}_1)] \cdot E[\phi(\mathbf{KW}_1)] & \forall \quad x = 1 \\ \Bigg[\sum\limits_{j=1}^{2^{(x-2)}} \underbrace{VT_j^{(x-1)} \cdot \dfrac{\lambda_x^{\{\mathrm{a,w}\}}}{\lambda[VA_j^x]}}_{VT_j^x} \cdot (1 - \underbrace{R_x^{\{\mathrm{a,w}\}} \cdot \mathrm{Z\ddot{U}W}_x}_{VA_j^x}) + & \forall \quad x \geq 2 \\ + \sum\limits_{j=1}^{2^{x-2}} \underbrace{VT_j^{(x-1)} \cdot \dfrac{-\lambda_x^{\{\mathrm{a,w}\}}}{\lambda[VA_{j+2^{(x-2)}}^x]}}_{VT_{j+2^{(x-2)}}^x} \cdot (1 - \underbrace{R_x^{\{\mathrm{a,w}\}} \cdot \mathrm{Z\ddot{U}W}_x \cdot VA_j^{(x-1)}}_{VA_{j+2^{(x-2)}}^x}) \Bigg] \cdot \\ \cdot E[\phi(\mathbf{KA}_x)] \cdot E[\phi(\mathbf{KW}_x)] \end{cases}
$$

(6) mit $\mathbf{KA}_x, \mathbf{KW}_x \subseteq \mathbf{K}$ und

$$
(7)\qquad \phi(\mathbf{KA}_x) = \bigwedge_{g}^{g_{\max}} K_g \quad ,
$$

g ist der Zähler der *aktiven* and *aktiv-heißen* Komponenten nach dem Funktionsverlust der Komponente i zu der Degradationsstufe x,

$$
(8)\qquad \phi(\mathbf{KW}_x) = \bigwedge_{h}^{h_{\max}} K_h \quad ,
$$

h ist der Zähler der *passiv-warmen* Komponenten nach dem Funktionsverlust der Komponente i zu der Degradationsstufe x.

Durch Summation aller Zustandswahrscheinlichkeiten gemäß Gleichung 5, für die gilt $\phi(\mathbf{K}) = 1$, kann die quantitative Sicherheit und Zuverlässigkeit eines Systemmodells berechnet werden [7, 8]. Im Gegensatz zur Modellierung mit Hilfe von Zuverlässigkeitsblockdiagrammen ist dabei die Berücksichtigung von aktiven und passiven Komponenten zur Analyse unterschiedlicher Redundanzstrategien möglich.

TABELLE 1: Zustandsübergangswahrscheinlichkeiten

Zustandsübergang der Komponente	ZÜW
aktiv oder *aktiv-heiss* zu *isoliert*	(R_i^{a})
passiv-warm zu *isoliert*	(R_i^{w})
aktiv oder *aktiv-heiss* zu *passiv-warm*	$\left(\frac{R_i^{\mathrm{a}}}{R_i^{\mathrm{w}}}\right)$
aktiv oder *aktiv-heiss* zu *passiv-kalt*	(R_i^{a})
passiv-warm zu *aktiv* oder *aktiv-heiss*	$\left(\frac{R_i^{\mathrm{w}}}{R_i^{\mathrm{a}}}\right)$
passiv-warm zu *passiv-kalt*	(R_i^{w})
passiv-kalt zu *aktiv* oder *aktiv-heiss*	$\left(\frac{1}{R_i^{\mathrm{a}}}\right)$
passiv-kalt zu *passiv-warm*	$\left(\frac{1}{R_i^{\mathrm{w}}}\right)$

Für die anschließende Optimierung kann sowohl der Ansatz eigenständiger Zuverlässigkeitsblockdiagramme als auch das hybride Systemmodell genutzt werden, die Einbringung von Freiheitsgraden in die beschreibenden Systemgleichungen wird in dem nachfolgenden Abschnitt näher betrachtet.

3 MEHRKRITERIELLER OPTIMIERUNGSANSATZ

Die Nutzung des vorgestellten hybriden Systemmodells im Rahmen eines Redundanzallokationsproblems bedingt die Berücksichtigung von architekturellen Freiheitsgraden im Systemmodell. Das Prinzip des hierfür verwendeten

Mehrfach Redundanten Systemmodells wird im Folgenden vorgestellt, ebenso die Variation und Beschränkung der variablen Systemarchitektur mit Hilfe unterschiedlicher kombinatorischer Optimierungsansätze und die Zielwertberechnung der generierten Konzepte.

3.1 Mehrfach redundantes Systemmodell

Das hybride Analysemodell kann mit Hilfe künstlicher Redundanzen und serieller Anordnungen in der Systemstruktur für eine variable Architekturanalyse genutzt werden. Die resultierenden Modellgleichungen dieses mehrfach redundanten Systemmodells werden mit Hilfe der in Abschnitt 2 dargestellten Verfahren unter Berücksichtigung von Nebenbedingungen auf eine gültige Architektur beschränkt. Die zeitvarianten probabilistischen Sicherheits- und Zuverlässigkeitsfunktionen werden somit in eine Menge $\mathcal{R}(t,\mathbf{x})$ von zeit- und architekturvarianten Funktionen überführt. Der binäre Architekturvektor $\mathbf{x}$ steuert dabei die Berücksichtigung variabler Ereignisse aus der Menge K_v, daneben werden Komponenten der Menge der konstanten Komponenten K_f in der Systemarchitektur nicht variiert.
Der Einfluss des Architekturvektors $\mathbf{x}$ variiert in Abhängigkeit des verwendeten Systemmodells und der Anordnung der variablen Komponente. Für eine Modellierung unter ausschließlicher Verwendung der Zuverlässigkeitsblockdiagramme gilt für das Szenario j und das Ereignis i in Abhängigkeit der Fehlerrate, des Strukturvektors χ_i^j und der Analysezeit:

$$R_i = e^{-\lambda \cdot t \cdot \chi_i} \quad . \tag{9}$$

Die Einträge des szenariospezifischen Strukturvektors $\boldsymbol{\chi}_\mathbf{j}$ sind dabei wie folgt definiert:

$$\chi_i^j \begin{cases} = 1 & \text{für } x_i = 1 \quad \text{mit} \quad x_i \in \mathbf{x} \quad , \\ & \text{bei beliebiger Anordnung.} \\ \to \infty & \text{für } x_i = 0 \quad \text{mit} \quad x_i \in \mathbf{x} \quad , \\ & \text{bei paralleler Anordnung.} \\ = 0 & \text{für } x_i = 0 \quad \text{mit} \quad x_i \in \mathbf{x} \quad . \\ & \text{bei serieller Anordnung.} \end{cases} \tag{10}$$

Somit wird durch den Eintrag $x_i = 0$ bei serieller Anordnung ein neutrales Element bewirkt oder es werden bei einer parallelen Anordnung alle Minimalpfade mit der betroffenen variablen Komponente aus den Strukturfunktionen entfernt. Dieses kann mit Hilfe der unidirektionalen Querverbindung K_5 der zuvor dargestellten Brückenstruktur als optionale Komponente veranschaulicht werden. Durch Variation von x_5 ergeben sich die folgenden probabilistischen Strukturfunktionen:

$$R_S = \begin{cases} R_1R_2 + R_3R_4(1-R_1R_2) + \ldots \\ \ldots + R_1R_4R_5(1-R_2)(1-R_3) & \text{für } x_5 = 1 \\ \\ R_1R_2 + R_3R_4(1-R_1R_2) & \text{für } x_5 = 0 \end{cases} \tag{11}$$

Der Einfluss des Architekturvektors auf das hybride Systemmodell wurde bereits in einer vorherigen Veröffentlichung vorgestellt [8].

Die verwendeten Optimierungsverfahren nutzen zur Zielwertberechnung somit nur eine generische probabilistische Strukturfunktion, so dass eine wiederholte Erstellung mit Hilfe der Minimalpfade oder der Analyse des vollständigen Zustandsraums entfällt. Neben der Steuerung des Architekturvektors ist zudem häufig eine Beschränkung des Lösungsraums auf technisch sinnvolle Lösungen notwendig, dieses wird im folgenden Abschnitt näher erläutert.

3.2 Optimierungsprozess

Bei dem formulierten Redundanzallokationsproblem (engl. *Redundancy Allocation Problem (RAP)*) mit Hilfe des MRS handelt es sich um ein NP-schweres kombinatorisches Optimierungsproblem, d.h. das Problem ist mit steigender Größe der Menge K_v deterministisch nicht in polynomiell ansteigender Zeit effizient berechenbar [9]. Entsprechende Verfahren zur Lösung gängiger kombinatorischer Probleme, wie den bekannten Formulierungen des *Travelling Salesman Problem* oder des Rucksackproblems, wurden bereits vielfach untersucht und deren Anwendbarkeit nachgewiesen [10]. Die Anwendung der nichtlinearen Verfahren dieser bewährten Algorithmen auf das aufgestellte MRS erlaubt somit eine probate Optimierung beliebiger komplexer Systemstrukturen.

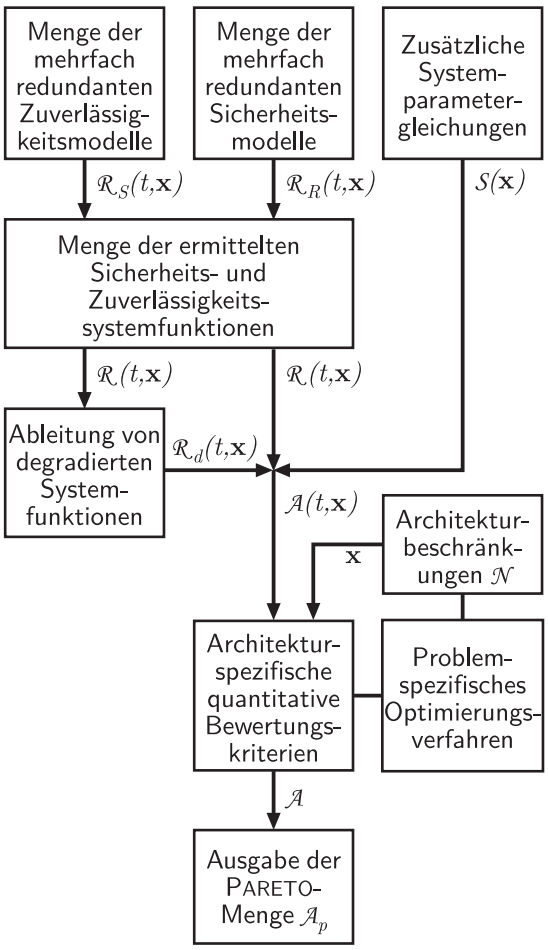

ABBILDUNG 4: Optimierungsprozess für mehrfache Sicherheits- und Zuverlässigkeitsanforderungen

Abbildung 4 verdeutlicht den Optimierungsprozess beginnend mit den probalistischen Gleichungsmengen $\mathcal{R}_S(t,\mathbf{x})$ und $\mathcal{R}_R(t,\mathbf{x})$ der MRS sowie entsprechender Komponentenparameter und der Gleichungsmenge $\mathcal{S}(\mathbf{x})$ für zusätzliche Systemparameter, wie z.B. der Systemmasse.
Mit Hilfe der in Abschnitt 2 vorgestellten Verfahren können entsprechend der Systemmodelle die probabilistischen Strukturfunktionen $\mathcal{R}(t,\mathbf{x})$ erstellt werden. Zur quantitativen Sicherheits- und Zuverlässigkeitsbewertung degradierter Systemzustände, z.B. MMEL Bedingungen, können die ermittelten Systemfunktionen zur Ableitung degradierter Funktionen genutzt werden [8]. Die generische Architekturbeschreibung $\mathcal{A}(t,\mathbf{x})$ wird im weiteren Verlauf durch ein problemspezifisches Optimierungsverfahren konkretisiert und mit Hilfe einer Gleichungsmenge $\mathcal{N}$ beschränkt. In Abhängigkeit der Problemgröße und Eigenschaften der Zielgrößen stehen drei Verfahren zur Lösung des formulierten RAP zur Auswahl:

- Vollständige Enumeration (VE) basierend auf einem TIEFE-ZUERST-Algorithmus: Systeme mit einer kleinen bis mittleren Variantenanzahl, beliebige Zielgrößen.
- Branch & Bound Verfahren (BB) auf Basis von NAKAGAWA [11]: Systeme mit einer mittleren Variantenanzahl, eine probabilistische Zielgröße und beliebige weitere Systemparameter.
- Genetischer Algorithmus (GA) auf Grundlage des NSGA-II [13]: Systeme mit einer großen Variantenanzahl, beliebige Zielgrößen.

Der Ansatz des Mehrfach Redundanten Systemmodells ermöglicht die generische Aufstellung unterschiedlicher System- und Ereignisstrukturen. Hierbei können konträre Sicherheits- und Zuverlässigkeitswerte abgebildet werden. Neben diesen probabilistischen Modellen besteht zudem die Möglichkeit zur Berücksichtigung weiterer Systemparameter, z.B. die Abschätzung der Systemmasse. Jede Systemarchitektur ist somit durch die Kombination der beschreibenden Systemparameter $\mathcal{A}(t,\mathbf{x})$ und den Architekturvektor $\mathbf{x}$ für eine gegebene Analysezeit t charakterisiert. Die Bestimmung optimaler Lösungen bietet auf Basis dieser Ergebnisse des mehrkriteriellen Optimierungsproblems drei unterschiedliche Verfahren an [14]:

- A-priori Entscheidung, die unterschiedlichen Architekturcharakteristika werden vorab geordnet und eventuell entsprechend gewichtet, so dass eine beschreibende Zielfunktion erreicht wird.
- A-posteriori Entscheidung, die unterschiedlichen Architekturcharakteristika werden nach der Optimierung gewichtet, das Ziel des Prozesses ist die Ermittlung der möglichst vollständigen PARETO-Menge.
- Progressive Entscheidung, nach jeder Optimierungsschleife greift der Nutzer in den Optimierungsprozess ein und wählt präferierte Architekturen zur weiteren Verwendung aus.

Aufgrund der zu erhaltenen Transparenz im komplexen Vorentwurfsprozess ist vor allem eine a-posteriori Entscheidung nach der eventuell teilweisen Ermittlung der PARETO-Menge sinnvoll. Die Lösungen der PARETO-Menge $\mathcal{A}_p$ sind dabei im Falle der Minimierung der Menge der Zielfunktionen $\mathcal{A}(t,\mathbf{x})$ wie folgt definiert [14]:

Definition 1 *Streng* PARETO*-optimal: eine Architektur* $\mathbf{x}^\star$ *ist streng* PARETO*-optimal, wenn keine Architektur* $\mathbf{x}$*,* $\mathbf{x} \neq \mathbf{x}^\star$ *besteht, für die gilt* $\mathcal{A}_i(t,\mathbf{x}) \leq \mathcal{A}_i(t,\mathbf{x}^\star)$ *mit* $t = \mathrm{const}$ *und* $i = 1,\ldots,k$ *Bewertungskriterien.*

Das Ergebnis der Optimierung ist somit nicht eine Lösung, z.B. basierend auf einer gewichteten Summe, sondern eine Vielzahl optimaler Lösungen mit unterschiedlichen Vor- und Nachteilen, resultierend aus den konträren Zielfunktionen. Mit Hilfe der visualisierten PARETO-Menge als Ergebnisfront kann im Anschluss an die Optimierung der Lösungsraum für die Entscheidungsfindung quantitativ beschränkt und eine Untermenge für weitere detaillierte Untersuchungen ausgewählt werden.
Das Verfahren wurde mit der institutseigenen Software SYRELAN (*System Reliablity Analysis*) in Verbindung mit MATHWORKS MATLAB umgesetzt.
Exemplarisch wird im folgenden der Ablauf des verwendeten Genetischen Algorithmus näher betrachtet. Dieser basiert auf dem Prinzip der natürlichen Selektion und ermittelt heuristisch einen Teil der PARETO-optimalen Menge, die im folgenden Abschnitt näher erläutert wird. Aufgrund des heuristischen Ansatzes ist eine Ermittlung der vollständigen PARETO-Menge nicht garantiert, dafür bietet der Ansatz jedoch die Möglichkeit zur Lösung großer Probleme. Der Optimierungsalgorithmus kann dabei in die folgenden Abschnitte unterteilt werden [15]:

Initialisierung: Erzeugung einer zufälligen Population von n Chromosomen, jede Lösung wird durch einen binären Architekturvektor $\mathbf{x}$ definiert.
Zielwertberechnung: die Zielwerte jeder Architektur werden mit Hilfe der ermittelten probabilistischen Zielfunktionen und den zusätzlichen Systemgleichungen berechnet.
Pareto-Dominanz: Nach jeder Generation wird die PARETO-Dominanz aller bisher ermittelten Architekturen bestimmt, anschließend werden alle Lösungen anhand der Dominanzwerte sortiert.
Fitness Berechnung: Die PARETO-Dominanz der Architekturen dient zur Bestimmung der Fitness, d.h. der Wahrscheinlichkeit der Reproduktion in der nächsten Generation.
Selektion: Lösungen höher Fitness werden für die nächste Optimierungsschleife ausgewählt.
Kreuzung: Die binären Architekturvektoren der Eltern werden mit einer gegebenen Wahrscheinlichkeit gekreuzt.
Mutation: Zur Vermeidung der frühzeitigen Konvergenz auf einen bestimmten Lösungsbereich werden die binären Architekturvektoren einer zufälligen Mutation unterzogen.
Termination: Die Optimierung terminiert nach nach g Generationen.

Die Beschränkungen zur Untersuchung ausschließlich technisch sinnvoller und zulässiger Lösungen können mit Hilfe einer Menge $\mathcal{N}$ von beschränkenden Gleichungen und Ungleichungen unter Nutzung der Architekturvektoren realisiert werden. Dieses wird in dem folgenden Abschnitt anhand des Beispielsystems näher erläutert.

4 BEISPIELANWENDUNG

Nachdem in den vorherigen Kapiteln die Grundlagen der Zielwertberechnung und der Optimierungsprozess dargelegt wurden, folgt in diesem Abschnitt die Anwendung des Verfahrens auf ein vereinfachtes Beispielsystem. Hierfür werden die MRS mit den Architekturbeschränkungen vorgestellt, der ermittelte vollständige Architekturraum mit der tatsächlichen Pareto-Front und das ermittelte Ergebnis unter Anwendung des Genetischen Algorithmus. Als Beispielanwendung dient ein vereinfachtes elektrisches Split-Bus-System eines zweistrahligen Kurzstreckenflugzeugs.

4.1 Systemmodelle

Das Beispielsystem bietet die Möglichkeit zur Einbringung einer erzeugerseitigen heißen Redundanz im Flug, die auch in der Lage wäre, Leistungsspitzen der Hauptgeneratoren aufzufangen. Durch die Einbringung könnten die Hauptgeneratoren entsprechend kleiner ausgelegt werden bzw. unterhalb der Nennleistung betrieben werden, was zu einer höheren MTBF (engl. *Mean Time Between Failure*) führen könnte.
Neben der Einbringung der heißen Erzeugerredundanz stehen unterschiedliche Konzepte mit möglichen Querverbindungen zwischen den Hauptpfaden zur Diskussion. Hierfür können entsprechende Schalter (*Bus Tie Contactor (GLC)* und *DC Tie Contactor (DCTC)*) in der Architektur berücksichtigt werden. Die Querverbindungen beziehen sich dabei auf die Möglichkeit elektrische Leistung von einem Pfad zu einem parallelen Pfad zu speisen, sofern der primäre Generator des Pfades ausgefallen sein sollte. Die Abbildung der Schaltmöglichkeit kann im Modell entweder mit Hilfe der nebenläufigen, endlichen Automaten oder als konservative Modellierung unter Nutzung der Zuverlässigkeitsblockdiagramme abgebildet werden.
Neben architekturellen Fragen ist ebenso eine Analyse zur Nutzung unterschiedlicher Hersteller für die Hauptgeneratoren (GEN 1 und 2), den redundanten Generator (Gen 3) und die Gleichrichter (engl. *Transformer Rectifier Unit (TRU)*) möglich. Die Sammelschienen (*AC* und *DC*) unterschiedlicher Spannungsniveaus sowie die *Generator Line Contactors (GLC)* werden als feste Komponente berücksichtigt.
In Abbildung 5 ist das MRS für die Sicherheitsanforderung „Wahrscheinlichkeit der Leistungsversorgung einer DC Sammelschiene.“ dargestellt.

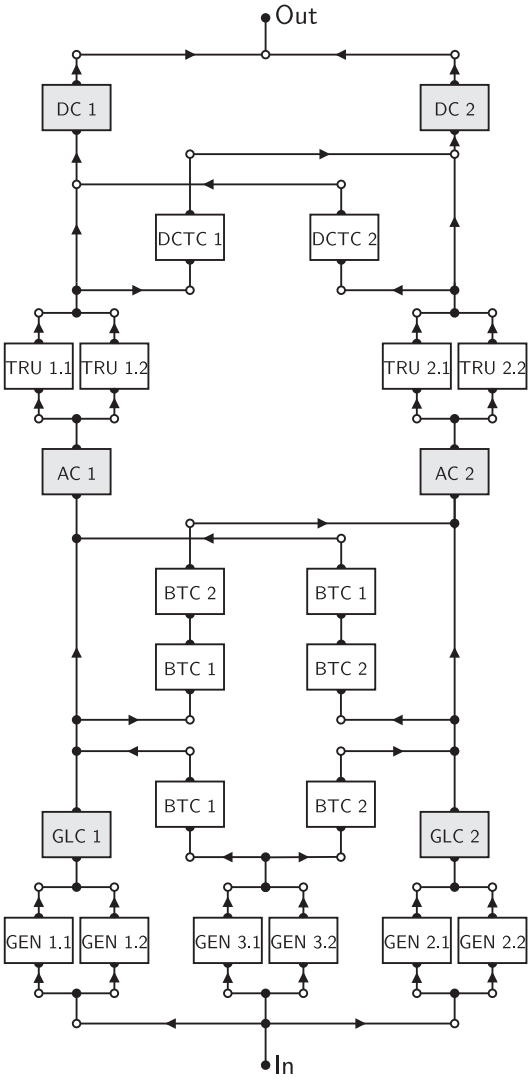

ABBILDUNG 5: Vereinfachtes mehrfach redundantes Systemmodell zur Untersuchung der Wahrscheinlichkeit der Versorgung der DC-Sammelschienen

Neben diesem Ereignis wird zudem die operationelle Zuverlässigkeit des Systems betrachtet. Diese Anforderung verhält sich gegensätzlich zur Systemsicherheit, da in diesem Fall keine MMEL-Bedingungen betrachtet werden und somit fehlerhafte Komponenten der Generatoren zum verspäteten Abflug führen würden. Das Zuverlässigkeitsmodell besteht in diesem Fall aus der seriellen Anordnung der Komponenten bis zu den AC-Sammelschienen. Generell ermöglicht der Ansatz der Minimalpfade jedoch die Einbringung jeder komplexen Struktur.
Aufgrund der Systembeschränkungen schränkt sich der gesamte Lösungsraum auf sechs Architekturen ein, die in Tabelle 2 dargestellt sind und durch einen Satz von Testparametern automatisch identifiziert werden können. Hierfür werden identische Fehlerraten und zusätzliche Parameter für alle Komponenten angenommen. Alle Architekturen mit gleichen Zielfunktionswerten bilden somit eine Untermenge des vollständigen Lösungsraumes $\mathcal{A}$ als identische

Architekturen gemäß Tabelle 2. Die Topologien sind zur vereinfachten Darstellung in der Tabelle um neunzig Grad gedreht dargestellt.

TABELLE 2: Prinzipielle Architekturen

Architektur	Topologie
Architektur 1, keine Querverbindungen, kein zusätzlicher Generator	
Architektur 2, Querverbindungen auf AC-Ebene, kein zusätzlicher Generator	
Architektur 3, Querverbindungen auf DC-Ebene, kein zusätzlicher Generator	
Architektur 4, Querverbindungen auf AC- und DC-Ebene, kein zusätzlicher Generator	
Architektur 5, Querbindung auf AC-Ebene, zusätzlicher Generator	
Architektur 6, Querverbindung auf AC- und DC-Ebene, zusätzlicher Generator	

Die Einschränkungen der möglichen Architekturen basieren dabei auf der folgenden Menge $\mathcal{N}$ der beschränkenden Nebenbedingungen in Form von Gleichungen und Ungleichungen der Einträge des Architekturvektors **x**.

$$
\begin{aligned}
x_{GEN1.1} + x_{GEN1.2} &= 1 \\
x_{GEN2.1} + x_{GEN2.2} &= 1 \\
x_{GEN1.1} - x_{GEN2.1} &= 0 \\
x_{GEN3.1} + x_{GEN3.2} &\leq 1 \\
x_{GEN3.1} + x_{GEN3.2} - x_{BTC1} - x_{BTC2} &\leq 0 \\
x_{BTC1} - x_{BTC2} &= 0 \\
x_{TRU1.1} + x_{TRU1.2} &= 1 \\
x_{TRU2.1} + x_{TRU2.2} &= 1 \\
x_{TRU1.1} - x_{TRU1.3} &= 0 \\
x_{DCXC1} - x_{DCXC2} &= 0
\end{aligned}
\tag{12}
$$

Aufgrund der Nebenbedingungen werden somit nur technisch sinnvolle Lösungen unter Gesichtspunkten der Zertifizierung berücksichtigt. Die sechs prinzipiellen Architekturen gemäß Tabelle 2 werden aufgrund der unterschiedlichen Komponentenkombinationen auf insgesamt 32 Lösungen erweitert.
Neben einer Sicherheits- und Zuverlässigkeitsanforderung wird zudem die approximative Systemmasse durch Summation der Komponentenmassen bestimmt.

4.2 Optimierungsergebnisse

Die Ergebnisse der Vollständigen Enumeration und des Genetischen Algorithmus sind in Abbildung 6 für die Sicherheit $F_{S,sys}$ und Zuverlässigkeit $F_{R,sys}$ dargestellt. Abbildung 7 veranschaulicht die Ergebnisse für die Sicherheit und relative Systemmasse m. Aufgrund der Variantenzahl wäre in diesem Fall ausschließlich eine Optimierung auf Grundlage der Vollständigen Enumeration angebracht, zur Darstellung der Ergebnisse des Genetischen Algorithmus werden auch diese betrachtet.

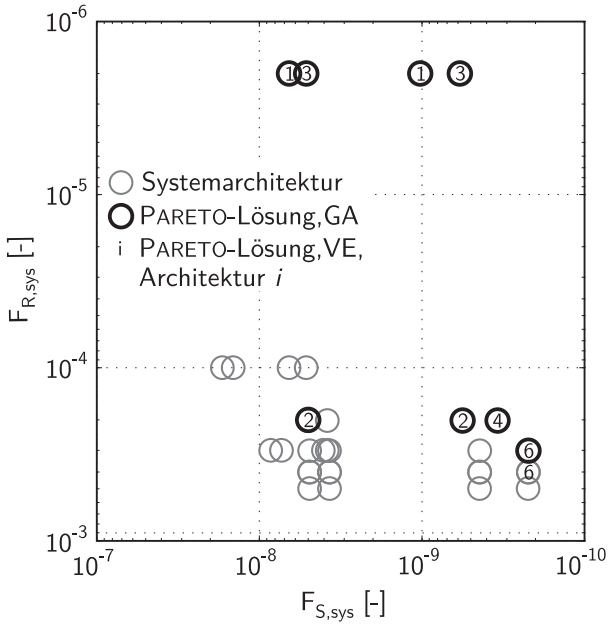

ABBILDUNG 6: Projektion des Lösungsraums mit PARETO-Menge für die Zielwerte Sicherheit und Zuverlässigkeit

Durch die drei unterschiedlichen Bewertungskriterien ist die grafische Interpretation der PARETO-Eigenschaft nur begrenzt gegeben. Dennoch lassen sich die konträren Zusammenhänge zwischen der maximalen quantitativen Sicherheit und Zuverlässigkeit in Abbildung 6 erkennen, ebenso zwischen der maximalen Sicherheit und der minimalen Systemmasse in Abbildung 7. Dabei zeigt sich, dass der Genetische Algorithmus nicht die vollständige Menge der nichtdominierten Lösungen ermitteln konnte. Trotzdem ist die ermittelte Lösungsmenge eine gute Approximation der tatsächlichen Front.
Die Annäherung der PARETO-Menge mit Hilfe des Genetischen Algorithmus wurde mit einer Populationsgröße von 140 Chromosomen innerhalb von dreißig Generationen ermittelt. Der gesamte Optimierungsprozess dauerte dabei auf einem Standard-PC 283 s. In dieser Zeit wurden ca. 1500, größtenteils identische, Architekturen

durch den Genetischen Algorithmus erzeugt und gerechnet. Die übrigen Architekturen, die sich durch Multiplikation der Chromosonen- und Generationenanzahl ergibt, wurden aufgrund von Verstößen gegen die Nebenbedingungen in den Zielgrößen abgewertet und somit für die Selektion nicht berücksichtigt. Bei einer unbeschränkten Variantenanzahl von $2^{14} = 16384$ aufgrund der Größe der variablen Ereignismenge K_v und nur 32 gültigen Architekturen zeigt sich somit eine gute Konvergenz der Populationen hin zu gültigen Systemarchitekturen. Die Betrachtung der Anfangs- und Endpopulation bestätigt dieses. Während die zufällige Initialpopulation unter einem Prozent gültige Lösungen enthält, steigt die Anzahl bis zur letzten Population auf ca. 85 % an.

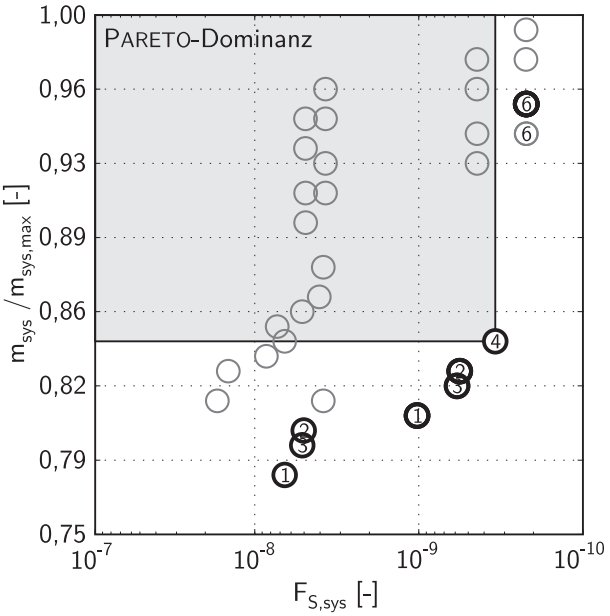

ABBILDUNG 7: Projektion des Lösungsraums mit PARETO-Menge für die Zielwerte Sicherheit und Systemmasse

Die ermittelten PARETO-optimalen Lösungen verdeutlichen, dass die Architektur 5 mit einem zusätzlichen Generator und einer Querverbindung auf dem AC-Spannungslevel keine Vorteile bezüglich der untersuchten Zielwertfunktionen bietet und somit für weitere Betrachtungen vernachlässigt werden kann. Wie erwartet stellen die Architekturen 1 und 6 die Extremlösungen der PARETO-Menge dar. Für den weiteren Entscheidungsprozess ist vor allem die ausgeglichene so genannte *Knie*-Region der Menge der nichtdominierten Lösungen interessant. Diese verdeutlicht in Abbildung 7, dass die Architekturen 2, 3 und 4 mit den komponentenspezifischen Derivaten annährend die quantitativen Sicherheitswerte der hochredundanten Architektur 6 erreichen, jedoch ohne einen dritten Generator auskommen und somit einer relativen Massenersparnis zwischen den unterschiedlichen Regionen von ca. 13%.
Im weiteren Entwicklungsprozess kann somit die Lösungsmenge entsprechend der sinnvollen Lösungen der PARETO-Menge eingeschränkt und detailliert qualitativ und quantitativ betrachtet werden.

5 ZUSAMMENFASSUNG

Der vorliegende Beitrag stellt ein Verfahren zur Optimierung fehlertoleranter Flugzeugsystemarchitekturen vor. Das Analysemodell besteht hierbei aus dem hybriden Systemmodell unter Verwendung von Zuverlässigkeitsblockdiagrammen zur Abbildung der Systemtopologie und unter Umständen nebenläufigen, endlichen Zustandsautomaten zur Modellierung eines dynamischen, zustandsdiskreten Systemverhaltens. Dieses Systemmodell wird zur Modellierung eines Mehrfach Redundanten Systemmodells genutzt, dessen generische Strukturfunktionen zur Berechnung der Zielwerte eines problemspezifischen Optimierungsverfahrens genutzt werden, zusätzlich ist die Berücksichtigung weiterer summativer Zielgrößen möglich. Aufgrund der konträren Zielgrößen und der Komplexität des Systementwurfs ist das Ziel der Optimierung keine Einzellösung, sondern die PARETO-Menge des vollständigen Architekturraums.
Der Optimierungsprozess wurde für mehrfache probabilistische Anforderungen mit Hilfe eines vereinfachten elektrischen Energieversorgungssystems demonstriert. Zur Ermittlung des vollständigen Architekturraums und der tatsächlichen PARETO-MENGE diente die Vollständige Enumeration auf Basis der TIEFE-ZUERST-Suche. Neben dem exakten Ergebnis wurde zudem die Approximation der nichtdominierten Lösungen mit Hilfe eines Genetischen Algorithmus dargestellt.
Das Ziel der weiterführenden Betrachtungen sollte die Integration der Optimierungsergebnisse in den Systementwicklungsprozess sein. Somit wäre neben der üblichen parametrischen Optimierung von Komponenten eine optimale Architektur und Redundanzstrategie gewährleitet, die maßgeblich die Sicherheit und Zuverlässigkeit von Flugzeugsystemen bestimmen.

DANKSAGUNG

Die Autoren danken den beiden Gutachtern für die konstruktiven Anmerkungen und Korrekturen im Review-Prozess.

LITERATUR

[1] EUROPEAN AVIATION SAFETY AGENCY 2008, *Certification Specification 25 - Large Aeroplanes*, Brüssel, Belgien.

[2] KOEPPEN, C. 2006, *Methodik zur modellbasierten Prognose von Flugzeugsystemparametern im Vorentwurf von Verkehrsflugzeugen*, Technische Universität Hamburg-Harburg, Diss., Schriftenreihe Flugzeug-Systemtechnik, Shaker Verlag.

[3] VAHL, A. 1998, *Interaktive Zuverlässigkeitsanalyse von Flugzeug-Systemarchitekturen*, Hamburg University of Technology, Diss., Fortschrittsberichte VDI, Volume 10, Issue 565, Düsseldorf.

[4] BIROLINI, A. 2006, *Reliability Engineering: Theory and Practice*, Springer Verlag, Berlin, Heidelberg.

[5] HEIDTMANN, K. D. 1989, *Smaller Sums of Disjoint Products by Subproduct Inversion*, IEEE Transactions on Reliability, Volume 38, Issue 3, pp. 305-311.

[6] REHAGE, D. 2009, *Zustandsmodellierung- und Zuverlässigkeitsanalyse fehlertoleranter Systemarchitekturen auf Basis Integrierter Modularer Avionik*, Technische Universität Hamburg-Harburg, Diss., Schriftenreihe Flugzeug-Systemtechnik, Shaker Verlag.

[7] REHAGE, D., CARL, U. B., VAHL, A. 2005, *Redundancy Management of Fault Tolerant Aircraft System Architectures - Reliability Synthesis and Analysis of Degraded System States*, Aerospace Science and Technology, Volume 9, Issue 4, pp. 337-347.

[8] RAKSCH, C., REHAGE, D., THIELECKE, F. 2009, *Multiobjective Optimization of Fault-Tolerant Aircraft Systems Considering System Degradation.* European Safety and Reliability Conference, Prag, Tschechien, 7.-10. September 2009.

[9] LIANG, Y.-C., SMITH, AL. E. 2004, *An Ant Colony Optimization Algorithm for the Redundancy Allocation Problem (RAP).* IEEE Transactions on Reliability, Volume 53, Issue 3, pp. 417 - 423.

[10] DIESNER, S. 2006, *Alternative leistungselektronische Schaltungskonzepte im PKW-Innenraum - Entwurf, Optimierung und Bewertung*, Technische Universität Dresden, Dissertation.

[11] NAKAGAWA, Y., NAKASHIMA, K., HATTORI, Y. 1978 , *Optimal Reliability Allocation by Branch-and-Bound Technique.* IEEE Transactions on Reliability vol. R-27 (1978), S. 31-38

[12] BUSACCA, P. G., MARSEGUERRA, M., ZIO, E. 2001, *Multi-objective Optimization by Genetic Algorithms: Application to Safety Systems*, Reliability Engineering and Safety Systems, Volume 72, pp. 59-74.

[13] DEB, K., AGRAWAL, S., PRATAB, A., MEYARIVAN, T. 2000, *A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II.* Parallel Problem Solving from Nature VI Conference, Paris, Frankreich, 2000, pp. 849-858.

[14] COELLO COELLO, A. C., LAMONT, G. B., VAN VELDHUIZEN, D. A. 2007, *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd Edition, Springer, New York, USA.

[15] TABOADA, H. A., ESPIRITU, J. F., COIT, D. W. 2008, *MOMS-GA: A Multi-Objective Multi-State Genetic Algorithm for System Reliability Optimization Design Problems.* IEEE Transactions on Reliability, Volume 57, Issue 1, pp. 182 - 191.