

DLR-IB-FA-BS-2021-90

**Analytische Bestimmung von
Restriktionsgradienten**

Studienarbeit

Autor Lennart Tönjes



DLR

**Deutsches Zentrum
für Luft- und Raumfahrt**

Institut für Faserverbundleichtbau und Adaptronik

DLR-IB-FA-BS-2021-90

Analytische Bestimmung von Restriktionsgradienten

Zugänglichkeit:

Stufe 1 Allgemein zugänglich: Der Interne Bericht wird elektronisch ohne Einschränkungen in ELIB abgelegt. Falls vorhanden, ist je ein gedrucktes Exemplar an die zuständige Standortbibliothek und an das zentrale Archiv abzugeben.

Braunschweig, 05, 2021

Abteilungsleiter: Prof. Dr.-Ing. Christian Hühne

Der Bericht umfasst: 66 Seiten

Autoren: Lennart Tönjes

Autor 2 / Betreuer: Sascha Dähne



DLR

Deutsches Zentrum
für Luft- und Raumfahrt



Analytische Bestimmung von Restriktionsgradienten

Studienarbeit

Technische Universität Braunschweig

Vorgelegt von:

B. Sc. Lennart Tönjes, 4684659

Betreut durch:

Prof. Dr.-Ing. Christian Hühne (IMA TU Braunschweig)

M. Sc. Hendrik Traub (IMA TU Braunschweig)

M. Sc. Sascha Dähne (DLR Braunschweig)

Inhaltsverzeichnis

Abbildungsverzeichnis	I
Nomenklatur	II
1 Einleitung	1
1.1 Stand der Technik	1
1.2 Ziel der Arbeit	1
1.3 Vorgehensweise	2
1.4 Lightworks	3
2 Grundlagen	4
2.1 Klassische Laminattheorie (KLT)	4
2.2 Lamination-Parameter	5
2.3 Strukturoptimierung.....	8
2.4 Entwurfsvariablen	10
2.5 Restriktionen	13
2.5.1 Rampenwinkel-Restriktion	14
2.5.2 Lamination-Parameter-Gültigkeitsbereich	15
2.5.3 Versagensrestriktionen	17
2.5.4 Modellierung der Versagensrestriktionen	18
2.6 Gradienten.....	19
2.6.1 Kettenregel	19
2.6.2 Finite Differenzen	19
2.6.3 Automatische Differenzierung	21
2.6.4 Autograd.....	22
2.6.5 Vergleich der Methoden.....	23
3 Implementierung	25
3.1 Aufstellung der Kettenregel	25
3.1.1 Strukturmodell	26
3.1.2 Finite-Elementen-Solver (FE-Solver).....	28
3.1.3 Restriktionsprozessor	28
3.2 Gradientenprozessor	33
4 Validierung	36
4.1 Geschwindigkeit	36
4.1.1 Haut-ABD-Matrix	37
4.1.2 Stringer-ABD-Matrix	38
4.1.3 Paneel-ABD-Matrix	41
4.1.4 Stringerabstand-ABD-Matrix	42

Inhaltsverzeichnis

4.1.5	Restriktionen	43
4.1.6	Vergleich globale und modulare Gradienten	47
4.2	Genauigkeit	49
5	Anwendung	52
6	Fazit	55
6.1	Zusammenfassung.....	55
6.2	Ausblick.....	56
	Literaturverzeichnis.....	58

Abbildungsverzeichnis

Abbildung 1: Auftretende Kraftflüsse bezogen auf die Breite im Scheibenelement (links) und Plattenelement (rechts)	4
Abbildung 2: Schematischer Aufbau eines symmetrischen Laminats	7
Abbildung 3: Optimierungsschleife	10
Abbildung 4: Aufbau eines versteiften Flugzeugflügels	11
Abbildung 5: Skizze eines I-Stringers	11
Abbildung 6: Dickenänderung zwischen zwei Paneelen	14
Abbildung 7: Skizze der Rampenwinkel-Restriktion	14
Abbildung 8: Lamination-Parameter-Gültigkeitsbereich	16
Abbildung 9: Verlauf der maximalen-Dehnungs-Restriktion unmodelliert und modelliert (links) und dessen Gradienten (rechts)	18
Abbildung 10: Relative Abweichung der finiten Differenzen über der Schrittweite	20
Abbildung 11: Grafische Darstellung der Vorwärtsmethode	21
Abbildung 12: Grafische Darstellung der Rückwärtsmethode	22
Abbildung 13: Vergleich der Rechenzeiten der Gradientenmethoden bei der Ableitung einer Sinus-Funktion	24
Abbildung 14: Modulare Aufteilung der Optimierungsprozesses	26
Abbildung 15: Darstellung des Stringerabstand- und offenen Bereiches in einer Paneelhaut	31
Abbildung 16: Ablauf des Optimierungsprozesses in lightworks	33
Abbildung 17: Implementierte Methoden im Gradientenprozessor	34
Abbildung 18: Rechenzeiten der Haut-Gradienten nach globalen Parametern (Paneel nicht versteift)	38
Abbildung 19: Rechenzeiten der Stringer-Gradienten nach globalen Parametern (Schwert-Paneel) ..	39
Abbildung 20: Rechenzeiten der Stringer-Gradienten nach globalen Parametern (I-Paneel)	40
Abbildung 21: Rechenzeiten der Paneel-Gradienten nach Haut-ABD-Matrix (links: Paneel nicht versteift, rechts: I-Paneel)	41
Abbildung 22: Rechenzeiten der Stringerabstands-Gradienten nach Haut- und Fuß-ABD-Matrix (I-Paneel)	42
Abbildung 23: Rechenzeiten der Rampenwinkel-Restriktions-Gradienten nach Hautdicke	43
Abbildung 24: Rechenzeiten der LP-Gültigkeitsbereichs-Gradienten nach LP (Paneel nicht versteift) ..	44
Abbildung 25: Rechenzeit der versagensrestriktions-Gradienten nach Spannung (links: Paneel nicht versteift, rechts: I-Paneel)	45
Abbildung 26: Rechenzeiten der Versagensrestriktions-Gradienten nach Haut-ABD-Matrix (I-Paneel)	46
Abbildung 27: Vergleich der Rechenzeiten zwischen globalen und modularen Gradienten (Paneel nicht versteift)	48
Abbildung 28: Vergleich der Rechenzeiten zwischen globalen und modularen Gradienten (I-Paneel) ..	48
Abbildung 29: Relative Abweichungen von finiten Differenzen ggü. automatischer Differentiation... ..	50
Abbildung 30: Konvergenzverlauf globaler und modularer Gradienten (Paneel nicht versteift)	52
Abbildung 31: Konvergenzverlauf globaler und modularer Gradienten (I-Paneel)	54

Nomenklatur

Abkürzungen

AD	Automatische Differentiation
CPACS	Common Parametric Aircraft Configuration Schema
DLR	Deutsches Luft- und Raumfahrtzentrum
FD	Finite Differenzen
FE	Finite Elemente
KLT	Klassische Laminattheorie
LP	Lamination-Parameter

Symbole

A	Scheiben- Nachgiebigkeitsmatrix
B	Koppel- Nachgiebigkeitsmatrix
D	Platten- Nachgiebigkeitsmatrix
d	Stringerabstand
f	Zielfunktion
g	Restriktionsfunktion
h	Schrittweite
h	Stringerhöhe
J	Jacobi-Matrix
$\hat{n}\hat{m}$	Kraftflüsse
Q	Steifigkeitsmatrix
t, T	Dicke
V	Lamination-Parameter
w	Fuß-, Flanschbreite
x	Globale Parameter
∂	Partielle Ableitung
ε, γ	Verzerrung
θ	Winkel
κ	Wölbung
σ, τ	Materialspannung

Indizes

A, B, D	Nachgiebigkeitsmatrizen
e	Stringerelement
f	Fuß
fl	Flansch
g	Global
i	Lage (Laminat)
lp	Lamination-Parameter
p	Paneel
pf	Versagensrestriktion (Part Failure)
r	Verteilt (Redistributed)
sk	Haut (Skin)
ss	Stringerabstand (Spacing-Span)
str	Stringer
ts	Rampenwinkel (Taper-Slope)
x, y, z	Globales Koordinatensystem

1 Einleitung

1.1 Stand der Technik

Beim Konstruieren, in welcher Branche auch immer, stellt sich ein Konstrukteur früher oder später die Frage: Wie kann ich meinen Entwurf noch weiter verbessern und effizienter gestalten? Im Flugzeugbau ist die Antwort dieser Frage die Masse des Flugzeuges möglichst kleinzuhalten. Während des Lösens dieses Problems kommen weitere Fragen auf. Was kann an der Konstruktion geändert werden, um die Masse zu verringern? Was muss dabei beachtet werden? Woher weiß ich, ob die gefundene Lösung wirklich die Optimale ist und nicht noch weiter verbessert werden kann?

Mit solchen Fragen beschäftigt sich die Strukturoptimierung. Durch die Optimierung wird eine optimale Lösung eines Problems gefunden, indem vorgegebene Parameter verändert werden. Ein solches Problem ist häufig die Minimierung der Masse von einer Struktur. Speziell im Flugzeugbau hat eine geringe Masse des Flugzeugs sehr viele Vorteile. Eine Reduktion der Masse wirkt sich direkt auf den Kraftstoffverbrauch aus und somit auch auf die freigesetzten Emissionen. Auch können mit einer geringeren Masse des Flugzeuges mehr Passagiere transportiert werden, was letztendlich ebenfalls auf die Emissionen auswirken kann. Gerade in der heutigen Zeit ist das Reduzieren der Emissionen ein sehr großes Thema und darf nicht vernachlässigt werden.

Es gibt mittlerweile etliche Algorithmen, die entwickelt wurden um eine optimale Lösung zu finden. Gerade für größere Optimierungsprobleme erweisen sich gradientenbasierte Optimierungsalgorithmen als sehr effizient. Für solche Algorithmen ist die Berechnung der Gradienten von Zielfunktion und Restriktionsfunktionen notwendig. Auch bei der Berechnung der Gradienten gibt es viele verschiedene Methoden, um dies zu bewerkstelligen.

1.2 Ziel der Arbeit

Das Deutsche Luft- und Raumfahrtzentrum hat eine Strukturoptimierungssoftware namens *lightworks* entwickelt, die speziell bei der Optimierung von Flugzeugflügeln angewandt wird. Bei der Optimierung von Flugzeugflügeln können sehr viele Parameter einen Einfluss auf dessen optimale Auslegung haben. Dementsprechend kann der Rechenaufwand vom Optimierungsprozess sehr groß werden. Gradientenbasierte Strukturoptimierungsalgorithmen können dabei helfen den Aufwand zu reduzieren. Dafür sind allerdings Gradienten für die Zielfunktion und Restriktionsfunktionen notwendig. Diese werden nach jetzigem Stand der Arbeit mit finiten Differenzen bestimmt. Nachteil dabei ist, dass diese Methode sehr rechenintensiv bei hoher Anzahl an Designparametern werden kann. Außerdem liefert sie nur eine Approximation des tatsächlichen Gradienten und kann daher fehlerbehaftet sein.

Ziel dieser Arbeit ist es, effizientere und genauere Methoden der Gradientenberechnung zu implementieren. Dabei wird der komplette Prozess der Gradientenberechnung in mehrere Teilprozesse unterteilt. In jedem Teilprozess können die jeweiligen Gradienten berechnet und zum Schluss wieder mithilfe der Kettenregel zu dem kompletten Prozess zusammengefügt werden. Das Endergebnis soll mit dem des Referenzprozesses, welcher mit finiten Differenzen berechnet wurde, übereinstimmen. Darüber hinaus wird die Rechenzeit der neu implementierten Methode mit der des Referenzprozesses verglichen, wobei eine Zeitreduktion durch die neuen Methoden erwartet wird.

1.3 Vorgehensweise

Zunächst werden in dieser Arbeit die nötigen Grundlagen geklärt, welche zum Verständnis nötig sind. Dazu gehört, wie sich das Material eines Flügels zusammensetzt. Weiterhin wird eine kurze Einführung in den Ablauf einer Strukturoptimierung gegeben. Im Anschluss werden die Parameter des Flügels vorgestellt, welche sich bei der Strukturoptimierung verändern lassen, um die Gesamtmasse des Flügels zu minimieren. Dabei wird auch auf den Aufbau eines Flügels eingegangen. Anschließend werden die Restriktionen vorgestellt, die während der Optimierung eingehalten werden sollen und die es abzuleiten gilt. Weiterhin werden die Berechnungsmethoden für die Gradienten eingeführt und auf deren Vor- und Nachteile eingegangen. Zusätzlich wird ein Versuch mit einer einfachen Funktion durchgeführt, welche es gilt abzuleiten, um den Unterschied der Methoden zu demonstrieren.

Im dritten Kapitel wird der Optimierungsprozess vom Strukturmodell bis zur Berechnung der Restriktionen in die einzelnen Module zerlegt. Dadurch ist es möglich jedes Modul separat zu betrachten und die Rechenvorschriften für die dazugehörigen partiellen Gradienten herzuleiten. Für das Herleiten wird die Kettenregel eingesetzt, welche im vorherigen Kapitel kurz besprochen wird. Die Berechnung mit der Kettenregel stellt eine neue Methode der Gradientenberechnung dar, die die bisher implementierte Methode der Gradientenberechnung ersetzen soll. Anschließend wird beschrieben, wie die neue Methode in dem jetzigen Optimierungsprozess die bisherige Methode ersetzen wird.

Im vierten Kapitel wird die neu implementierte Methode der alten Methode hinsichtlich Geschwindigkeit und Genauigkeit gegenübergestellt. Dafür werden mehrere Versuche durchgeführt und im Anschluss diskutiert. Ziel dieser Versuche ist zu ermitteln, ob durch die neu implementierte Methode die Geschwindigkeit und Genauigkeit der berechneten Gradienten verbessert werden konnte.

Zum Schluss findet die neue Methode Anwendung in einer Ersatz-Optimierung. Die gleiche Optimierung wird ebenfalls für die herkömmliche Methode durchgeführt. Die resultierenden Konvergenzverläufe und Rechenzeiten werden im Anschluss miteinander verglichen. Daraus soll sich der Nutzen der neu implementierten Methode der Gradientenberechnung herauskristallisieren.

1.4 Lightworks

Lightworks wird vom Deutschen Luft- und Raumfahrtzentrum (DLR) entwickelt. Das Ziel dieser Software ist es, die Haut eines Flugzeugflügels unter Einhaltung bestimmter Restriktionen zu optimieren und so die Gesamtmasse zu minimieren. Alle benötigten strukturellen Informationen des Flügels werden aus einer XML-basierten CPACS-Datei gelesen. CPACS steht für Common Parametric Aircraft Configuration Schema und wurde ebenfalls vom DLR entwickelt. Der Flügel wird in der CPACS-Datei in mehrere Segmente unterteilt. Dort sind die Materialeigenschaften der Schalen, Rippen und Stege der einzelnen Segmente definiert und können als Entwurfsvariablen für die Optimierung genutzt werden [1].

Hier setzt nun *lightworks* an. Durch die Hinzunahme eines unveränderten Strukturmodells des zu optimierenden Flügels, wird die Optimierung gestartet. Durch verschiedene Optimierungsalgorithmen, die in *lightworks* implementiert sind, werden bestimmte Entwurfsvariablen des Strukturmodells so bestimmt, dass sich eine optimale Masse einstellt. Dabei müssen bestimmte Restriktionen eingehalten werden, sodass der Flügel durch in der Praxis auftretende Lasten nicht versagt. Die nun optimierten Entwurfsvariablen sollen dann im letzten Schritt wieder in die CPACS-Datei geschrieben werden und die ursprünglichen Parameter ersetzen.

2 Grundlagen

2.1 Klassische Laminattheorie (KLT)

Faserverbundmaterialien haben großes Potenzial die Masse eines Flügels zu reduzieren. Eine Eigenschaft von Fasern ist, dass sie keine Kräfte in Querrichtung aufnehmen können. Daher werden üblicherweise mehrere Einzellagen übereinandergelegt, wobei die Fasern verschiedene Ausrichtungen haben, damit das Material ebenfalls so Quer- und Schubkräfte aufnehmen kann. Diese Konstellation wird auch als Mehrschichtverbund bezeichnet. Bei der KLT wird der Mehrschichtverbund als Kombination aus Scheiben- und Plattenelementen angesehen. Scheibenelemente werden nur parallel und symmetrisch zur Mittelebene belastet, wohingegen Plattenelemente auch senkrecht zur Mittelebene durch Querkräfte und Momente belastet werden können. In Abbildung 1 sind die auftretenden Kräfte und Momente beim Scheiben- bzw. Plattenelement bezogen auf die Breite des Elements dargestellt. Sie werden daher auch als Kraft- bzw. Momentfluss bezeichnet. Bei den Scheibenelementen treten durch diese Kräfte Verzerrungen ε bzw. γ auf und im Plattenelement Wölbungen κ [2].

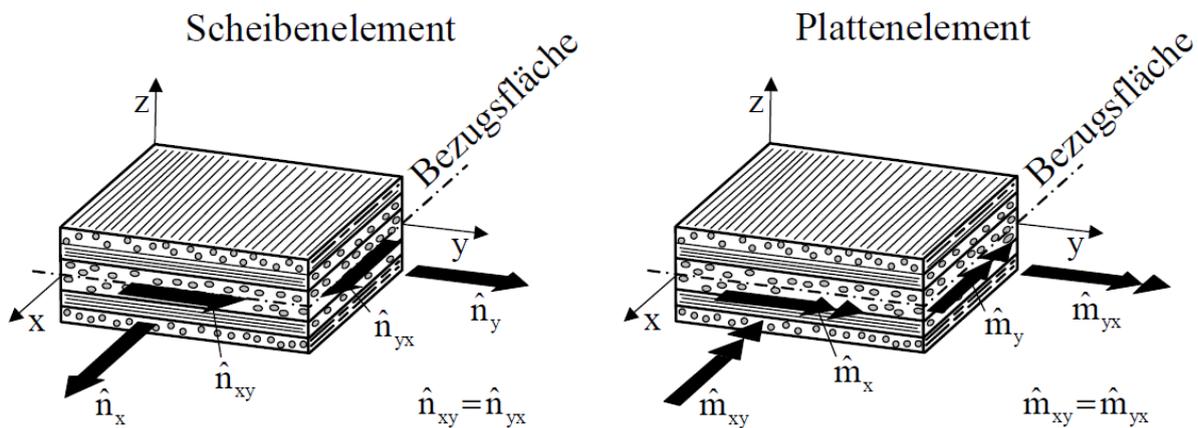


Abbildung 1: Auftretende Kraftflüsse bezogen auf die Breite im Scheibenelement (links) und Plattenelement (rechts) [Schürmann]

Den Zusammenhang zwischen den Kraftflüssen (\hat{n}, \hat{m}) bzw. Spannungen (σ, τ) und den Verzerrungen und Wölbungen ($\varepsilon, \gamma, \kappa$), stellt die ABD-Matrix her. Sie definiert somit das Elastizitätsgesetz des kombinierten Scheiben- Plattenelements.

$$\begin{pmatrix} \hat{n}_x \\ \hat{n}_y \\ \hat{n}_{xy} \\ \hat{m}_x \\ \hat{m}_y \\ \hat{m}_{xy} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & A_{16} & B_{11} & B_{12} & B_{16} \\ A_{21} & A_{22} & A_{26} & B_{21} & B_{22} & B_{26} \\ A_{61} & A_{62} & A_{66} & B_{61} & B_{62} & B_{66} \\ B_{11} & B_{12} & B_{16} & D_{11} & D_{12} & D_{16} \\ B_{21} & B_{22} & B_{26} & D_{21} & D_{22} & D_{26} \\ B_{61} & B_{62} & B_{66} & D_{61} & D_{62} & D_{66} \end{pmatrix} \begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \\ \kappa_x \\ \kappa_y \\ \kappa_{xy} \end{pmatrix} = \sum_{k=1}^n \int_{z_{k-1}}^{z_k} \begin{pmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \\ \sigma_x \cdot z \\ \sigma_y \cdot z \\ \tau_{xy} \cdot z \end{pmatrix} dz \quad (2.1)$$

Für die Gesamtspannung können die Spannungen der Einzelschichten aufsummiert werden. Hier wird der Abstand der Einzelschicht zur Bezugsebene als z_k bezeichnet. Die Bezugsebene kann beliebig gewählt werden. Bei mittensymmetrischen Aufbauten wird empfohlen die Bezugsebene der Mittelebene gleichzusetzen.

Die ABD-Matrix besteht aus drei Komponenten. Die A-Matrix beschreibt die Scheibensteifigkeitsmatrix mit

$$A_{ij} = \sum_{k=1}^n Q_{ij,k}(z_k - z_{k-1}) = \sum_{k=1}^n Q_{ij,k} \cdot t_k \quad (2.2)$$

Da Scheibenelemente nur Kräfte parallel zur Bezugsebene aufnehmen können, ist die Wahl der Bezugsebene hier irrelevant. Somit sind die A-Komponenten nur von den Dicken der Einzelschichten und den jeweiligen Steifigkeiten abhängig.

Die D-Matrix steht für die Plattensteifigkeitsmatrix. Die einzelnen Komponenten werden durch

$$D_{ij} = \frac{1}{3} \sum_{k=1}^n Q_{ij,k}(z_k^3 - z_{k-1}^3) = \sum_{k=1}^n Q_{ij,k} \left(\frac{t_k^3}{12} + t_k \left(z_k - \frac{t_k}{2} \right)^2 \right) \quad (2.3)$$

berechnet. Anders als bei der Scheibensteifigkeitsmatrix, hat hier die Wahl der Bezugsebene, zusätzlich zu den Dicken und Steifigkeiten, einen Einfluss auf die Einträge der D-Matrix.

Da es sich bei der KLT um kombinierte Scheiben- und Plattenelemente handelt, wird ebenfalls die Koppelmatrix B benötigt. Dessen Komponenten sind ebenfalls von der Bezugsfläche abhängig und werden wie folgt berechnet:

$$B_{ij} = \frac{1}{2} \sum_{k=1}^n Q_{ij,k}(z_k^2 - z_{k-1}^2) = \sum_{k=1}^n Q_{ij,k} \cdot t_k \left(z_k - \frac{t_k}{2} \right) \quad (2.4)$$

[2]

2.2 Lamination-Parameter

Wie in Kapitel 2.1 beschrieben besteht ein Laminat aus mehreren übereinanderliegenden Schichten, wobei die Fasern der Schichten jegliche Ausrichtungen haben können. Hinzu kommt, dass beliebig viele Schichten aufaddiert werden können, wobei jede Schicht eine andere Dicke besitzen kann. Das erlaubt

große Freiheiten beim Auslegen eines Laminats. Für den Optimierungsprozess bedeutet dies allerdings einen Nachteil, da deshalb mehr Designvariablen in Betracht gezogen werden müssen. Um die Steifigkeit eines Laminats optimieren zu können, werden kontinuierliche Parameter benötigt. Aus diesem Grund wurden die Lamination-Parameter (LP) eingeführt, mit denen es möglich ist, die ABD-Matrix kontinuierlich zu beschreiben.

Die Materialkonstanten für eine orthotropische Einzelschicht lauten:

$$\begin{aligned}
 U_1 &= \frac{1}{8} (3Q_{11} + 3Q_{22} + 2Q_{12} + 4Q_{66}) \\
 U_2 &= \frac{1}{2} (Q_{11} - Q_{12}) \\
 U_3 &= \frac{1}{8} (Q_{11} + Q_{22} - 2Q_{12} - 4Q_{66}) \\
 U_4 &= \frac{1}{8} (Q_{11} + Q_{22} + 6Q_{12} - 4Q_{66}) \\
 U_5 &= \frac{1}{8} (Q_{11} + Q_{22} - 2Q_{12} + 4Q_{66})
 \end{aligned} \tag{2.5}$$

Die Steifigkeiten können mithilfe folgender Formeln, in Abhängigkeit vom Lagenwinkel θ und den Materialkonstanten transformiert werden.

$$\begin{aligned}
 Q'_{11} &= U_1 + U_2 \cos(2\theta) + U_3 \cos(4\theta) \\
 Q'_{22} &= U_1 - U_2 \cos(2\theta) + U_3 \cos(4\theta) \\
 Q'_{12} &= U_4 - U_3 \cos(4\theta) \\
 Q'_{66} &= U_5 - U_3 \cos(4\theta) \\
 Q'_{16} &= U_6 \cos(2\theta) - \frac{1}{2} U_2 \sin(2\theta) - U_3 \sin(4\theta) \\
 Q'_{26} &= U_6 \cos(2\theta) - \frac{1}{2} U_2 \sin(2\theta) + U_3 \sin(4\theta)
 \end{aligned} \tag{2.5}$$

Werden die transformierten Steifigkeiten beispielsweise für A_{11} in Gleichung 2.2 eingesetzt, erhält man:

$$A_{11} = \sum_{k=1}^n (U_1 + U_2 \cos(2\theta_k) + U_3 \cos(4\theta_k)) (z_k - z_{k-1}) \tag{2.6}$$

Aus dieser Gleichung können nun die LP definiert werden, indem die Sinus- und Kosinus-Terme substituiert werden. Es ergibt sich:

$$A_{11} = T(U_1 + U_2 \cdot V_1^A + U_3 \cdot V_2^A) \tag{2.7}$$

Die gleiche Vorgehensweise kann auf jeden der 36 Einträge der ABD-Matrix angewendet werden [3].

Eine Zusammenfassung ist in Gleichung 2.9 zu finden.

$$\begin{pmatrix} A_{11}, B_{11}, D_{11} \\ A_{22}, B_{22}, D_{22} \\ A_{12}, B_{12}, D_{12} \\ A_{66}, B_{66}, D_{66} \\ A_{16}, B_{16}, D_{16} \\ A_{26}, B_{26}, D_{26} \end{pmatrix} = \left(T, \frac{T^2}{4}, \frac{T^3}{12} \right) \begin{bmatrix} 1 & V_1^{A,B,D} & V_2^{A,B,D} & 0 & 0 \\ 1 & -V_1^{A,B,D} & V_2^{A,B,D} & 0 & 0 \\ 0 & 0 & -V_2^{A,B,D} & 1 & 0 \\ 0 & 0 & -V_2^{A,B,D} & 0 & 1 \\ 0 & V_3^{A,B,D}/2 & V_4^{A,B,D} & 0 & 0 \\ 0 & V_3^{A,B,D}/2 & -V_4^{A,B,D} & 0 & 0 \end{bmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{pmatrix} \quad (2.9)$$

Somit ergeben sich für jede Teilmatrix in der ABD-Matrix vier LP. Sie sind wie folgt definiert:

$$\begin{aligned} V_{[1,2,3,4]}^A &= \frac{1}{T} \sum_{k=1}^n (z_k - z_{k-1}) \cdot [\cos(2\theta_k), \cos(4\theta_k), \sin(2\theta_k), \sin(4\theta_k)] \\ V_{[1,2,3,4]}^B &= \frac{1}{T^2} \sum_{k=1}^n (z_k^2 - z_{k-1}^2) \cdot [\cos(2\theta_k), \cos(4\theta_k), \sin(2\theta_k), \sin(4\theta_k)] \\ V_{[1,2,3,4]}^D &= \frac{1}{T^3} \sum_{k=1}^n (z_k^3 - z_{k-1}^3) \cdot [\cos(2\theta_k), \cos(4\theta_k), \sin(2\theta_k), \sin(4\theta_k)] \end{aligned} \quad (2.8)$$

Die LP können Werte zwischen ± 1 annehmen. Die zwölf Parameter können auf insgesamt fünf Werte reduziert werden, wenn es sich bei dem Laminat um ein symmetrisch balanciertes Laminat handelt, welche häufig im Flugzeugbau eingesetzt wird. Hierbei werden ausschließlich Fasern in 0° , $\pm 45^\circ$ und 90° Richtung eingesetzt. Die Dicken der 45° -Ebenen werden gleich groß gewählt. Dadurch entstehen zwei Symmetrieebenen. Die dritte Symmetrieebene entsteht durch eine mittensymmetrische Anordnung der Ebenen. Ein solches Laminat ist schemenhaft in Abbildung 2 dargestellt.

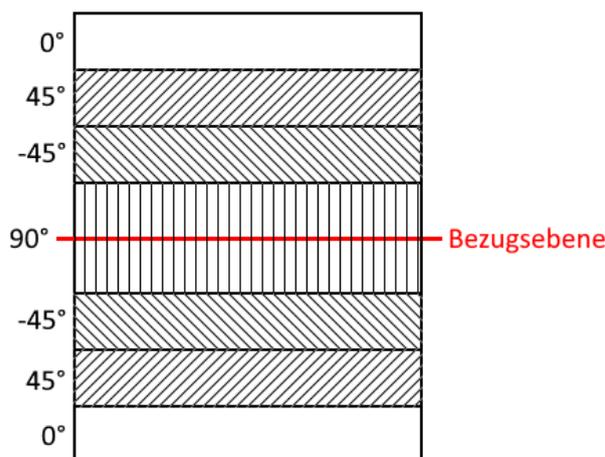


Abbildung 2: Schematischer Aufbau eines symmetrischen Laminats

Ein symmetrisch balanciertes Laminat hat den Vorteil, dass sich die Koppelmatrix komplett aufhebt und somit auch die V^B -Parameter [2].

Die Parameter V_4^A und V_4^D heben sich ebenfalls auf. Die Sinusterme werden zu null unabhängig davon, welche Schicht betrachtet wird.

Bei V_3^A und V_3^D gibt es Unterschiede, die zu beachten sind. Für die 0° - und 90° -Schichten werden die Sinusterme zu null und entfallen somit. Da es genauso viele -45° - wie 45° -Schichten gibt, bei denen die Sinusterme ± 1 ergeben, heben diese sich gegenseitig auf. Das ist allerdings nur bei den V^A -Parametern der Fall, da hier lineare Dicken vorliegen, die bei beiden Schichten gleich groß sind. Beim Betrachten der V^D -Parameter ist auffällig, dass die V_3^D -Parameter trotz der Symmetrie nicht vernachlässigbar sind, da hier die Dicken kubisch statt linear sind. Es ist nun relevant, wie weit die Schichten von der Bezugsebene entfernt sind. Dadurch ergeben sich für die $\pm 45^\circ$ -Schichten unterschiedliche Werte und heben sich somit nicht gegenseitig auf.

Es bleiben somit nur noch die $V_{[1,2]}^A$ - und die $V_{[1,2,3]}^D$ -Parameter bestehen. Der symmetrische Aufbau des Laminats ist für die Optimierung vorteilhaft, da somit weniger Entwurfsvariablen ausgewertet werden müssen [4].

2.3 Strukturoptimierung

In den meisten Anwendungsfällen ist das Ziel einer Strukturoptimierung die Masse des zu optimierenden Systems zu minimieren und dabei die zulässigen Spannungen, Verformungen sowie andere selbst definierte Restriktionen nicht zu überschreiten. Die Masse des Systems wird deshalb als Zielfunktion bezeichnet. Als Ausgang jeder Strukturoptimierung dient ein Analysemodell, welches die Struktur des Systems zusammenfasst. Es ist zu entscheiden, welche Parameter verändert bzw. optimiert werden können. Solche Parameter werden auch als Entwurfsvariablen bezeichnet. Weiterhin muss beachtet werden, welche Restriktionen eingehalten werden müssen. Oft spielen auch geometrische Restriktionen eine Rolle. Die Restriktionen definieren den zulässigen Bereich der Zielfunktion. Die Optimierung durchläuft nun einen iterativen Prozess, indem die Entwurfsvariablen mit bestimmten Optimierungsalgorithmen Schritt für Schritt angepasst werden. Nach jedem Iterationsschritt wird überprüft, ob ein Optimum gefunden wurde. Ist dies nicht der Fall, werden die Entwurfsvariablen in einem weiteren Iterationsschritt weiter angepasst. Zur Überprüfung, ob ein Optimum gefunden wurde, dienen bestimmte Abbruchkriterien. Abbildung 3 zeigt einen solchen Prozess. Im Folgenden werden die gängigen Abbruchkriterien kurz erläutert.

Beim ersten Kriterium wird der Gradient der Zielfunktion nach den Entwurfsvariablen gebildet. Die Werte der aktuellen Entwurfsvariablen werden dann in den Gradienten eingesetzt und für jede

Variable aufsummiert. Geht diese Summe gegen null, so befindet sich in diesem Punkt ein Extremum. Ist die Zielfunktion in dem Bereich konvex, so kann davon ausgegangen werden, dass es sich bei dem Extremum um ein Minimum handelt.

1. Abbruchkriterium:

$$\sum_i \left| \left(\frac{\partial f(x)}{\partial x_i} \right) \right| < \varepsilon \quad (2.9)$$

Das ε sollte dabei möglichst klein gewählt werden. Größenordnungen von 10^{-5} bis 10^{-8} sind dabei üblich.

Beim nächsten Abbruchkriterium wird der Fortschritt der Optimierung überprüft. Dabei wird die relative Änderungsrate der Zielfunktion des aktuellen Iterationsschritts und des letzten Iterationsschritts gebildet. Ist diese genügend klein, wird die Optimierung an dieser Stelle beendet.

2. Abbruchkriterium:

$$\frac{f(x^{(k)}) - f(x^{(k+1)})}{\max\{|f(x^{(k+1)})|, 10^{-8}\}} < \varepsilon_{rel} \quad (2.10)$$

Auch hier sollte ε_{rel} möglichst klein gewählt werden.

Das dritte Abbruchkriterium dient zur Sicherheit, falls kein Optimum gefunden werden kann oder der Prozess zu aufwendig ist. Wenn das der Fall ist, muss überprüft werden, ob der verwendete Optimierungsalgorithmus für dieses Optimierungsproblem geeignet ist. Hier wird die Anzahl der Iterationen k auf ein Maximum begrenzt.

3. Abbruchkriterium:

$$k = k_{max} \quad (2.13)$$

Das letzte Kriterium wird bei restringierten Optimierungsproblemen benötigt. Dieses stellt sicher, dass alle Restriktionen erfüllt sind [5].

4. Abbruchkriterium:

$$\vec{\nabla} f(\vec{x}) + \sum_j^m \lambda_j \vec{\nabla} g_j(\vec{x}) = \vec{0} \quad (2.11)$$

Grafisch kann die Gleichung so interpretiert werden, dass der steilste Anstieg der Zielfunktion und steilster Anstieg der Restriktion genau gegeneinander gerichtet sind. Dieses Phänomen ist nur in einem lokalen Optimum möglich. Somit befindet sich der aktuelle Systemzustand genau auf einer Restriktionsgrenze [6].

Anzumerken ist, dass auch wenn eines der obigen Abbruchkriterien erreicht wurde, noch nicht sichergestellt ist, ob es sich bei dem gefundenen Optimum um das globale Minimum handelt oder nur ein lokales Minimum des abgesuchten Bereiches ist.

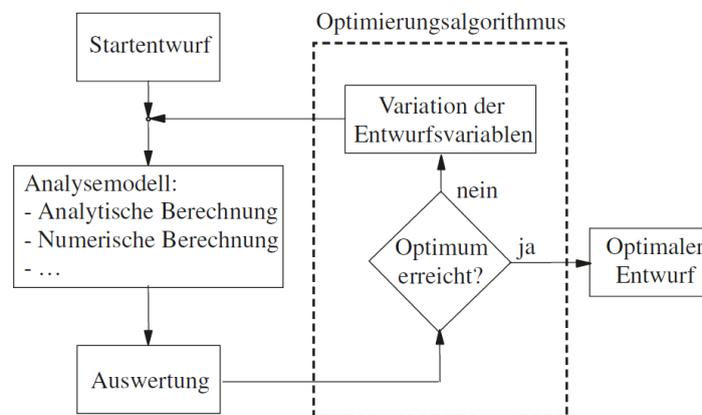


Abbildung 3: Optimierungsschleife [5]

2.4 Entwurfsvariablen

Um die optimale Masse des Systems zu finden, müssen zunächst Entwurfsvariablen gewählt werden, welche im Laufe der Optimierung angepasst werden sollen. Eine Möglichkeit die Masse z. B. eines Flügels zu minimieren ist es, die Wandstärken der einzelnen Materiallagen zu verringern. Aber auch andere Variablen können angepasst werden. Diese müssen nicht unbedingt Einfluss auf die Zielfunktion haben, sondern können auch nur die Restriktionsfunktionen beeinflussen. In diesem Kapitel wird zunächst der grobe Aufbau eines Flügels dargestellt und anschließend die Entwurfsvariablen vorgestellt, die für die Strukturoptimierung eines Flügels relevant sind.

Eine Flügelstruktur besteht im Grunde aus Rippen und Holmen, welche das Skelett des Flügels bilden. Auf diesem Skelett wird eine Haut gespannt. Diese Haut wird in den meisten Fällen zusätzlich durch Stringer versteift. Abbildung 4 stellt ein versteiftes Flügelsegment dar.

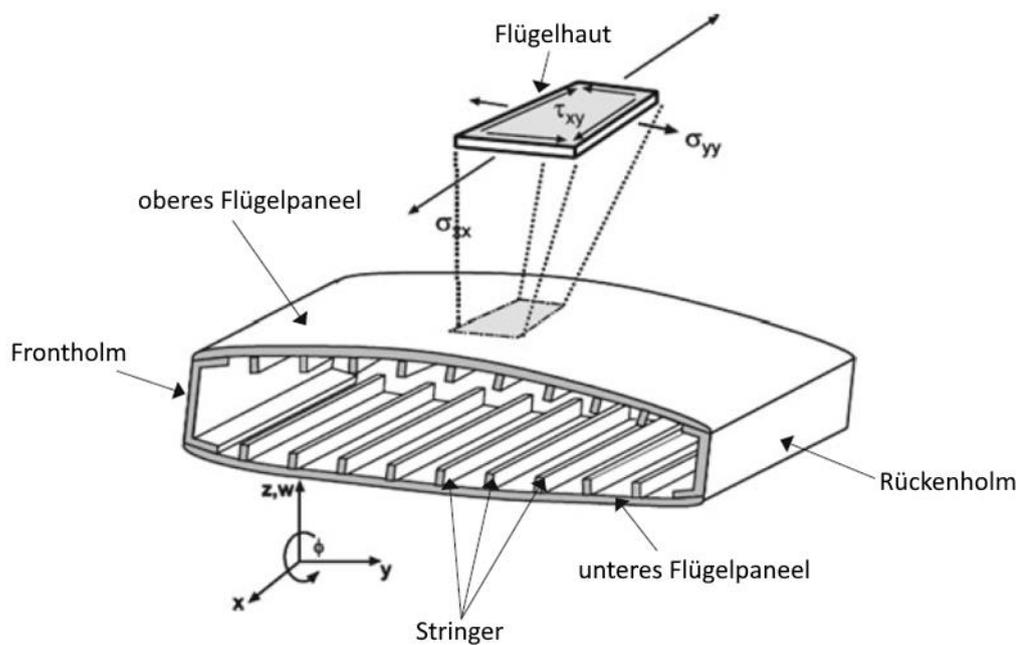


Abbildung 4: Aufbau eines Flugzeugflügels [7]

Es wird zwischen versteiften Paneelen, bestehend aus Haut und Stringer und nicht versteiften Paneelen, welche nur aus Haut bestehen, unterschieden. Je nach Profil besteht ein Stringer aus bis zu drei Elementen: Steg, Fuß und Flansch. Abbildung 5 stellt die drei Elemente am Beispiel eines I-Profils dar.

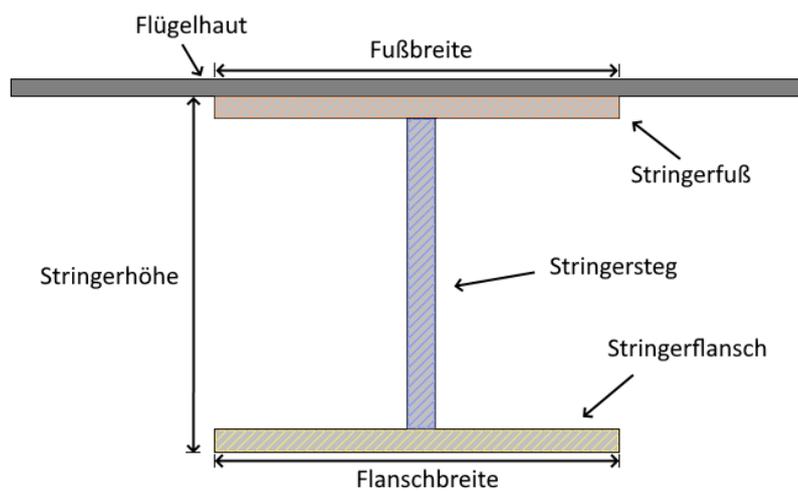


Abbildung 5: Skizze eines I-Stringers

Als erste Entwurfsvariable kann die Gesamtdicke der Flügelhaut gewählt werden. Bei einem versteiften Paneel können zusätzlich zur Gesamtdicke der Flügelhaut genauso die Gesamtdicken der einzelnen Stringerelemente angepasst werden. Da es sich bei der Flügelhaut meistens um Laminat handelt, können auch statt der Gesamtdicke die Einzeldicken der Laminatschichten als Entwurfsvariablen gewählt werden. Die Schichtwinkel der einzelnen Schichten können ebenfalls für die Optimierung als Entwurfsvariablen verändert werden. Im Flugzeugbau werden allerdings meist feste Schichtwinkel vorgeschrieben und daher nur selten angepasst.

Wie in Kapitel 2.2 gezeigt, kann ein Laminat auch mithilfe von LP beschrieben werden. Diese eignen sich sehr gut zur Optimierung, da diese die Schichtdicken und Schichtwinkel zusammenfassen. Mit den LP kommen zusätzlich zwölf weitere Parameter als Entwurfsvariablen infrage. Handelt es sich bei dem Laminat um ein symmetrisch balanciertes Laminat, verringert sich die Anzahl von zwölf auf fünf Parameter.

Alle bisher genannten Parameter befassen sich ausschließlich mit der Flügelhaut bzw. mit der Stringerhaut. Zusätzlich kann die Geometrie der Stringer verändert werden. Dazu gehören die Stringerhöhe sowie -winkel und die Fuß- und Flanschbreite, welche in Abbildung 5 dargestellt sind. Nicht in der Abbildung eingezeichnet, ist der Stringerabstand, der den Abstand zwischen den Stringer beschreibt und ebenfalls angepasst werden kann.

In Tabelle 1 sind die möglichen Entwurfsvariablen aufgelistet. Sie soll aufzeigen wie viele zusätzliche Parameter bei der Optimierung in Betracht gezogen werden, wenn nach den jeweiligen Parametern optimiert wird. Es wird dabei zwischen Schwert-, T- und I-Stringer unterschieden. Schwert-Stringer bestehen ausschließlich aus einem Steg. T-Stringer besitzen zusätzlich zum Steg einen Fuß, haben aber keinen Flansch, wie es beim I-Stringer der Fall ist.

Tabelle 1: Anzahl der Parameter für verschiedene Paneel-Arten mit n -Schichten

Parameter	Schwert-Stringer	T-Stringer	I-Stringer
Hautdicke	1	1	1
Stringerelementdicke	1	2	3
Schichtdicke	2n	3n	4n
Faserwinkel	2n	3n	4n
Lamination-Parameter (symmetrisch balanciert)	24 (10)	36 (15)	48 (20)
Stringerhöhe	1	1	1
Stringerabstand	1	1	1
Stringerwinkel	1	1	1
Fußbreite	0	1	1
Flanschbreite	0	0	1

Anzumerken ist, dass hier davon ausgegangen wird, dass für die Haut und den Stringerelementen unterschiedliche Lamine verwendet werden. Gibt es Duplikate bei den Elementen, werden diese, damit sie nicht doppelt ausgewertet werden, in der Optimierung als das selbe Material betrachtet. Somit wird die Laufzeit der Optimierung reduziert

Um zu veranschaulichen, dass das Optimierungsmodell zur Optimierung der Haut eines Flügels sehr schnell sehr komplex werden kann, soll folgendes Gedankenexperiment dienen.

Es liegt ein versteiftes Paneel mit I-Stringer vor. Es sollen nun Haut- und Stringerelementdicken, LP, sowie jegliche Parameter, die zur Stringergeometrie gehören, optimiert werden. Vergleicht man diesen Fall mit der obigen Tabelle, werden 57 verschiedene Parameter optimiert, wenn es sich um symmetrisch balancierte Lamine handelt, 29 Parameter. Eine zu optimierende Struktur besteht jedoch meist aus mehreren Paneelen. Angenommen in dieser Region befinden sich 20 verschiedene Paneele und alle besitzen unterschiedliche Lamine, dann erhöht sich die Zahl der Parameter auf 1140 Parameter. Anzumerken ist, dass es sich nicht bei allen Paneelen in einer Optimierungsregion um Paneele mit I-Profil-Stringern handeln muss. Bei Schwert- oder T-Stringer würden entsprechend weniger Parameter hinzukommen. Außerdem werden oftmals bei den Paneelen die gleichen Lamine mehrmals verwendet und wie bereits angemerkt, werden diese nur einmal ausgewertet. Bei der Zahl oben handelt es sich also um ein „worst-case“-Szenario. Allerdings werden bei der Optimierung auch mehrere Lastfälle betrachtet. Die Analyse wird also mehrmals ausgeführt. Zusätzlich können in einem Flügel hunderte Optimierungsregionen vorhanden sein. Bei der Optimierung des Flügels muss also jede Region ausgewertet werden, mit unterschiedlichen Lastfällen. Dabei kann es vorkommen, dass jede Region mehrere hundert Entwurfsvariablen besitzt.

Bei der Menge an Parametern, die für den Optimierungsalgorithmus zu beachten sind, kann es vorkommen, dass der komplette Optimierungsprozess mehrere Tage dauert. Aus diesem Grund muss unbedingt darauf geachtet werden, dass der komplette Optimierungsprozess mit Gradientenberechnung, Festigkeitsanalyse und der eigentlichen Optimierung möglichst effizient implementiert ist.

2.5 Restriktionen

Restriktionen dienen zum Begrenzen des Bereiches, in welchem sich ein zulässiges Optimum der Zielfunktion befindet. In diesem Kapitel wird geklärt, welche Restriktionen definiert sind, die den zulässigen Bereich eines Flügels definieren.

2.5.1 Rampenwinkel-Restriktion

Es kommt oft vor, dass zwei benachbarte Paneele eine unterschiedliche Hautdicke besitzen. Dabei besitzt die dünnere Haut weniger Schichten als die dickere. Eine Gestaltungsrichtlinie von Laminaten besagt, dass alle Schichten des dünneren Laminats auch im dickeren, benachbarten Laminat vorkommen müssen. Dadurch gehen die Schichten, die in beiden Laminaten verwendet werden, ineinander über. Die Schichten, welche nur im dickeren Laminat Verwendung finden, werden beim Übergang zum dünneren Laminat abgeschnitten. Dadurch entsteht ein Rampenbereich (*Taper Section*) zwischen den beiden Paneelen, wie in Abbildung 6 ersichtlich [8].

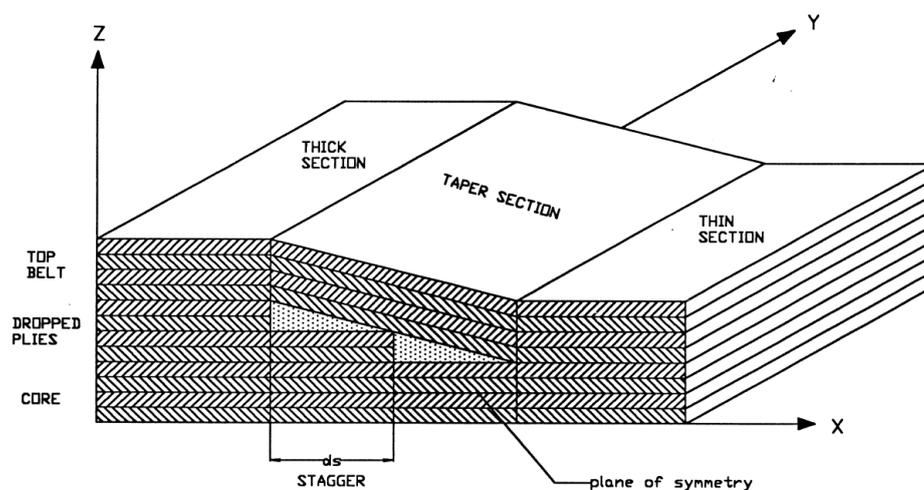


Abbildung 6: Dickenänderung zwischen zwei Paneelen [8]

Diese Dickenänderungen und damit einhergehend die Steigung des Rampenbereichs sollten auf ein Minimum begrenzt werden. Zu starke Dickenänderungen können zu ungleichmäßigen Steifigkeiten des Laminats im Rampenbereich führen und damit ein Versagen desselben hervorrufen. Es wird empfohlen die Steigung der Rampe in Lastrichtung unter 5 % und in anderen Richtungen unter 10 % zu halten [9].

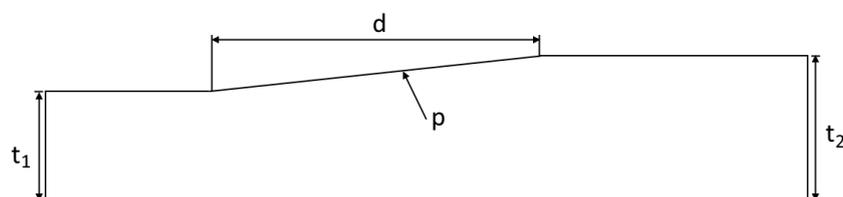


Abbildung 7: Skizze der Rampenwinkel-Restriktion

Mithilfe von Abbildung 7 kann die Restriktionsfunktion hergeleitet werden. Es wird verlangt, dass die tatsächliche Steigung zwischen Paneel 1 mit der Dicke t_1 und Paneel 2 mit der Dicke t_2 nicht die festgelegte Steigung p_{max} übersteigt. Der Rampenbereich hat eine Länge von d . Somit berechnet sich die tatsächliche Steigung p mit:

$$p = \frac{|t_1 - t_2|}{d} \quad (2.15)$$

Die Restriktionsfunktion g wird gebildet, indem das Verhältnis von der tatsächlichen Steigung p zur maximalen Steigung p_{max} gebildet wird. Bei den Optimierungsalgorithmen wird der zulässige Bereich als der Bereich definiert, in welchem die Restriktionsfunktion kleiner null ist. Aus diesem Grund muss das Verhältnis mit eins subtrahiert werden, damit diese Bedingung erfüllt ist. Somit ergibt sich für die Restriktionsfunktion:

$$g = \frac{|t_1 - t_2|}{d \cdot p_{max}} - 1 \quad (2.16)$$

Für den zulässigen Bereich gilt:

$$g \leq 0 \quad (2.17)$$

2.5.2 Lamination-Parameter-Gültigkeitsbereich

Die Lamination-Parameter (Kapitel 2.2) können nicht beliebig gewählt werden. Es sind nur bestimmte Kombinationen von LP möglich. Der zulässige Bereich kann grafisch beschrieben werden. Abbildung 8 zeigt einen solchen zulässigen Bereich an dem Beispiel von $V_{[1,2]}^A$ -Parametern für den Fall, dass nur 90° -, 0° -, $\pm 45^\circ$ -, $\pm 30^\circ$ - und $\pm 60^\circ$ -Schichtwinkel verwendet werden. Alle möglichen Kombinationen von diesen LP spannen eine minimale konvexe Hülle auf. Eine Eigenschaft von konvexen Hüllen ist, dass jeder Punkt in diesem Bereich durch eine Gerade miteinander verbunden werden kann, ohne dass diese die konvexe Hülle verlässt. Die konvexe Hülle kann durch Hyperebenen begrenzt werden. Die Koordinaten eines Punktes $x \in \mathbb{R}^n$ einer Hyperebene $H \subset \mathbb{R}^n$ wird berechnet durch einen Stützvektor p , addiert mit linear unabhängigen Richtungsvektoren e_1, \dots, e_{n-1} mit den Vorfaktoren h_1, \dots, h_{n-1} .

$$x = p + h_1 e_1 + \dots + h_{n-1} e_{n-1} \quad (2.18)$$

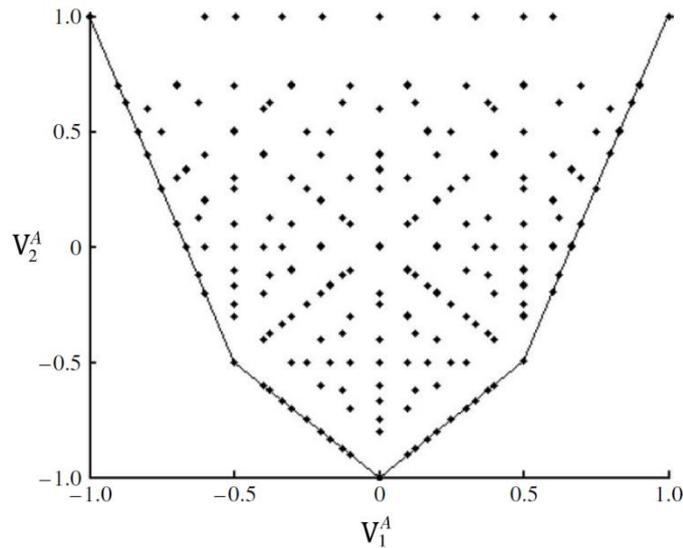


Abbildung 8: Lamination-Parameter-Gültigkeitsbereich

Es bietet sich hier an, als Richtungsvektoren die LP zu wählen. Für $n = 4$ gilt für einen Punkt auf der Grenze des zulässigen Bereichs, der auch gleichzeitig ein Punkt auf der Hyperebene beschreibt:

$$\bar{h}_1^j V_1^j + \bar{h}_2^j V_2^j + \bar{h}_3^j V_3^j + \bar{h}_4^j V_4^j + h_5^j = 0 \quad (2.12)$$

Für einen Punkt innerhalb des zulässigen Bereiches gilt:

$$h_1^j V_1^j + h_2^j V_2^j + h_3^j V_3^j + h_4^j V_4^j - 1 \leq 0 \quad (2.13)$$

wobei hier $h_i^j = -(\bar{h}_i^j / \bar{h}_5^j)$ ist. Diese Ungleichung muss bei der Optimierung eingehalten werden und stellt damit die Restriktion für die LP dar. Die Werte für h wird dem Optimierungsprozess als LP-Design-Raum zur Verfügung gestellt. Dieser Raum wird als Matrix dargestellt. Für jede Hyperebene besitzt die Matrix eine Zeile und stellt pro Zeile eine Restriktion dar. Die Anzahl der Spalten wird durch die Anzahl der LP vorgegeben, wobei jede Spalte einem LP zugeordnet wird. In Gleichung 2.19 ist zu erkennen, dass h_5 mit keinem LP multipliziert wird. Aus diesem Grund besitzt die Matrix noch eine zusätzliche Spalte. Um die Matrix mit den LP multiplizieren zu können, wird dem LP-Vektor noch eine konstante eins hinzugefügt [10].

$$\begin{bmatrix} h_1^A & h_2^A & h_3^A & h_4^A & h_1^B & h_2^B & h_3^B & \dots & h_5 \end{bmatrix} \begin{bmatrix} V_1^A \\ V_2^A \\ V_3^A \\ V_4^A \\ V_1^B \\ V_2^B \\ V_3^B \\ \vdots \\ 1 \end{bmatrix} \leq 0 \quad (2.14)$$

2.5.3 Versagensrestriktionen

Mithilfe der Versagensrestriktionen kann eine Aussage darüber getroffen werden, ob ein Bauteil unter bestimmten Lasten versagt oder ob es diesen Stand hält. Es wird dabei zwischen Beulstabilität des Paneels und Festigkeit des Materials unterschieden. Das Versagen kann durch Druck- oder Schublasten oder durch Kombination von beiden entstehen. Die Wahl des richtigen Kriteriums ist dabei essenziell für die wahrhafte Aussage über das Versagen des Bauteils. Je nach Bauteilgeometrie, auftretenden Lasten oder Materialart sollten verschiedene Versagenskriterien angewendet werden. Das Ergebnis des Versagenkriteriums beschreibt dabei die maximal zulässigen Spannungen σ_{max} , die auftreten dürfen, bevor das Bauteil versagt. Das Bauteil sollte also so ausgelegt werden, sodass σ_{max} bei realistischen Lastfällen nicht überschritten wird. Aus dieser Bedingung lassen sich nun Restriktionen herleiten, welche eingehalten werden müssen. Wird also das Verhältnis von tatsächlich auftretenden Spannungen σ zu maximal zulässigen Spannungen σ_{max} berechnet, so sollte das Ergebnis kleiner 1 sein, damit die Restriktion erfüllt ist [11]. Die Restriktion, um das Versagen eines Bauteils auszuschließen, lautet also:

$$g = \frac{n}{n_{max}} - 1 \leq 0 \quad (2.15)$$

Die tatsächlich auftretenden Spannungen in dem Bauteil werden in *lightworks* durch finite Elemente Methoden bestimmt. Dabei handelt es sich um einen numerischen Solver, der das Bauteil, von dem die Spannungen zu bestimmen sind, in finite Elemente einteilt. Diese Elemente sind durch Knoten miteinander verbunden. Die Knoten sind frei beweglich in alle drei Richtungen und können ebenfalls beliebig verdreht werden. Sie besitzen somit sechs Freiheitsgrade. Durch die auftretenden Lasten am Bauteil entstehen entsprechende Verzerrungen an den Knotenpunkten. Aus diesen Verzerrungen lassen sich die Spannungen in diesem Punkt herleiten. Gleichung 2.1 zeigt die Beziehungen zwischen Verzerrungen (ε, γ), den Kraftflüssen (\hat{n}, \hat{m}) und den Spannungen (σ, τ) auf.

2.5.4 Modellierung der Versagensrestriktionen

Der Verlauf des maximalen Dehnungs-Kriteriums erweist sich als unstetig. Unstetige Funktionen sind wie bekannt an der Unstetigkeitsstelle nicht differenzierbar. Um die Gradienten dieses Kriteriums zu bilden ist dies aber notwendig. Um dieses Problem zu lösen, wurde die Kreisselmeier-Steinhauser Funktion verwendet.

$$KS(g_j(x)) = \frac{1}{\rho} \ln \left[\sum_j^n e^{\rho g_j(x)} \right] \quad (2.16)$$

[12]

Mit dieser Funktion ist es möglich mehrere Restriktionen g zu einer stetigen Restriktion zu vereinen. Die entstandene Funktion schmiegt sich der maximalen Restriktion in jedem Punkt an. In dem maximalen Dehnungs-Kriterium wird für jede der drei Belastungsrichtungen eine Funktion berechnet und lediglich die maximal auftretende Dehnung wird zum Evaluieren des Versagens verwendet. Wird der Verlauf des Kriteriums über eine Belastungsrichtung aufgezeichnet, wird es einen Punkt geben, in dem die maximal auftretende Dehnung von einer Funktion auf die andere wechselt. In diesem Punkt entsteht dann die Unstetigkeitsstelle.

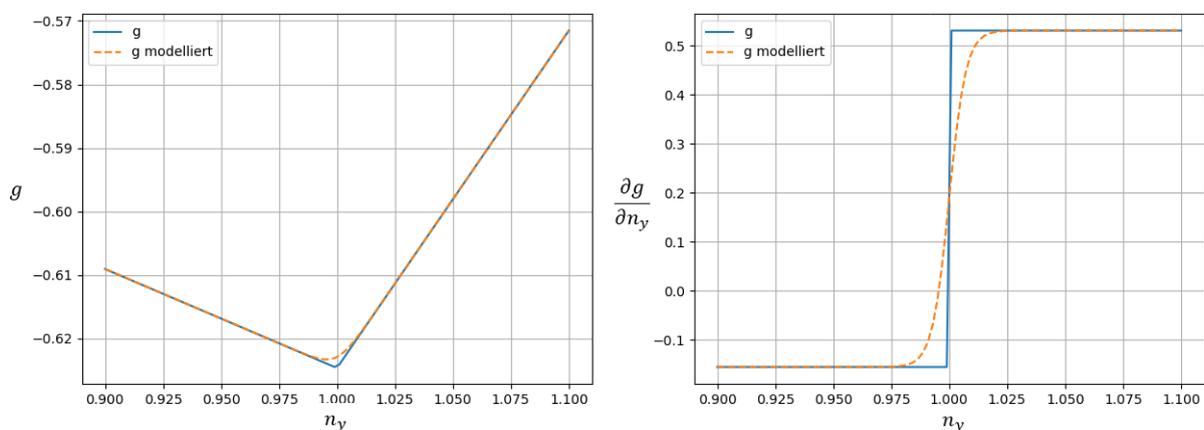


Abbildung 9: Verlauf der maximalen-Dehnungs-Restriktion unmodelliert und modelliert (links) und dessen Gradienten (rechts)

In Abbildung 9 links ist der Verlauf des maximalen-Dehnungs-Kriteriums über die Belastung in y -Richtung aufgezeichnet. Rechts ist der dazugehörige Verlauf des Gradienten abgebildet. Die blaue durchgezogene Linie zeigt den Verlauf der durch das Kriterium berechneten Restriktion. Zu erkennen ist der Punkt als Unstetigkeitsstelle, in dem die maximale Belastung die Richtung ändert. Charakteristisch für einen unstetigen Punkt ist der Sprung des Gradienten, der rechts zu erkennen ist. Die orange Linie zeigt den Verlauf der modellierten Restriktion mittels Kreisselmeier-Steinhauser-Funktion. Der Verlauf liegt exakt auf dem der nicht modellierten Funktion. Im Bereich der Unstetigkeit

jedoch hebt diese sich etwas von dem ursprünglichen Verlauf ab, wodurch die Unstetigkeit verschwindet. Im Verlauf des Gradienten ist demnach kein Sprung mehr zu verzeichnen.

Die Wahl der Größe des Parameters ρ hat Einfluss auf den Verlauf der Funktion. Je größer dieser gewählt wird, desto mehr schmiegt sich die modellierte Funktion der Ursprungsfunktion an. Für diesen Anwendungsfall ergaben sich die besten Ergebnisse, wenn ρ möglichst groß gewählt wurde. Der maximal mögliche Wert wird durch den maximal möglichen Wert einer 64-Bit Fließkommazahl und der Summe der drei Belastungsfunktionen festgelegt. Der Ansatz zur Bestimmung von ρ_{max} lautet:

$$e^{\sum \rho g_j(x)} \leq 10^{308} \quad (2.17)$$

Für ρ_{max} gilt dann:

$$\rho_{max} = \frac{\ln(10^{308})}{\sum_j g_j} \quad (2.18)$$

2.6 Gradienten

In diesem Kapitel werden die Verfahren vorgestellt, die für diese Arbeit verwendet werden, um Gradienten zu berechnen. Das Ziel ist es, die Gradienten der Restriktionsfunktionen zu berechnen. Dafür werden die Restriktionsfunktionen unterteilt partiell abgeleitet. Im Anschluss werden die Gradienten mithilfe der Kettenregel zusammengefügt, um die totalen Gradienten der Restriktionsfunktionen nach den globalen Parametern zu erhalten. Die Gradientenmethoden werden am Ende dieses Kapitels hinsichtlich Rechenaufwand miteinander verglichen.

2.6.1 Kettenregel

Mithilfe der Kettenregel können verkettete Funktionen getrennt und die partiellen Gradienten der Teilfunktionen einzeln bestimmt werden, um sie im Anschluss wieder zum totalen Gradienten zusammenzufügen. Einer Funktion $f(x(y), y)$, welche aus einer Funktion x und einem Parameter y besteht, abgeleitet nach Parameter y wird mithilfe der Kettenregel wie folgt berechnet:

$$\frac{\partial f}{\partial y} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial y} + \frac{\partial f}{\partial y} \quad (2.19)$$

2.6.2 Finite Differenzen

Die Gradienten mit finiten Differenzen zu berechnen stellt eine näherungsweise Berechnung der Gradienten dar. Hier wird zunächst der Funktionswert an der Stelle ausgewertet, an der die Gradienten der Funktion berechnet werden sollen. Ein zweiter Funktionswert wird im Abstand h zum ersten

gewählt und ebenfalls ausgewertet. Die Differenz der beiden Funktionswerte dividiert durch die Schrittweite h , stellt die Steigung bzw. den Gradienten der Funktion an der Stelle des ersten Funktionswertes dar. Die Vorwärtsdifferenz ist als

$$\frac{\partial f(\vec{x})}{\partial x_k} = \frac{f(\vec{x} + h_k \vec{e}_k) - f(\vec{x})}{h_k} + \mathcal{O}(\|h\|) \quad (2.20)$$

für ein mehrdimensionales Problem definiert. Hier ist h_k die Schrittweite für den Parameter x_k . Der Vektor e_k beschreibt einen Einheitsvektor, der in die Richtung des Parameters x_k weist. Der Term $\mathcal{O}(\|h\|)$ gibt die Größenordnung des Fehlers an, der bei der Berechnung mit finiten Differenzen gegenüber analytischen Berechnungen auftritt.

Die Berechnung mit finiten Differenzen kann auch als Annäherung einer Sekante, die durch zwei Punkte mit dem Abstand h der Funktion geht, angesehen werden. Wird der Betrag von h infinitesimal klein gewählt, dann wird aus der Sekante eine Tangente. Aus diesem Grund sollte die Schrittweite h intuitiv möglichst klein gewählt werden, damit die Sekante möglichst der Tangente entspricht. Wird die Schrittweite allerdings zu klein gewählt, überwiegen Rundungsfehler aufgrund der Maschinengenauigkeit und der Fehler wird wieder größer.

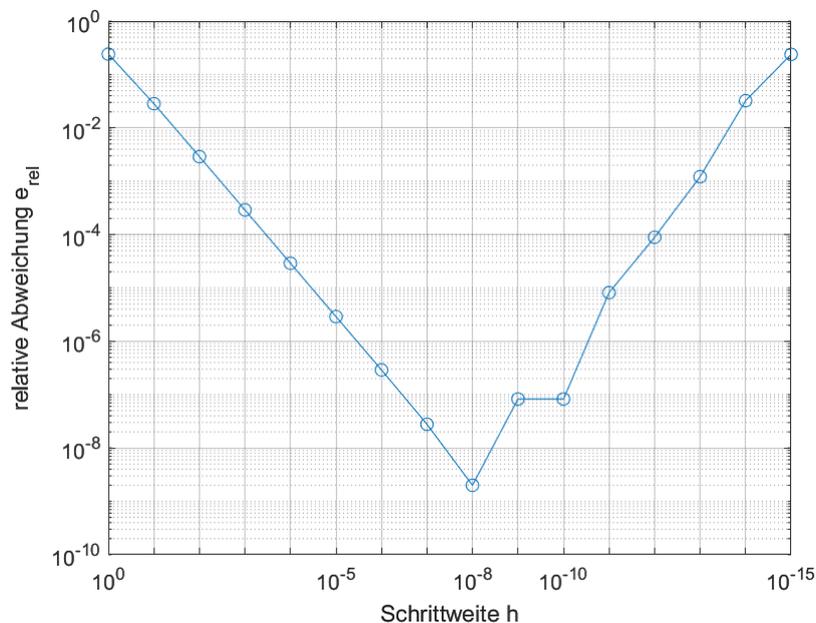


Abbildung 10: Relative Abweichung der finiten Differenzen über der Schrittweite

In Abbildung 10 ist aufgezeichnet welchen Einfluss die Schrittweite h auf die relative Abweichung e_{rel} und somit auf die Genauigkeit der Gradienten hat. Die Abweichung wurde mit

$$e_{rel} = \frac{|g_{exakt} - g_{approx}|}{|g_{approx}|} \quad (2.21)$$

berechnet, wobei g_{exakt} der tatsächliche Gradient der Funktion ist und g_{approx} der angenäherte Gradient durch finite Differenzen. Zu erkennen ist, dass die relative Abweichung zunächst stetig kleiner wird mit kleiner werdenden h . Bei einer Schrittweite von ungefähr 10^{-8} erreicht die relative Abweichung ihr Minimum, wonach sie aber mit kleiner werdenden h wieder ansteigt. Grund dafür ist die Gleitkommazahlgenauigkeit, welche den kleinsten darstellbaren Betrag von Rechnern darstellt. Daraus lässt sich die optimale Schrittweite berechnen, bei der der kleinste Fehler auftritt. Die optimale Schrittweite berechnet sich durch

$$h_{k,\text{opt}} = \sqrt{\varepsilon} * (1 + |x_k|) \quad (2.29)$$

[13]

Hierbei ist ε die Gleitkommazahlgenauigkeit. Bei für Rechner üblichen ε in der Größenordnung von 10^{-16} ergeben sich demnach optimale Schrittweiten im Bereich von 10^{-8} , was das Minimum im Diagramm aus Abbildung 10 erklärt.

2.6.3 Automatische Differenzierung

Es handelt sich hierbei um die Differenzierung von Programmcode durch Zerlegung einer Funktion in elementare Unterfunktionen. Die Funktion wird so oft zerlegt, sodass die „unterste“ Ebene nur noch von den Variablen abhängig ist, nach denen differenziert werden soll. Es kann dabei zwischen zwei Richtungen unterschieden werden.

2.6.3.1 Vorwärts-Methode

Bei dieser Methode wird zunächst die Funktion $f(x)$ in mehrere Unterfunktionen x_i unterteilt. Beim Teilen der Funktion wird die Kettenregel verwendet. Bei einer Funktion $f(x_1(x_2))$ wird bei der Vorwärtsmethode zunächst die unterste Ebene abgeleitet. Die darüber liegenden Ebenen werden dann Schritt für Schritt ausgewertet bis zur obersten.

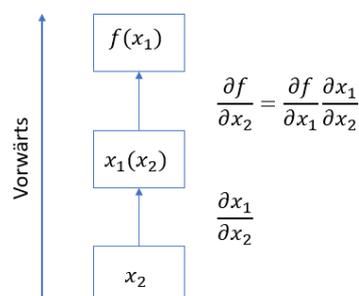


Abbildung 11: Grafische Darstellung der Vorwärtsmethode

Anders als bei den finiten Differenzen geben die automatischen Differenzen den exakten Wert wieder und ist keine Approximation. Der Nachteil bei der Vorwärtsmethode ist, dass für die Berechnung der Gradienten nach jedem abhängigen Parameter ein Funktionsdurchlauf pro Parameter benötigt wird. Ähnlich wie bei den finiten Differenzen sind hier n -Funktionsaufrufe nötig, wobei n die Anzahl der abhängigen Parameter ist.

2.6.3.2 Rückwärts-Methode

Anders als bei der Vorwärtsmethode wird bei der Rückwärtsmethode zunächst die oberste Ebene berechnet und im Anschluss die darunter liegenden. Bei dieser Methode ist daher das Speichern der zuvor berechneten Gradienten erforderlich.

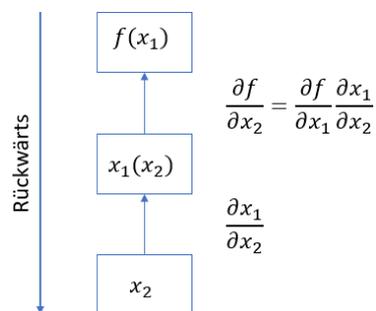


Abbildung 12: Grafische Darstellung der Rückwärtsmethode

Der Vorteil bei dieser Methode gegenüber der Vorwärts-Methode ist, dass hier nur ein Durchlauf erforderlich ist, um einen Gradienten nach allen abhängigen Parametern zu berechnen. Dieses Verfahren kann also bei großer Anzahl an Parametern Rechenaufwand einsparen. Nachteilig ist aber, dass hier die Zwischenergebnisse der Ableitungen gespeichert werden müssen. Bei komplexen oder vielen Funktionen kann diese Methode also einen hohen Speicherbedarf erfordern.

Die Wahl der beiden Verfahren ist also von der Anzahl der Parameter n , sowie Anzahl der Funktionen m abhängig. Ist $n \ll m$, sollte auf die Vorwärts-Methode zurückgegriffen werden, um einen möglichen hohen Speicherbedarf durch die Rückwärts-Methode zu verhindern. Ist $n \gg m$, sollte dann die Rückwärts-Methode angewendet werden, da so viel Rechenzeit eingespart werden kann [14].

2.6.4 Autograd

Für die automatische Differenzierung wurde die Python-Bibliothek *autograd* verwendet, da *lightworks* ebenfalls ausschließlich in Python geschrieben ist. *Autograd* wurde speziell für eine gradientenbasierte Optimierung entwickelt und kann Gradienten mit der Vorwärts- sowie Rückwärtsmethode ermitteln.

Für die Anwendung müssen Funktionen definiert werden, die genau ein Argument als Input besitzen und einen als Output. Der Eingabeparameter kann ein skalarer Wert sein oder auch ein Vektor aus mehreren Werten sein. Die Funktion wird dann nach diesen Parametern abgeleitet. Die Ausgabe kann ebenfalls ein skalarer Wert oder Vektor sein. Dieser stellt dann das Ergebnis der Funktion dar. Aus einer Funktion mit n Eingabeparametern und m Ausgabeparametern lässt sich eine $m \times n$ -Jacobi-Matrix bilden. In einer Jacobi-Matrix sind die Ableitungen der Funktionen g_0, \dots, g_m nach den Parametern x_0, \dots, x_n aufgelistet.

$$J = \begin{pmatrix} \frac{\partial g_0}{\partial x_0} & \dots & \frac{\partial g_0}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_m}{\partial x_0} & \dots & \frac{\partial g_m}{\partial x_n} \end{pmatrix} \quad (2.22)$$

Zum Berechnen dieser Jacobi-Matrix besitzt *autograd* eine Methode, die das bewerkstelligt. Zunächst muss dieser Methode eine Funktion übergeben werden, welche abgeleitet werden soll. Aus dieser Funktion erstellt die Jacobi-Methode eine Ableitungsfunktion, welcher dann die Parameter übergeben werden kann, nach welchen die ursprüngliche Funktion abgeleitet werden soll. Die Ableitungsfunktion verfolgt jede Operation, die mit eingegebenen Parametern durchgeführt werden, um auf das Endergebnis zu kommen. Aus den durchgeführten Operationen kann ein Graph erstellt werden, ähnlich wie in Abbildung 11 und Abbildung 12. Zum Schluss werden die bekannten Ableitungsregeln auf jede durchgeführte Operation angewendet. Das Resultat daraus ist die Jacobi-Matrix. In dieser Arbeit wird diese Methode als AD-Jacobi bezeichnet. Diese Methode funktioniert nach dem Prinzip der Rückwärtsmethode [15].

Autograd besitzt zwei weitere Methoden zur Gradientenberechnung, welche allerdings nicht explizit in der *autograd*-Dokumentation erwähnt werden. Eventuell kann durch einer dieser beiden Funktionen die Effizienz einiger Gradienten noch gesteigert werden. Bei den beiden Methoden handelt es sich um eine Rückwärts- und eine Vorwärtsmethode. Vom Prinzip funktionieren diese Methoden wie die oben beschriebene. Anders als oben ergibt sich nach der Berechnung der Funktion keine Jacobi-Matrix, sondern eine weitere Funktion, aus der die Jacobi-Matrix selbst erzeugt werden muss. In diese erzeugte Funktion muss ein Einheitsvektor eingefügt werden, welche dieselbe Länge wie die der abzuleitenden Parameteranzahl hat. Ergebnis ist die Spalte des Parameters in die Richtung, in die der Einheitsvektor gerichtet ist.

2.6.5 Vergleich der Methoden

Um den Unterschied der Methoden aufzuzeigen, soll eine einfache Beispielfunktion abgeleitet werden. In diesem Fall wird die Sinusfunktion abgeleitet. Da sich das Verhalten der Gradientenmethoden über

die Anzahl der Parameter verändern kann, wird bei dem Test die Anzahl der Parameter erhöht. Bei der abzuleitenden Funktion wird für jeden übergebenen Parameter die Sinusfunktion an dieser Stelle ausgewertet. Die Ausgabe der Funktion soll sich aber nicht ändern, um nur den Einfluss der Parameteranzahl zu überprüfen. Deshalb wird lediglich der Mittelwert aller berechneten Sinusfunktionen zurückgegeben.

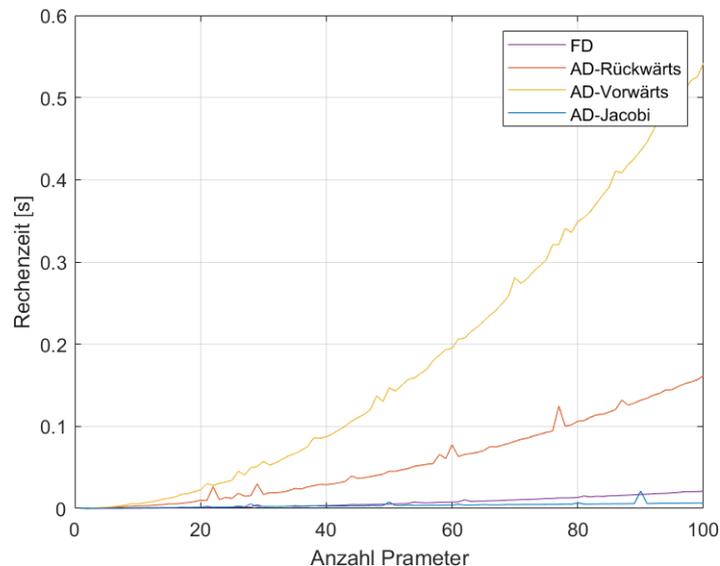


Abbildung 13: Vergleich der Rechenzeiten der Gradientenmethoden bei der Ableitung einer Sinus-Funktion

Wie zu erwarten steigt der Aufwand der Vorwärtsmethode mit der Parameteranzahl an. Dieser steigt aber deutlich größer an als bei den finiten Differenzen, welcher ebenfalls mit Erhöhung der Parameteranzahl ansteigt. Bei der AD-Rückwärtsmethode steigt der Aufwand ebenfalls weiter an, wenn die Anzahl der abhängigen Parameter erhöht wird. Die Rechenzeit sollte hier nach der Theorie kaum von der Parameteranzahl abhängig sein. Dennoch ist dieser Effekt hier zu erkennen. Die AD-Jacobi-Methode scheint kaum von der Parameteranzahl abhängig zu sein. Diese Methode funktioniert ebenfalls gemäß der Rückwärtsmethode, ist aber deutlich effizienter als die AD-Rückwärtsmethode.

Um Gradienten zu berechnen ist also laut diesem Beispiel die empfohlene Methode die AD-Jacobi-Methode. Die anderen Methoden werden aber für die späteren Tests noch nicht verworfen, da sich die Effizienz der Methoden für andere Gradienten noch ändern kann.

3 Implementierung

3.1 Aufstellung der Kettenregel

In diesem Kapitel werden die Kettenregeln für das Optimierungsproblem hergeleitet, um die Gradienten aller Restriktionen nach allen globalen Parametern zu bilden. Dazu werden die Restriktionsfunktionen zerlegt und deren Gradienten ermittelt. Mit der Kettenregel aus Kapitel 3.1 können die Gradienten der partiellen Funktionen zusammengefügt werden. So ergeben sich die totalen Gradienten der Restriktionsfunktionen. Dazu ist zu klären, wie sich die Restriktionsfunktionen zusammensetzen. Die hier behandelten Restriktionsfunktionen wurden in Kapitel 2.5 vorgestellt. Es wird unterschieden zwischen:

- Rampenwinkel-Restriktion (g_{ts})
- Lamination-Parameter-Gültigkeitsbereich (g_{lp})
- Versagensrestriktion (g_{pf})

Die globalen Parameter nach denen abgeleitet werden kann, wurden in Kapitel 2.4 behandelt. Diese sind:

- Hautdicke der Flügelhaut (t_{sk})
- Hautdicke der Stringerelemente (t_e)
- Schichtdicken (t_i)
- Schichtwinkel (θ_i)
- Lamination-Parameter ($V_{[1,2,3,4]}^{A,B,D}$)
- Stringerwinkel (θ_{str})
- Stringerhöhe (h_{str})
- Stringerabstand (d_{str})
- Fußbreite (w_f)
- Flanschbreite (w_{fl})

Im Folgenden werden die Restriktionsfunktionen Schritt für Schritt behandelt und geklärt, inwieweit diese von den oben genannten globalen Parametern abhängig sind. Daraus werden dann die partiellen Ableitungen der Restriktionsfunktionen nach den globalen Parametern gebildet.

Die Rampenwinkel-Restriktion beschreibt die maximale Dickenänderung zwischen zwei Paneelen. Aus Gleichung 2.16 ist zu erkennen, dass diese Restriktion direkt und allein von der Hautdicke des Flügels abhängig ist. Demnach ist hier keine Aufteilung notwendig. Die Restriktion kann direkt nach der Hautdicke abgeleitet werden. Ableitungen nach anderen Parametern sind dementsprechend null.

Der LP-Gültigkeitsbereich gibt die möglichen LP Kombinationen an. Laut Gleichung 2.21 ist diese Restriktion direkt von den LP abhängig. Ähnlich wie bei der Rampenwinkel-Restriktion ist hier ebenfalls keine Aufteilung nötig. Die Restriktion wird direkt nach den LP abgeleitet und alle Ableitungen nach anderen Parametern sind ebenfalls null.

Wird die Gleichung 2.22 der Versagensrestriktion betrachtet, fällt auf, dass diese nun nicht mehr direkt von den oben genannten Parametern abhängig ist. Diese Restriktion aufzuteilen ist hier also erforderlich. Dabei ist zu klären, wie die Parameter und die Restriktion zusammenhängen.

Alle Parameter, nach denen abgeleitet werden kann, sind im Strukturmodell enthalten. Mit diesen Parametern sind im Strukturmodell die Paneele definiert. Hier werden ebenfalls die ABD-Matrizen der Paneele berechnet, welche für den weiteren Verlauf benötigt werden. Zusammen mit Lasten werden die Paneele in einen FE-Solver geladen, wo Deformationen und Spannungen der Paneele ausgegeben werden. Zusammen mit den Paneel-ABD-Matrizen werden diese dann in den Restriktionsprozessor gegeben. Dort werden dann die Restriktionen aufgrund der eingespeisten Informationen berechnet. In Abbildung 14 ist der Verlauf schematisch dargestellt.

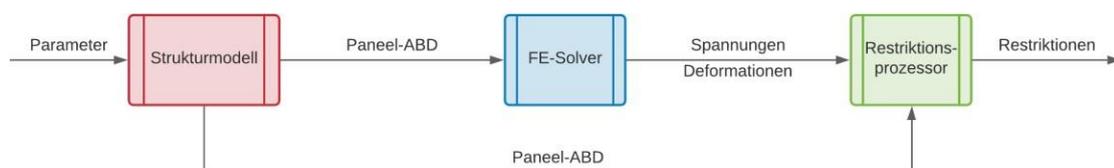


Abbildung 14: Modulare Aufteilung der Optimierungsprozesses

Bei der Aufteilung der Gradienten wird so vorgegangen, dass die einzelnen Module, welche aus Abbildung 14 erkenntlich sind, separat betrachtet werden. Für jedes einzelne Modul werden die nötigen partiellen Gradienten berechnet. Was aus dem Modul als Ergebnis berechnet wird, kann dabei nach dem, was in das Modul eingeführt wurde, abgeleitet werden. So kann im Strukturmodell beispielsweise die Paneel-ABD-Matrix nach den globalen Parametern abgeleitet werden und im FE-Solver die Spannungen und Deformationen wiederum nach der Paneel-ABD-Matrix.

3.1.1 Strukturmodell

Wie schon oben beschrieben wird hier die Paneel-ABD-Matrix nach den globalen Parametern x abgeleitet. Im Strukturmodell muss bei der Berechnung der ABD-Matrix zwischen versteiften und nicht versteiften Paneelen differenziert werden. Bei nicht versteiften Paneelen gilt, dass die Paneel-ABD-Matrix mit der ABD-Matrix der Haut gleichzusetzen ist. Es gilt also:

$$ABD_p = ABD_{sk} \quad (3.1)$$

Demnach berechnet sich der Gradient der ABD-Matrix wie folgt:

$$\frac{dABD_p}{dx} = \frac{dABD_{sk}}{dx} \quad (3.2)$$

Bei der Ableitung der ABD-Matrix handelt es sich um eine elementweise Ableitung aller Einträge der Matrix.

$$\frac{\partial ABD_p}{\partial x} = \begin{pmatrix} \frac{\partial A_{11}}{\partial x} & \dots & \frac{\partial B_{16}}{\partial x} \\ \vdots & \ddots & \vdots \\ \frac{\partial B_{61}}{\partial x} & \dots & \frac{\partial D_{66}}{\partial x} \end{pmatrix} \quad (3.3)$$

Bei versteiften Paneelen muss die Stringer-ABD-Matrix mit in die Berechnung einbezogen werden. Hier gilt:

$$ABD_p = ABD_{sk} + ABD_{str} \quad (3.4)$$

[4]

Um den Gradienten der Paneel-ABD-Matrix nach den ableitbaren Parametern zu erhalten, muss die Kettenregel aus Kapitel 2.6.1 angewendet werden.

$$\frac{dABD_p}{dx} = \frac{\partial ABD_p}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} + \frac{\partial ABD_p}{\partial ABD_{str}} \frac{\partial ABD_{str}}{\partial x} \quad (3.5)$$

Diese Berechnung gilt für Paneele mit Stringer, die keinen Fuß haben. Besitzt der Stringer jedoch einen Fuß, wie es z. B. bei einem T- oder I-Stringer der Fall ist, fließt die ABD-Matrix des Fußes direkt in die Berechnung des a_{33} -Eintrags der Paneel-ABD-Matrix. Für den Eintrag gilt:

$$a_{33} = \frac{d_{str}}{\frac{w_f}{(a_{33,f} + a_{33,sk})} + \frac{d_{str}}{a_{33,sk}}} \quad (3.6)$$

[16]

Erkennbar ist, dass Gleichung 3.6 auch direkt vom Stringerabstand und der Fußbreite abhängig ist, welche auch als globale Parameter gewählt werden können. Die Kettenregel der Gleichung 3.5 für ein versteiftes Paneel mit einem Stringer mit Fußelement erweitert sich demnach um der Ableitung der Paneel-ABD-Matrix nach der ABD-Matrix des Fußes.

$$\frac{dABD_p}{dx} = \frac{\partial ABD_p}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} + \frac{\partial ABD_p}{\partial ABD_{str}} \frac{\partial ABD_{str}}{\partial x} + \frac{\partial ABD_p}{\partial ABD_f} \frac{\partial ABD_f}{\partial x} \quad (3.7)$$

Speziell für den Fall, dass nach Stringerabstand oder Fußbreite abgeleitet werden soll, wird folgender Term zu der Kettenregel hinzugefügt.

$$\begin{aligned} \frac{dABD_p}{d[d_{str}, w_f]} &= \frac{\partial ABD_p}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial [d_{str}, w_f]} + \frac{\partial ABD_p}{\partial ABD_{str}} \frac{\partial ABD_{str}}{\partial [d_{str}, w_f]} + \frac{\partial ABD_p}{\partial ABD_f} \frac{\partial ABD_f}{\partial [d_{str}, w_f]} \\ &+ \frac{\partial ABD_p}{\partial [d_{str}, w_f]} \end{aligned} \quad (3.8)$$

3.1.2 Finite-Elementen-Solver (FE-Solver)

Als Nächstes wird der FE-Solver behandelt. Hier ergeben sich die Materialspannung und Deformationen eines Panels unter bestimmten Lastfällen. Aus Gleichung 2.1 ist zu erkennen, dass die Spannungen von der Panel-ABD-Matrix abhängig sind. Dementsprechend sollen hier die Spannungen und Deformationen nach diesen ABD-Matrizen abgeleitet werden. Mit der Kettenregel können dann auch hier die Gradienten in Abhängigkeit der globalen Parameter bestimmt werden. Die Gradienten des Spannungsvektors berechnen sich also wie folgt.

$$\begin{pmatrix} \frac{\partial \hat{n}_x}{\partial x} \\ \frac{\partial \hat{n}_y}{\partial x} \\ \frac{\partial \hat{n}_{xy}}{\partial x} \\ \frac{\partial \hat{m}_x}{\partial x} \\ \frac{\partial \hat{m}_y}{\partial x} \\ \frac{\partial \hat{m}_{xy}}{\partial x} \end{pmatrix} = \begin{pmatrix} \frac{\partial \hat{n}_x}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} \\ \frac{\partial \hat{n}_y}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} \\ \frac{\partial \hat{n}_{xy}}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} \\ \frac{\partial \hat{m}_x}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} \\ \frac{\partial \hat{m}_y}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} \\ \frac{\partial \hat{m}_{xy}}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} \end{pmatrix} = \frac{\partial \vec{n}}{\partial x} \quad (3.9)$$

3.1.3 Restriktionsprozessor

Im Restriktionsprozessor werden nun die Stabilitäts- und Festigkeitsrestriktionen berechnet. Diese sind nach Gleichung 2.22 direkt von den auftretenden Materialspannungen und Deformationen im Panel abhängig. Bei der Ableitung der Restriktionen ist nun wieder abzuwägen, welche Art Panel hier abgeleitet wird. Ebenfalls ist zu unterscheiden, ob die Restriktion über Festigkeits- oder Stabilitätskriterien berechnet wurde. Die Festigkeitskriterien, welche eine Aussage über das Versagen eines Materials treffen, beziehen sich auf die Haut eines Panels. Deshalb muss hier die ABD-Matrix der Haut berücksichtigt werden. Die Stabilitätskriterien gehen auf das Beulen oder die Verformungen eines Panels ein. Deshalb hat die ABD-Matrix des Panels hier einen Einfluss auf das Kriterium.

3.1.3.1 Festigkeitskriterien

Es werden zunächst die Festigkeitskriterien betrachtet. Die Festigkeitskriterien werden lagenweise ausgewertet, solange die Haut als Laminat vorliegt. Aus Kapitel 2.2 ist bekannt, dass ein Laminat auch über LP beschrieben werden kann. Hierbei gehen die Informationen über den Lagenaufbau verloren. Eine lagenweise Auswertung ist also nicht mehr möglich und so muss das Festigkeitskriterium auf das komplette Laminat angewendet werden. Auch hier ist zu unterscheiden zwischen versteiften und nicht versteiften Paneelen. Für eine bessere Übersicht, aus welchen Modulen die partiellen Gradienten entstehen, werden die nachfolgenden Kettenregeln farblich gestaltet. Die Farben entsprechen denen aus Abbildung 14.

Es gibt Festigkeitskriterien, die werden nach der vorliegenden Dehnung ε ausgewertet und es gibt andere, die nach der vorliegenden Materialspannung σ ausgewertet werden. Je nachdem welches der beiden Fälle eintritt, ist entweder der linke oder der rechte Teil der untenstehenden Gleichung gleich null und kann somit ignoriert werden. Wenn das Festigkeitskriterium beispielsweise nach den Spannungen ausgewertet wird, ist dieses gleich null, wenn es nach den Dehnungen abgeleitet wird. Somit kann dann der linke Teil ignoriert werden.

$$\frac{dg_{pf}}{dx} = \frac{\partial g_{pf}}{\partial \varepsilon_g} \left(\frac{\partial \varepsilon_g}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} + \frac{\partial \varepsilon_g}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} \right) + \frac{\partial g_{pf}}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} \quad (3.10)$$

Wird das Festigkeitskriterium jedoch lagenweise ausgewertet, so wird die Belastung auf die Lagen umgerechnet. Die Lagenbelastung σ_l wird durch die globale Spannung σ_g und einer Transformationsmatrix, die mithilfe des Lagenwinkels bestimmt wird, berechnet. Somit ändert sich die rechte Seite der Gleichung 3.10.

$$\frac{dg_{pf}}{dx} = \frac{\partial g_{pf}}{\partial \varepsilon_g} \frac{\partial \varepsilon_g}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial x} + \frac{\partial g_{pf}}{\partial \sigma_l} \left(\frac{\partial \sigma_l}{\partial \varepsilon_g} \frac{\partial \varepsilon_g}{\partial x} + \frac{\partial \sigma_l}{\partial \theta} \right) \quad (3.11)$$

mit

$$\frac{d\varepsilon_g}{dx} = \left(\frac{\partial \varepsilon_g}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} + \frac{\partial \varepsilon_g}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} \right) \quad (3.12)$$

Die Ableitung der Lagenbelastung nach den Lagenwinkel fließt nur in die Berechnung mit ein, wenn die Gradienten des Festigkeitskriteriums nach den Lagenwinkel bestimmt werden.

Bei versteiften Paneelen wird das Festigkeitskriterium für die Haut des Paneels aber auch für die einzelnen Stringerelemente angewendet. Die Gleichung 3.10 gilt hier auch mit dem Unterschied, dass statt den Gradienten der Haut-ABD-Matrix die Gradienten der ABD-Matrix des jeweiligen Stringerelements für die Kettenregel verwendet werden. Die Last, welches auf das Paneel wirkt, muss auf die Stringerelemente verteilt werden. Hierfür wird die globale Belastung des Paneels σ_g , mit der ABD-Matrix des jeweiligen Stringerelements multipliziert. Daraus ergibt sich die verteilte Belastung σ_r .

Ebenfalls wird die globale Dehnung ε_g , welche sich aus globaler Belastung σ_g und inverser ABD-Matrix berechnet, auf die Stringerelemente verteilt. Das Resultat ist die verteilte Dehnung ε_g . Demnach hat die Paneel-ABD-Matrix und die ABD-Matrizen der Stringerelemente einen Einfluss auf die kritische Spannung des Kriteriums.

$$\frac{dg_{pf}}{dx} = \frac{\partial g_{pf}}{\partial \varepsilon_r} \frac{\partial \varepsilon_r}{\partial \varepsilon_g} \frac{\partial \varepsilon_g}{\partial x} + \frac{\partial g_{pf}}{\partial \sigma_r} \left(\frac{\partial \sigma_r}{\partial n_g} \frac{\partial \sigma_g}{\partial x} + \frac{\partial \sigma_r}{\partial ABD_e} \frac{\partial ABD_e}{\partial x} \right) \quad (3.13)$$

mit

$$\frac{d\varepsilon_g}{dx} = \left(\frac{\partial \varepsilon_g}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} + \frac{\partial \varepsilon_g}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} \right) \quad (3.14)$$

Mit dem Element Index e wird auf das jeweilige Element verweist, von welchem das Festigkeitskriterium berechnet wird. Wird z. B. die Festigkeit des Fußes überprüft, muss in der obigen Gleichung die ABD-Matrix des Fußes eingesetzt werden.

Auch hier muss bei lagenweiser Betrachtung die Belastung auf die jeweilige Lage umgerechnet werden. Gleichung 3.13 ändert sich in dem Fall zu:

$$\frac{dg_{pf}}{dx} = \frac{\partial g_{pf}}{\partial \varepsilon_r} \frac{\partial \varepsilon_r}{\partial \varepsilon_g} \frac{\partial \varepsilon_g}{\partial x} + \frac{\partial g_{pf}}{\partial \sigma_l} \left(\frac{\partial \sigma_l}{\partial \sigma_r} \frac{\partial \sigma_r}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial x} + \frac{\partial \sigma_l}{\partial \theta} \right) \quad (3.15)$$

3.1.3.2 Stabilitätskriterien

Auch bei den Stabilitätskriterien muss unterschieden werden, ob das Paneel Stringer besitzt oder nicht. Es wird zunächst der Fall ohne Stringer betrachtet. Die Stabilitätskriterien, die das Beulen eines Paneels beschreiben, gelten nur für symmetrische Lamine. Symmetrische Lamine zeichnen sich unter anderem dadurch aus, dass die B-Matrix der ABD-Matrix einer Nullmatrix gleicht. Ist das nicht der Fall, muss die D-Matrix mit folgender Formel angenähert werden.

$$\bar{D} = D - B^T A^{-1} B \quad (3.16)$$

[17]

Die Stabilitätskriterien eines Paneels sind für den Fall, dass das Paneel nicht versteift ist, direkt von der angenäherten D-Matrix des Paneels abhängig. Somit gilt hier für den Gradienten:

$$\frac{dg_{pf}}{dx} = \frac{\partial g_{pf}}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} + \frac{\partial g_{pf}}{\partial \bar{D}_{Sk}} \frac{\partial \bar{D}_{Sk}}{\partial ABD_{Sk}} \frac{\partial ABD_{Sk}}{\partial x} \quad (3.17)$$

Bei Paneelen mit Stringer werden verschiedene Fälle betrachtet. Zunächst wird das komplette Paneel mitsamt dem Stringer auf Beulen untersucht. Das wird auch als globales Beulen bezeichnet. In diesem Fall ist nur die D-Matrix des Paneels auf die Restriktion ausschlaggebend. Somit bedeutet das für den Gradienten:

$$\frac{dg_{pf}}{dx} = \frac{\partial g_{pf}}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial ABD_p} \frac{\partial ABD_p}{\partial x} + \frac{\partial g_{pf}}{\partial D_p} \frac{\partial D_p}{\partial x} \quad (3.18)$$

Weiterhin wird auch das lokale Beulen evaluiert. Hier muss zwischen dem offenen Bereich und dem Stringerabstands-Bereich unterschieden werden. Diese beschreiben den Bereich eines Paneels zwischen zwei Stringer. Abbildung 15 zeigt ein Paneel mit drei T-Stringer. Der gelb markierte Bereich ist der offene Bereich, bei der nur die Haut des Paneels betrachtet wird. Beim orangen markierten Stringerabstands-Bereich ist zu erkennen, dass der Stringerfuß ebenfalls einen Einfluss hat.

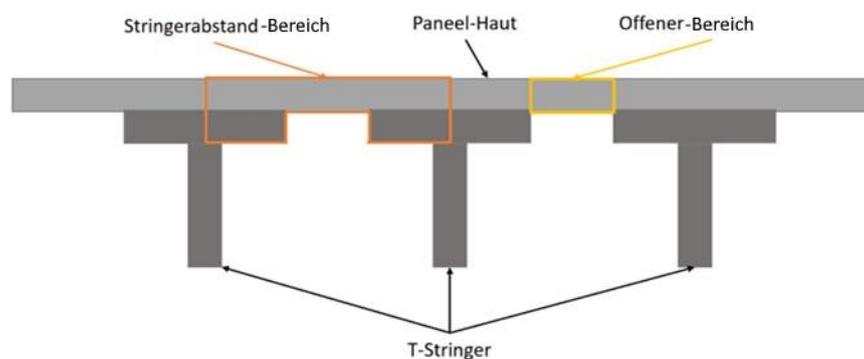


Abbildung 15: Darstellung des Stringerabstand- und offenen Bereiches in einer Paneelhaut

Im offenen Bereich wird nur die Haut des Paneels ohne den Einfluss des Stringers untersucht. Für die Auswertungen der Stabilitätskriterien werden die Materialbelastungen herangezogen. Ähnlich wie bei den Festigkeitskriterien muss die globale Belastung des Paneels zunächst in die lokale Belastung der Haut verteilt werden.

$$\frac{dg_{pf}}{dx} = \frac{\partial g_{pf}}{\partial \sigma_r} \left(\frac{\partial \sigma_r}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial x} + \frac{\partial \sigma_r}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} \right) + \frac{\partial g_{pf}}{\partial \bar{D}_{sk}} \frac{\partial \bar{D}_{sk}}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} \quad (3.19)$$

Die Geometrie des Paneels ist ebenfalls beim Beulen des jeweiligen Bereichs zu berücksichtigen. Aus Abbildung 15 ist zu erkennen, dass der offene Bereich sich aus dem Stringerabstand und der halben Fußbreite zusammensetzt. Wird das Kriterium nach diesen beiden Parametern abgeleitet, ist also noch ein zusätzlicher Term zur Gleichung 3.19 erforderlich.

$$\frac{dg_{pf}}{d[d_{str}, w_f]} = \frac{\partial g_{pf}}{\partial \sigma_r} \left(\frac{\partial \sigma_r}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial x} + \frac{\partial \sigma_r}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} \right) + \frac{\partial g_{pf}}{\partial \bar{D}_{sk}} \frac{\partial \bar{D}_{sk}}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} + \frac{\partial g_{pf}}{\partial [d_{str}, w_f]} \quad (3.20)$$

Der Stringerabstands-Bereich ist der zwischen zwei Stringern mit der Hinzunahme des Stringerfußes. Da der Fuß nicht beulen kann, hat dieser einen Einfluss auf die Beuleigenschaften des Paneels in diesem Bereich. Es können die gleichen Stabilitätskriterien wie im offenen Bereich verwendet werden, jedoch muss die ABD-Matrix des Paneels in diesem Bereich durch eine spezielle ABD-Matrix ABD_{ss} angenähert werden. Die globale Belastung muss für diesen Fall auf den Stringerabstand-Bereich verteilt werden. Die verteilte Belastung ist abhängig von der globalen Dehnung und Stringerabstand- ABD-Matrix.

$$\frac{dg_{pf}}{dx} = \frac{\partial g_{pf}}{\partial \sigma_r} \left(\frac{\partial \sigma_r}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial x} + \frac{\partial \sigma_r}{\partial ABD_{ss}} \frac{\partial ABD_{ss}}{\partial x} \right) + \frac{\partial g_{pf}}{\partial ABD_{ss}} \frac{\partial ABD_{ss}}{\partial x} \quad (3.21)$$

Hier ist aus Abbildung 15 ersichtlich, dass die Geometrie anders als im offenen Bereich nur vom Stringerabstand abhängig ist. Wird nach diesem abgeleitet, erweitert sich Gleichung 3.21 zu:

$$\frac{dg_{pf}}{dd_{str}} = \frac{\partial g_{pf}}{\partial \sigma_r} \left(\frac{\partial \sigma_r}{\partial \sigma_g} \frac{\partial \sigma_g}{\partial x} + \frac{\partial \sigma_r}{\partial ABD_{ss}} \frac{\partial ABD_{ss}}{\partial x} \right) + \frac{\partial g_{pf}}{\partial ABD_{ss}} \frac{\partial ABD_{ss}}{\partial x} + \frac{\partial g_{pf}}{\partial d_{str}} \quad (3.22)$$

Die Stringerabstand-ABD-Matrix setzt sich zusammen aus der ABD-Matrix der Haut und ABD-Matrix des Fußes. Außerdem haben deren Dicken, Fußbreite und Stringerabstand einen Einfluss auf die Stringerabstand-ABD-Matrix. Somit berechnet sich der Gradient der Stringerabstand-ABD-Matrix nach den oben genannten Parametern wie folgt:

$$\frac{dABD_{ss}}{d[t_{sk}, t_f, d_{str}, w_f]} = \frac{\partial ABD_{ss}}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial [t_{sk}, t_f, d_{str}, w_f]} + \frac{\partial ABD_{ss}}{\partial ABD_f} \frac{\partial ABD_f}{\partial [t_{sk}, t_f, d_{str}, w_f]} + \frac{\partial ABD_{ss}}{\partial [t_{sk}, t_f, d_{str}, w_f]} \quad (3.23)$$

Für alle anderen Parameter entfällt der letzte Term.

$$\frac{dABD_{ss}}{dx} = \frac{\partial ABD_{ss}}{\partial ABD_{sk}} \frac{\partial ABD_{sk}}{\partial x} + \frac{\partial ABD_{ss}}{\partial ABD_f} \frac{\partial ABD_f}{\partial x} \quad (3.24)$$

3.2 Gradientenprozessor

Zum jetzigen Zeitpunkt der Arbeit werden die Gradienten der Restriktionen direkt nach den globalen Parametern über finite Differenzen berechnet. An dieser Stelle soll jetzt ein Gradientenprozessor den bisherigen Prozess der Gradientenberechnung ersetzen. Die Aufgabe des Gradientenprozessors ist es, die für die aktuelle Optimierungsiteration benötigten partiellen Gradienten zu sammeln. Die partiellen Gradienten werden dann dort mithilfe der Kettenregel zusammengesetzt, um die gleichen totalen Gradienten der Restriktionen nach den globalen Parametern zu erhalten. Der Vorteil dadurch ist, dass einige einfache lineare partielle Gradienten analytisch berechnet werden können. Die analytische Berechnung der Gradienten bringt einen zeitlichen Vorteil gegenüber finiten Differenzen oder automatische Differenzierung mit sich. Generell kann dann für jeden partiellen Gradienten die optimale Methode der Gradientenberechnung gewählt werden. Das kann abhängig von der Komplexität, Anzahl der Parameter, nach denen abgeleitet wird oder Anzahl der Funktionen, die abgeleitet werden, sein. Insgesamt soll sich durch den Gradientenprozessor eine Reduzierung der Rechenzeit gegenüber der bisherigen Methode ergeben.

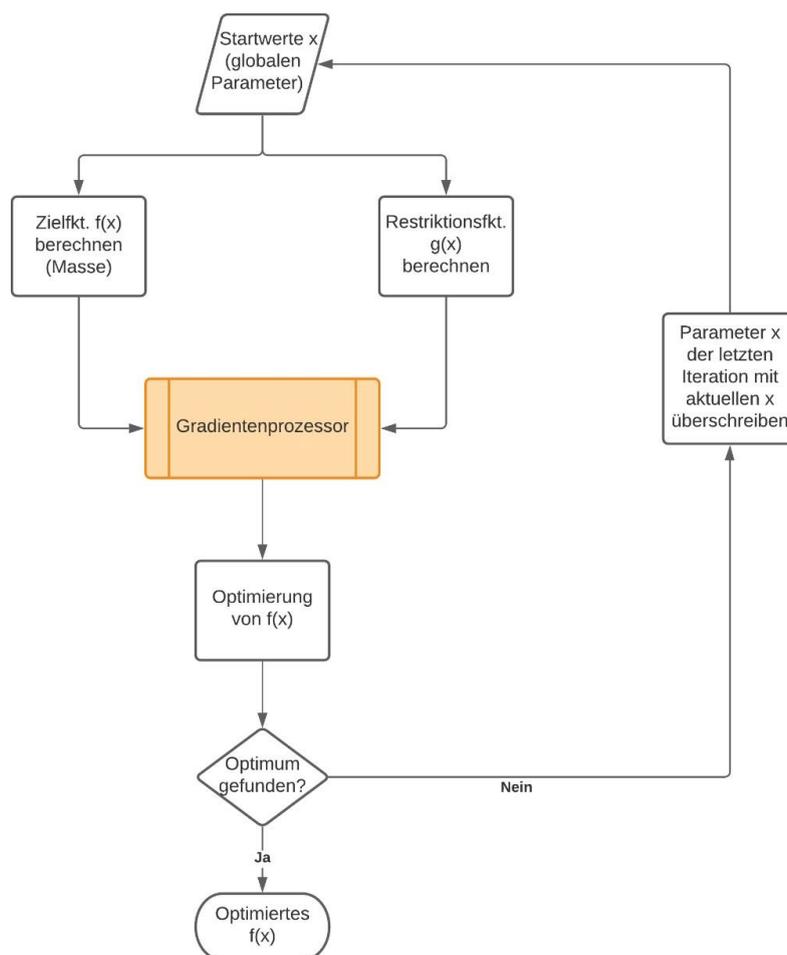


Abbildung 16: Ablauf des Optimierungsprozesses in lightworks

Abbildung 16 zeigt den veränderten Ablauf des Optimierungsprozesses, der in *lightworks* implementiert ist. Zunächst werden die Startparameter x_0 definiert. Dabei handelt es sich um die globalen Parameter, nach denen auch später abgeleitet werden soll. Im Anschluss werden Zielfunktion und Restriktionsfunktionen mit den globalen Parametern berechnet. Statt jetzt die Gradienten ausschließlich über finite Differenzen zu berechnen, wird an dieser Stelle ein Gradientenprozessor eingefügt. Dieser wird dann die partiellen Gradienten, welche in separaten Modulen berechnet werden, sammeln und die Gradienten der Restriktionen nach den globalen Parametern mithilfe der Kettenregel berechnen. Diese werden für die gradientenbasierten Optimierung, die im Anschluss stattfindet, benötigt. Ist nach dem Optimierungsschritt noch kein Abbruchkriterium erfüllt, sprich es wurde noch kein Optimum gefunden, so werden die letzten globalen Parameter x_i mit den durch die aktuelle Optimierungsiteration berechneten Parametern x_{i+1} ersetzt. Der Prozess wird nun erneut durchlaufen, nun aber nicht mehr mit den Startparametern x_0 , sondern mit den neuen Parametern x_{i+1} . Bis ein Optimum gefunden wurde, müssen die Gradienten in jeder Iteration neu berechnet werden.

Der Gradientenprozessor wird durch das Optimierungsmodell initialisiert, welches auch für die Optimierung verwendet wird. Im Optimierungsmodell sind alle nötigen Informationen enthalten, die für die Gradientenberechnung von Bedeutung sind. Dazu gehört unter anderem das Strukturmodell, in dem alle ABD-Matrizen aller Paneele enthalten sind. Die globalen Parameter, nach denen die Restriktionen abgeleitet werden sollen, sind direkt im Optimierungsmodell aufgelistet. Die Restriktionen sind alle im Restriktionsprozessor gespeichert und können durch ihn beschafft werden. Ebenfalls sind dort alle Lastfälle mit deren Kraftflüssen zu finden, mit denen die Bauteilspannungen und Deformationen berechnet werden. Zusätzlich muss zur Initialisierung die Methode angegeben werden nach welcher die Gradienten berechnet werden sollen. Eine Übersicht ist in Abbildung 17 zu sehen.

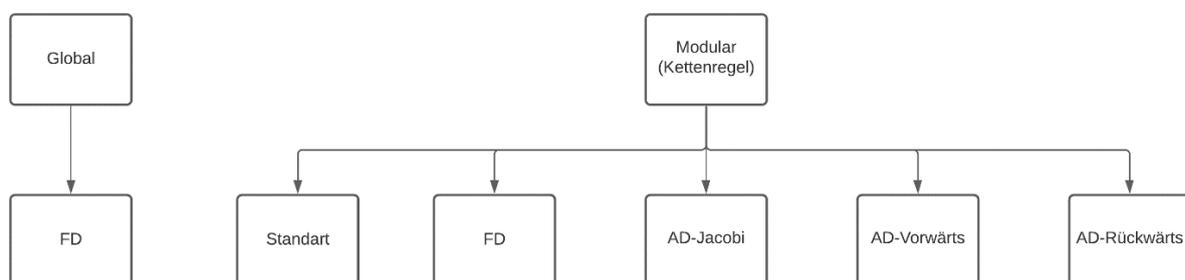


Abbildung 17: Implementierte Methoden im Gradientenprozessor

Zunächst muss die Methode gewählt werden, wie die Gradienten berechnet werden. Verfügbare Methoden sind die globale und modulare Methode. Bei der globalen Methode handelt es sich um die

herkömmliche Art die Gradienten zu berechnen. Bei der modularen Methoden werden die Gradienten mithilfe der Kettenregel aus Kapitel 3.1 berechnet. Anschließend muss die Berechnungsvorschrift für die Gradienten gewählt werden. Bei der globalen Methode sind lediglich die finiten Differenzen verfügbar, da der FE-Solver, der ebenfalls abgeleitet wird bisher nur über finite Differenzen berechnet werden kann. Bei der modularen Methode kann grundsätzlich zwischen finiten Differenzen und einer Methode der automatischen Differentiation gewählt werden. Zusätzlich ist eine Standarteinstellung möglich. Dabei werden die Gradienten mit der für die partiellen Gradienten effizientesten Methode berechnet. Diese werden aus dem Versuch aus Kapitel 4.1 bestimmt und sind in Tabelle 2 zusammengefasst.

4 Validierung

Die implementierten Methoden der Gradientenberechnung werden in diesem Kapitel miteinander verglichen. Zunächst steht im Vordergrund, wie viel Zeit die Methoden benötigen, um die Gradienten zu berechnen. Dafür werden mehrere Versuche für die verschiedenen partiellen Gradienten durchgeführt, um für jeden Gradienten die schnellste Methode ausfindig zu machen. Im Anschluss wird noch überprüft, wie groß der Fehlerterm der finiten Differenzen gegenüber den exakten Gradienten ist, welche mit den automatischen Differenzen berechnet wurden. Außerdem kann dort überprüft werden, ob die Kettenregel aus Kapitel 3.1 richtig aufgestellt wurde, indem die totalen Gradienten der Restriktionsfunktionen einmal nach den globalen Parametern abgeleitet werden. Diese Gradienten sollten identisch mit den Gradienten der Restriktionsfunktionen sein, welche ebenfalls mit der automatischen Differentiation berechnet wurden, aber diesmal modular über die Kettenregel.

4.1 Geschwindigkeit

In Kapitel 2.6.3 wurden die Methoden der finiten Differenzen und automatische Differentiation vorgestellt. Bei der automatischen Differentiation wird zwischen der Vorwärts- und Rückwärtsmethode unterschieden. Zusätzlich ist eine Jacobi-Methode in *autograd* implementiert. Es ist zunächst zu überprüfen, welche dieser Methoden für welche partiellen Gradienten die effizienteste in Hinblick auf Rechenzeit ist. Ziel hierbei ist es, die optimale Berechnungsmethode für jeden partiellen Gradienten zu bestimmen. Somit können bei der finalen Berechnung der Restriktionsgradienten mit der Kettenregel die partiellen Gradienten mit unterschiedlichen Methoden berechnet werden, um die größte Zeit bei der Gradientenberechnung einzusparen. Zu den partiellen Gradienten gehören:

- Haut-ABD-Matrix
- Stringer-ABD-Matrix
- Paneel-ABD-Matrix
- Stringerabstand-ABD-Matrix
- Restriktionen

Die Gradienten des FE-Solvers werden hier nicht mitberücksichtigt, da diese nur mit finiten Differenzen berechnet werden können. Grund dafür ist, dass es sich hierbei um einen separaten C++-Code handelt und dieser für die automatische Differenzierung ebenfalls angepasst werden muss. Das ist aber nicht Teil dieser Arbeit. Aus diesem Grund wird als FE-Solver in dieser Arbeit ausschließlich ein Dummy-FE-Solver verwendet. Dieser gibt feste Materialspannungen und -dehnungen wieder und ist somit hier unabhängig von den globalen Parametern. Somit sind die Solver-Gradienten gleich null und haben

keinen Einfluss auf die Kettenregel. In einem realen Optimierungsprozess würde ein richtiger FE-Solver für das Problem verwendet werden.

Für den Test soll zunächst die Anzahl der zu berechnenden Gradienten konstant gehalten und nur die Anzahl der Parameter erhöht werden. Hierdurch soll der Einfluss der Parameter auf die Berechnungsmethoden gezeigt werden. Zusätzlich können evtl. Parameter ausfindig gemacht werden, welche einen übermäßigen Einfluss auf die Berechnungszeit haben. Laut der Theorie sollte der Berechnungsaufwand für die finiten Differenzen linear ansteigen. Ebenso sollte sich der Aufwand der Vorwärtsmethoden mit der Anzahl der Parameter erhöhen. Einzig der Aufwand der Rückwärts- und Jacobi-Methode sollte bei Erhöhung der Parameteranzahl ungefähr konstant bleiben. Aus dem Versuch aus Kapitel 2.6.5 ist allerdings bereits bekannt, dass auch der Rechenaufwand der Rückwärtsmethode mit steigender Parameteranzahl ansteigt.

Bei diesem Versuch wird ausschließlich ein LP-Set verwendet, da dies in der späteren Optimierung eine größere Relevanz als die Laminat-Paneele hat.

4.1.1 Haut-ABD-Matrix

Bei den Gradienten der Haut-ABD-Matrizen handelt es sich um insgesamt 27 einzelne Gradienten. Für jeden Eintrag gibt es einen Gradienten. Die B-Matrix wird nicht doppelt berechnet. Für den Versuch der Haut-ABD-Matrix wird ein nicht versteiftes Paneel verwendet. Zunächst werden die Gradienten nur nach der Hautdicke berechnet. Anschließend werden dieselben Gradienten berechnet, wobei diesmal zusätzlich zur Dicke die Gradienten nach fünf LP berechnet werden. Es würde sich demnach um ein symmetrisches Laminat handeln. Zuletzt werden die Gradienten nach 12 LP und der Hautdicke berechnet.

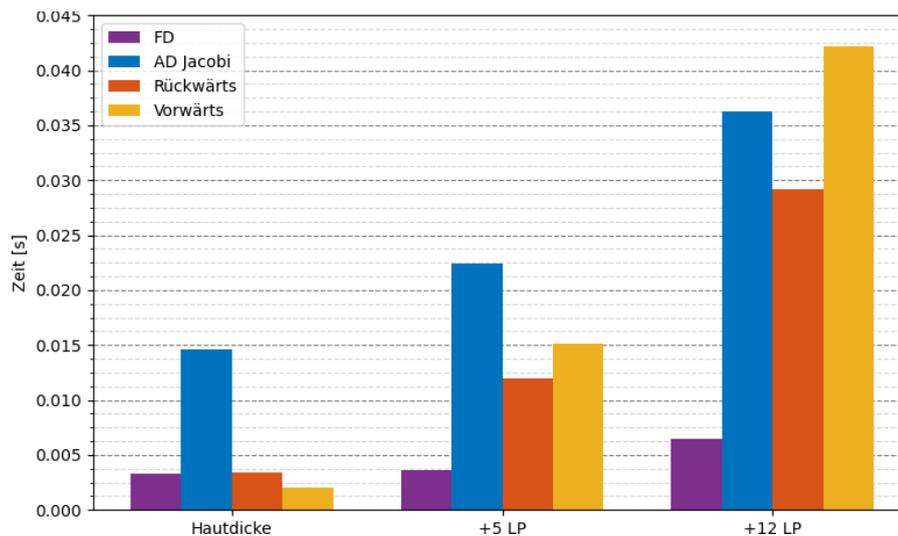


Abbildung 18: Rechenzeiten der Haut-Gradienten nach globalen Parametern (Paneel nicht versteift)

In Abbildung 18 sind Berechnungszeiten der Gradienten über die verschiedenen Parameterkombinationen für die vier Berechnungsmethoden gezeigt. Bei den finiten Differenzen ist zu erkennen, dass diese bei der Hinzunahme der fünf LP kaum eine Erhöhung der Rechenzeit aufzuweisen. Bei den zwölf LP ist auch nur eine relativ geringe Erhöhung der Rechenzeit zu erkennen. Die Vorwärtsmethode ist zunächst bei der Ableitung der Hautdicke die effizienteste, aber bei Erhöhung der Parameter steigt der Aufwand an. Dieses Verhalten ist nach der Theorie welche in Kapitel 2.6.3 vorgestellt wurde plausibel. Auch dass die Effizienz der Rückwärtsmethode sinkt, deckt sich mit dem Versuch aus Kapitel 2.6.5. Allerdings war die Erkenntnis aus dem Versuch, dass die Jacobi-Methode kaum einen Effizienzverlust mit steigender Parameteranzahl aufweist. In diesem Versuch aber ist das durchaus der Fall. Generell ist der Rechenaufwand der Jacobi-Methode im Gegensatz zu den anderen Methoden sehr hoch. Daher liegt die Vermutung nahe, dass die Jacobi-Methode für einfache Funktion besser geeignet ist, als die anderen beiden AD-Methoden. Bei komplexeren Funktionen, wie es hier der Fall ist, kann es vorteilhaft sein die Vor- oder Rückwärtsmethode anzuwenden.

Für die Gradienten der Haut-ABD-Matrix ist die finite Differenzen-Methode die effizienteste. Wird lediglich nach der Hautdicke abgeleitet, liefern die Rück- und Vorwärtsmethode ebenfalls schnell ein Ergebnis.

4.1.2 Stringer-ABD-Matrix

Auch bei dieser ABD-Matrix werden insgesamt 27 Gradienten pro versteiftes Paneel berechnet. Bei einem nicht versteiften Paneel werden keine Gradienten berechnet, da solche Paneele keine Stringer-

ABD-Matrix besitzen. Aus diesem Grund wird für diesen Versuch ein versteiftes Paneel verwendet. Da ein Schwert-Stringer den einfachsten Aufbau besitzt, wird zunächst dieses für den Versuch ausgewertet. Es werden die Gradienten der Stringer-ABD-Matrix nach Stringerdicke, -höhe, -abstand und -winkel berechnet. Außerdem erfolgt zusätzlich eine Berechnung der Gradienten nach den LP.

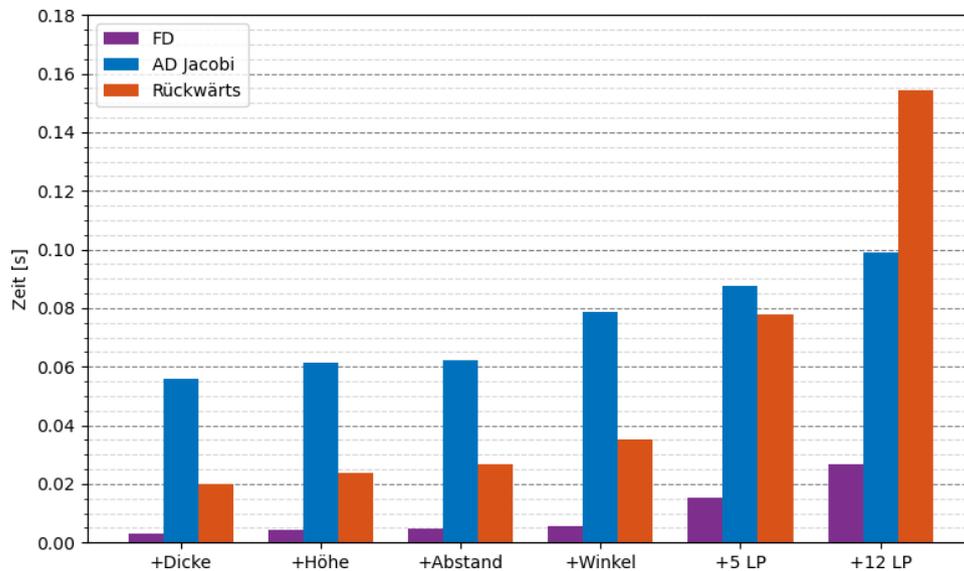


Abbildung 19: Rechenzeiten der Stringer-Gradienten nach globalen Parametern (Schwert-Paneel)

Auch hier ist die Methode der finiten Differenzen die effizienteste aller Methoden unabhängig davon wie viele Parameter ausgewertet werden.

Die Vorwärtsmethode kann bei diesen Gradienten nicht angewendet werden. Bei der Berechnung der Stringer-ABD-Matrix werden Matrixinversen gebildet, die von den oben genannten Parametern abhängig sind. Die Ableitung von diesen Inversen ist mit der Vorwärtsmethode von *autograd* nicht möglich. Diese werden in dieser Arbeit mit der Inversenfunktion von *numpy* gebildet. Weitere Informationen gehen nicht aus der Dokumentation von *autograd* hervor. In solchen Fällen muss die Rückwärts- oder Jacobi-Methode gewählt werden.

Der Anstieg des Aufwandes bei der Hinzunahme des Stringerwinkels erscheint relativ hoch im Gegensatz zu dem Anstieg bei der Hinzunahme von Stringerhöhe und -abstand, obwohl es sich bei dem Stringerwinkel auch nur um einen zusätzlichen Parameter handelt. Das ist besonders bei der Rückwärts- und Jacobi-Methode erkennbar. Bei den finiten Differenzen ist dieser Anstieg nicht sichtbar.

Bei Hinzunahme der LP ist bei der Rückwärtsmethode zu erkennen, dass hier der Aufwand stark ansteigt. Auch bei den finiten Differenzen ist ein starker Anstieg der Rechenzeit gegenüber den

Berechnungen ohne LP zu verzeichnen. Bei der Jacobi-Methode ist dieser nicht so signifikant. Grund dafür ist, dass bei der jetzigen Implementierung auch die LP der Haut ausgewertet werden. Diese haben aber keinen Einfluss auf die ABD-Matrix des Stringers und sind demnach null. Im Graphen ganz rechts werden also insgesamt zwölf LP der Haut, zwölf LP des Steges vom Stringer und die vier Stringerparameter ausgewertet. Die Jacobi-Methode erkennt, dass einige Parameter keinen Einfluss auf die Berechnung der Gradienten haben und ignoriert sie deshalb. Aus diesem Grund ist die Jacobi-Methode beim letzten Graphen effektiver als die Rückwärtsmethode, da nur 16 Parameter statt 28 Parameter ausgewertet werden.

Durch verbesserte Implementierung der Rückwärtsmethode und finiten Differenzen ist hier noch eine Effizienzsteigerung zu erwarten.

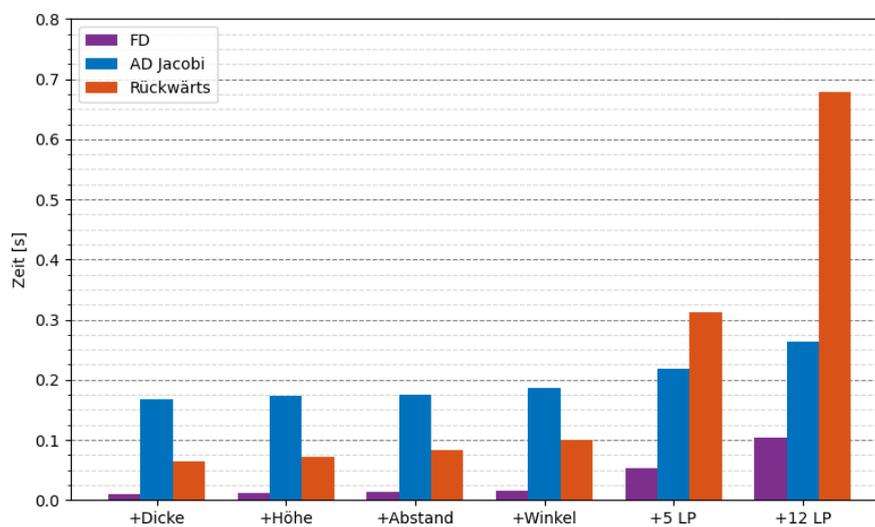


Abbildung 20: Rechenzeiten der Stringer-Gradienten nach globalen Parametern (I-Paneel)

Derselbe Versuch wurde erneut mit einem I-Stringer statt einem Schwert-Stringer durchgeführt. Dieser besitzt einen Steg, Fuß und Flansch. Da dieser Stringer mehr Stringerelemente besitzt als der Schwert-Stringer, muss die Stringer-ABD-Matrix auch nach mehr Parametern abgeleitet werden. Es werden hier nun nach drei Stringerdicken statt, wie im vorherigen Versuch mit dem Schwert-Stringer, nur nach einer Stringerdicke. Auch die Anzahl der LP verdreifacht sich. Die Anzahl der Stringerhöhe, -abstand und -winkel bleibt aber gleich.

Werden die Rechenzeiten von einem Paneel mit einem I-Stringer mit dem zuvor berechneten Schwert-Paneel verglichen, fällt auf, dass diese um ein Vielfaches höher sind. Das ist darauf zurückzuführen, dass auch mehr Parameter ausgewertet werden. Bei der Rückwärtsmethode scheint die Erhöhung der Parameteranzahl einen deutlich größeren Einfluss zu haben, als bei der Jacobi-Methode und finite Differenzen.

4.1.3 Paneel-ABD-Matrix

Bei der Berechnung der Gradienten von der Paneel-ABD-Matrix wird die Kettenregel aus Kapitel 3.1.1 verwendet. Hierfür werden zunächst die Gradienten der Haut-ABD-Matrix und Stringer-ABD-Matrix berechnet. Außerdem werden die Gradienten der Paneel-ABD-Matrix in Abhängigkeit der ABD-Matrix-Einträge der Haut und des Stringers benötigt. Wurden alle benötigten Gradienten berechnet, müssen diese nur noch im Gradientenprozessor nach der Kettenregel aufaddiert werden.

Für diese Gradienten gibt es eine weitere Methode diese zu berechnen zusätzlich zu den bisher vorgestellten Methoden. Da die Berechnung der Paneel-ABD-Matrix für ein Schwert-Paneel lediglich eine Addition von Haut-ABD-Matrix und Stringer-ABD-Matrix und somit sehr einfach ist, ist die Implementierung einer analytischen Berechnung der Gradienten ebenfalls möglich. Hierbei werden die Gradienten „per Hand“ berechnet und die Berechnung in den Programmcode implementiert.

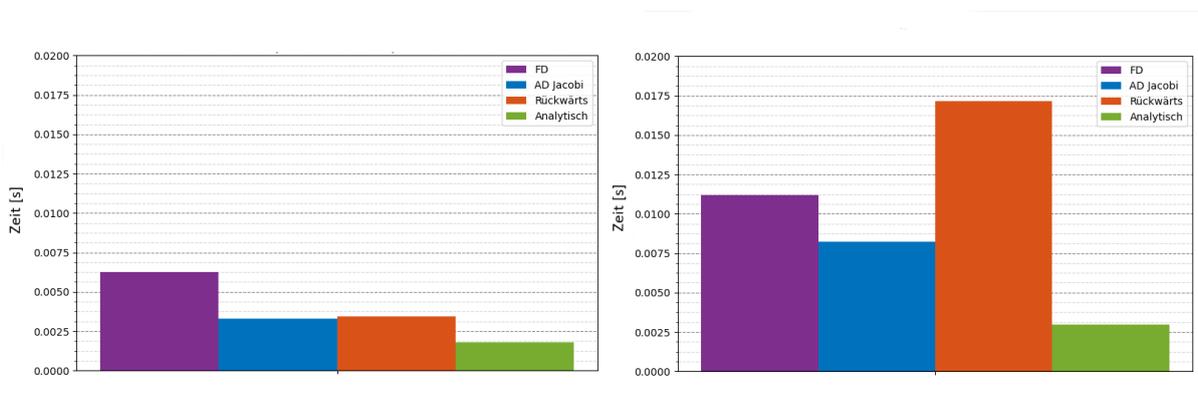


Abbildung 21: Rechenzeiten der Paneel-Gradienten nach Haut-ABD-Matrix (links: Paneel nicht versteift, rechts: I-Paneel)

Bei der Abbildung links sind die Ableitungen der Paneel-ABD-Matrix eines Schwert-Paneels nach der Haut- und Stringer-ABD-Matrix dargestellt. Es ist zu erkennen, dass die automatischen Methoden alle sehr eng beieinander liegen. Ebenfalls ist durch die Verwendung der automatischen Differenzen gegenüber den finiten Differenzen eine Halbierung der Rechenzeit möglich. Durch die Verwendung der analytischen Gradienten ist eine weitere Halbierung der Rechenzeit gegenüber den automatischen Differenzen möglich. Werden aber die Absolutbeträge mit denen von den Stringer-Gradienten des vorherigen Tests verglichen, dann gibt es hier kaum einen Unterschied welche Methode für diese Gradienten verwendet wird.

In der rechten Abbildung sind die gleichen Gradienten eines I-Paneels dargestellt. Hier werden zusätzlich die Gradienten der Paneel-ABD-Matrix nach der Fuß-ABD-Matrix berechnet. Da der Schwert-Stringer keinen Fuß besitzt, fällt er dort weg. Hier ist der Vorteil der analytisch berechneten Gradienten besser zu erkennen. Die Rechenzeit aller anderen Methoden hat sich verdoppelt, bei der Rückwärtsmethode sogar fast verfünffacht, wobei die Rechenzeit der analytischen Gradienten sich

kaum verändert hat. Aber auch hier hat die Wahl der Methode für diese partiellen Gradienten kaum einen Einfluss auf den totalen Gradienten der Pannel-ABD-Matrix nach den globalen Parametern, da die Unterschiede der Absolutbeträge vernachlässigbar klein sind.

4.1.4 Stringerabstand-ABD-Matrix

Auch bei dieser ABD-Matrix ist eine modulare Berechnung der Gradienten mithilfe der Kettenregel möglich. Hierfür werden die Gradienten der Haut- und Fuß-ABD-Matrix nach den globalen Parametern benötigt und die Gradienten der Stringerabstand-ABD-Matrix nach Haut- und Fuß-ABD-Matrix. Wie auch im vorherigen Versuch können letztere analytisch berechnet werden.

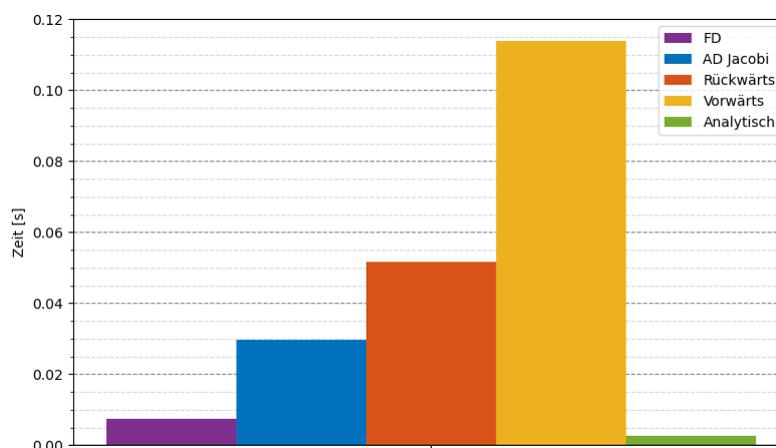


Abbildung 22: Rechenzeiten der Stringerabstands-Gradienten nach Haut- und Fuß-ABD-Matrix (I-Panel)

Abbildung 22 zeigt die Rechenzeit der Gradienten der Stringerabstand-ABD-Matrix nach der Haut- und Fuß-ABD-Matrix. Hier ist die Vorwärtsmethode wieder anwendbar, da keine Inversenberechnung stattfindet. Zu erkennen ist, dass diese aber einen großen Rechenbedarf besitzt. Grund dafür ist die Anzahl an Parametern, nach denen die Gradienten gebildet werden. Aus der Summe der Einträge der Haut- und Fuß-ABD-Matrix ergeben sich 54 abzuleitende Parameter. Um Rechenzeit zu sparen, ist es hier sinnvoll, die Gradienten analytisch abzuleiten. Auch die finiten Differenzen erweisen sich als relativ effizient. Demnach wird die geringste Rechenzeit für die Stringerabstand-ABD-Matrix benötigt, wenn die Gradienten von Haut- und Fuß-ABD-Matrix mit finiten Differenzen und die Stringerabstand-ABD-Matrix analytisch berechnet werden.

4.1.5 Restriktionen

4.1.5.1 Rampenwinkel-Restriktion

Diese Restriktion beschränkt die Dickenänderung zwischen zwei Paneelen. Demnach werden für diesen Test zwei Paneele benötigt. Die Restriktion ist allein von der Hautdicke der beiden Paneele abhängig. Die Gleichung hierfür lautet:

$$g = \frac{|t_1 - t_2|}{d \cdot p_{max}} - 1 \quad (4.1)$$

Auch hier ist durch die geringe Komplexität der Gleichung eine analytische Ableitung möglich. Diese lautet:

$$\begin{aligned} \frac{\partial g}{\partial t_1} &= 1 \\ \frac{\partial g}{\partial t_2} &= -1 \end{aligned} \quad (4.2)$$

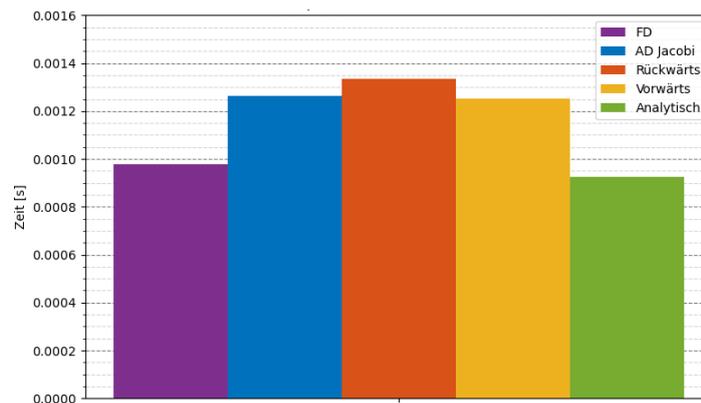


Abbildung 23: Rechenzeiten der Rampenwinkel-Restriktionen-Gradienten nach Hautdicke

Ähnlich wie im vorherigen Test ist die analytische Berechnung am schnellsten, die Wahl einer anderen Methode hat aber keinen großen Einfluss auf die absolute Berechnungszeit der Gradienten dieses Kriteriums. Hier hat auch der Typ des Paneels keinen Einfluss auf die Berechnungszeit, da bei jedem nur die Dicke der Haut ausgewertet wird und die Stringerelemente keinen Einfluss haben.

4.1.5.2 Lamination-Parameter-Gültigkeitsbereich

In Kapitel 2.2 wurde erklärt, dass nur bestimmte LP-Kombinationen erlaubt sind. Durch diese Restriktion wird gewährleistet, dass das durch die Optimierung evaluierte LP-Set eine valide Kombination aus LP darstellt. Zum Lösen dieser Restriktion wird ein LP-Design-Raum bereitgestellt gemäß Gleichung 2.21. Dieser Design-Raum wird mit dem LP-Parameter-Set multipliziert. Somit ist die

LP Restriktion lediglich von den LP abhängig. Leitet man die Restriktion nach dem LP-Set ab und erstellt daraus eine Jacobi-Matrix mit den Gradienten als Zeilen und LP als Spalten, ist festzustellen, dass diese identisch mit dem LP-Design-Raum ist. Die letzte Spalte wird dabei abgezogen, da sie nicht von den LP abhängig ist.

Demnach ist eine analytische Ableitung dieser Restriktion einfach umzusetzen, indem bei der Berechnung der Gradienten lediglich der LP-Design-Raum ohne die letzte Spalte wiedergegeben wird.

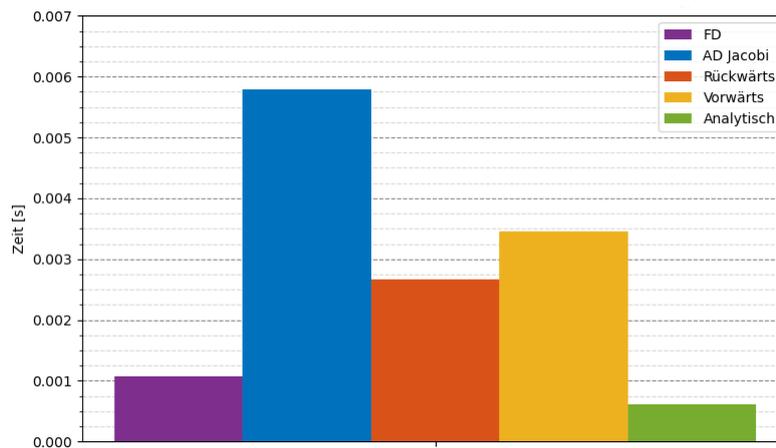


Abbildung 24: Rechenzeiten der LP-Gültigkeitsbereichs-Gradienten nach LP (Paneel nicht versteift)

Auch bei diesen Restriktionen soll untersucht werden welche der vorgestellten Methoden sich als am effizientesten erweist. Die analytische Bestimmung der Gradienten ist entsprechend der Erwartungen die Schnellste. Die Berechnung über finiten Differenzen dauert ungefähr doppelt so lange und die Methoden der automatischen Differentiation benötigen am längsten. Werden die absoluten Zeiten miteinander verglichen, so gibt es auch hier keinen großen Unterschied welche Methode für die Berechnung verwendet wird. Der Rechenzeiten steigen linear mit der Anzahl der Stringerelemente des Paneels an. Die Berechnung der Restriktion von einem I-Paneel ist also viermal so hoch für jede Methode, außer der analytischen, wie für ein nicht versteiftes Paneel. Im Umkehrschluss bedeutet dies, dass je mehr Stringerelemente der Stringer des Paneels besitzt, desto mehr lohnt sich die analytische Berechnung der Gradienten.

4.1.5.3 Versagensrestriktionen

Die Versagenskriterien stellen die komplexesten Kriterien dar. Deswegen ist hier keine analytische Berechnung der Gradienten sinnvoll. Der Aufwand diese zu implementieren ist zu hoch und das Risiko von Fehlern ebenfalls. Außerdem werden diese Gradienten modular über die Kettenregel berechnet.

Auch hier soll die Geschwindigkeit der vorgestellten Methoden überprüft werden. Die einsetzbaren Methoden sind: finite Differenzen, Rückwärtsmethode und Jacobi-Methode. Alle partiellen Gradienten der Kettenregel werden mit der ausgewählten Methode berechnet. Dazu zählen die Gradienten der Haut-, Stringer- und Stringerabstand-ABD-Matrix, sowie die Gradienten des FE-Solvers, welche aber wie oben schon beschrieben keinen Einfluss auf die Kettenregel haben.

Es werden die Gradienten der Versagensrestriktionen nach dem FE-Solver benötigt, also nach den Materialspannungen und Dehnungen. Ebenso werden die Gradienten der Restriktionen nach Haut- und Paneel-ABD-Matrix benötigt. In einem weiteren Versuch soll überprüft werden welche der drei Methoden am schnellsten zu einem Ergebnis kommt.

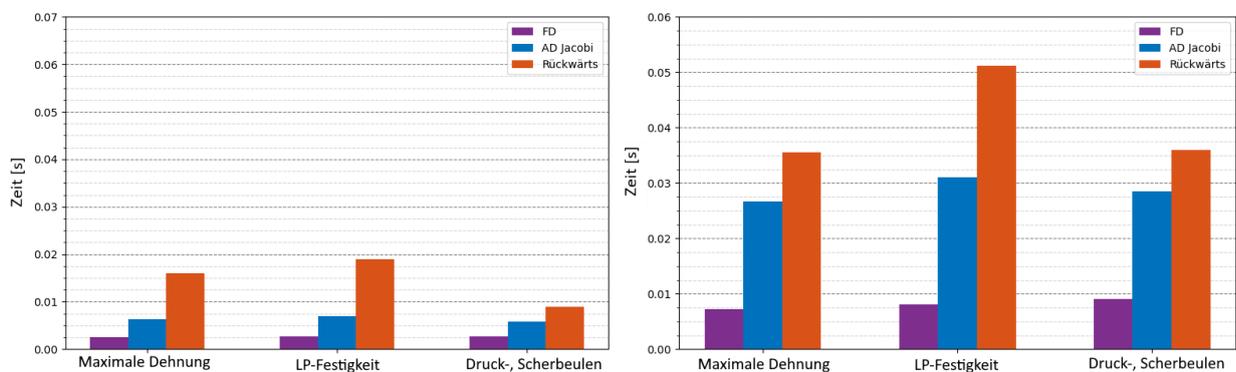


Abbildung 25: Rechenzeit der versagensrestriktions-Gradienten nach Spannung (links: Paneel nicht versteift, rechts: I-Paneel)

Die Gradienten für die Versagenskriterien wurden für drei verschiedene Kriterien ausgewertet, um eventuelle Unterschiede zwischen den Kriterien herauszufinden. Die Kriterien wurden hierbei für jedes Stringerelement ausgewertet, sowie auch global für das ganze Paneel. In der linken Abbildung wurden die Kriterien für ein nicht versteiftes Paneel, in der rechten für ein I-Paneel berechnet und nach den Materialspannungen abgeleitet. Da das I-Paneel mehr Elemente wie das nicht versteifte Paneel besitzt, werden dort auch mehr Restriktionen ausgewertet, da jedes Element auf Versagen überprüft wird. Demnach ist es plausibel, dass dort auch die Rechenzeiten höher sind. Wie auch in den meisten vorherigen Tests sind die finiten Differenzen schneller als die automatische Differentiation. Bei den Ableitungen nach den Materialdehnungen ergeben sich ähnliche Resultate.

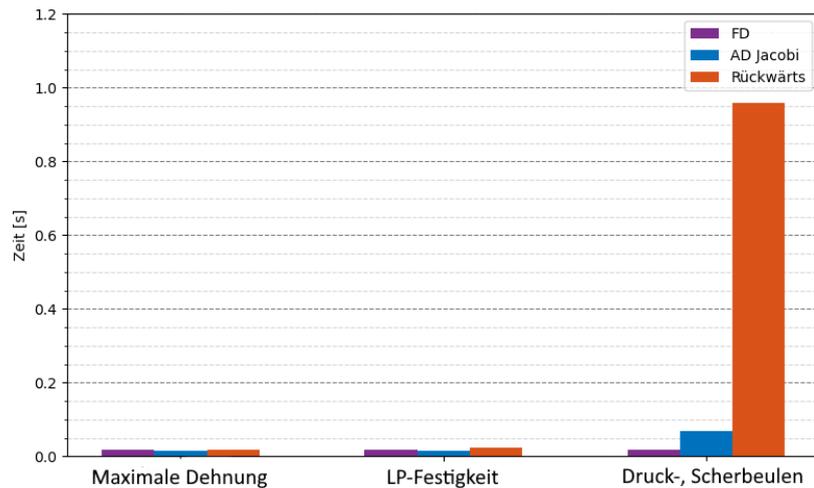


Abbildung 26: Rechenzeiten der Versagensrestriktions-Gradienten nach Haut-ABD-Matrix (I-Paneel)

In dieser Abbildung wird die Rechenzeit der Gradienten der Restriktionen nach der Haut-ABD-Matrix dargestellt. Die Rechenzeiten der beiden Festigkeitskriterien sind deshalb so niedrig, weil diese unabhängig von der Haut-ABD-Matrix sind. Besonders die Rückwärtsmethode weist eine ungewöhnlich hohe Rechenzeit gegenüber den anderen Methoden auf. Aus diesem Grund sollte die Rückwärtsmethode nicht für die Versagenskriterien verwendet werden. Am schnellsten sind hier wieder die finiten Differenzen.

Tabelle 2: Zeitgünstigste Methoden für jeden partiellen Gradienten

	Zeitgünstigste Methode	Zeitgünstigste AD-Methode	Absolute Zeitdifferenz In Sekunden	Relative Zeitdifferenz
Haut-ABD	FD	AD-Rückwärts	0.022	352%
Stringer-ABD	FD	AD-Jacobi	0.11	440%
Paneel-ABD	Analytisch	AD-Jacobi	0.0144	460%
Stringerabstand-ABD	Analytisch	AD-Jacobi	0.0275	1100%
Lamination-Parameter-Gültigkeitsbereich	Analytisch	AD-Rückwärts	0.00225	450%
Rampenwinkel-Restriktion	Analytisch	AD-Jacobi	0.00035	39%
Versagensrestriktion	FD	AD-Jacobi	0.02	200%

Welche Methode für die verschiedenen Gradienten die schnellst ist, ist zusammenfassend in Tabelle 2 aufgeführt. Zusätzlich ist angegeben, welche Methode der automatischen Differentiation am schnellsten zu einem Ergebnis gekommen ist. Falls genaue Gradienten benötigt werden, sollte eher

auf die automatischen als auf die finiten Differenzen zurückgegriffen werden. Die analytischen und automatischen Differenzen kommen exakt auf dasselbe Ergebnis. Können Gradienten also analytisch berechnet werden, gibt es keine Begründung für die Verwendung von automatischen oder finiten Differenzen, da die analytischen Gradienten für jeden partiellen Gradienten die beste Rechenzeit aufwiesen. Zur Vervollständigung ist aber auch bei solchen Gradienten die schnellste Methode der automatischen Differenzen aufgeführt. Falls doch die automatischen Differenzen statt der aufgeführten optimalen Methode verwendet werden, ist die absolute Zeit angegeben, die durch das Berechnen der Gradienten zusätzlich benötigt wird. Die Zeiten wurden jeweils an einem I-Paneel und bei Haut- und Stringergradienten mit allen globalen Parametern gemessen. Bei den Versagensrestriktionen wurden die Zeit an den Gradienten nach den Materialspannungen gemessen. Zusätzlich wird die zusätzliche relative Rechenzeit angegeben, welche benötigt wird, falls die AD-Methode statt derzeitgünstigsten Methode gewählt wird.

4.1.6 Vergleich globale und modulare Gradienten

In einem letzten Versuch sollen die modularen Gradienten, welche mit der Kettenregel berechnet werden, mit der bisherigen Berechnungsmethode der Gradienten verglichen werden. Mit der bisherigen Methode wurden die Gradienten der Restriktionen direkt mit finiten Differenzen nach den globalen Parametern berechnet. Nachteil dabei ist, dass alle Restriktionen für alle Parameter berechnet werden. Es werden also auch die Parameter von dem einem Paneel auf Restriktionen eines anderen Paneels angewendet. Bei der modularen Berechnung werden die Gradienten paneelweise berechnet und dabei auch nur für die Parameter, die aus diesem Paneel stammen.

Zum besseren Verständnis soll ein kurzes Gedankenexperiment dienen. Es sollen die Restriktionsgradienten von fünf Paneelen berechnet werden. Jedes Paneel besitzt einen Parameter und eine Restriktion. Das bedeutet, dass insgesamt fünf Restriktionen ausgewertet werden. Mit der globalen Methode werden die fünf Restriktionen für jede der fünf Parameter angewendet, also insgesamt 25 Auswertungen. Bei der Ableitung des FE-Solvers ist dies auch nötig, da Parameter von benachbarten Paneelen gegenseitigen Einfluss auf die Spannung und Deformation auf das Paneel haben können. Bei den anderen partiellen Gradienten haben nur die Parameter des betrachteten Paneels auf diesen Einfluss. Es ergeben sich mit solchen Gradienten viele Nulleinträge in der Jacobi-Matrix. Bei der modularen Berechnung werden die Restriktionen nur für die Parameter ausgewertet, die auch für diesen relevant sind. So ergeben sich insgesamt nur fünf Gradientenberechnungen. Der FE-Solver wird bei der modularen Berechnung identisch abgeleitet wie bei der globalen. Dennoch ist durch die modulare Methode eine Verringerung der Rechenzeit zu erwarten.

In diesem letzten Test dieses Kapitels soll nun die Auswirkung dieser beiden Rechenmethoden auf die Rechenzeiten der Gradienten gezeigt werden. Hier bleibt die Parameterkombination konstant und es wird nur die Anzahl der Paneele in dem Paneel-Set für den Test verändert. In diesem Test wird der Solver vernachlässigt und es wird ein Dummy-Solver zur Berechnung der Gradienten verwendet. Die hier berechneten Zeiten weichen also von den Rechenzeiten der Gradienten bei einer richtigen Optimierung ab. Zum Vergleich der Gradienten kann durch diesen Versuch trotzdem eine Aussage getroffen werden, da bei den globalen, sowie modularen Gradienten der Solver nach den gleichen Methoden abgeleitet wird und dort die Rechenzeiten bei beiden Methoden identisch sein werden. Die Gradienten wurden bei allen Methoden sequentiell berechnet. Durch eine Parallelisierung der Berechnung ist eine Senkung der Berechnungszeit möglich.

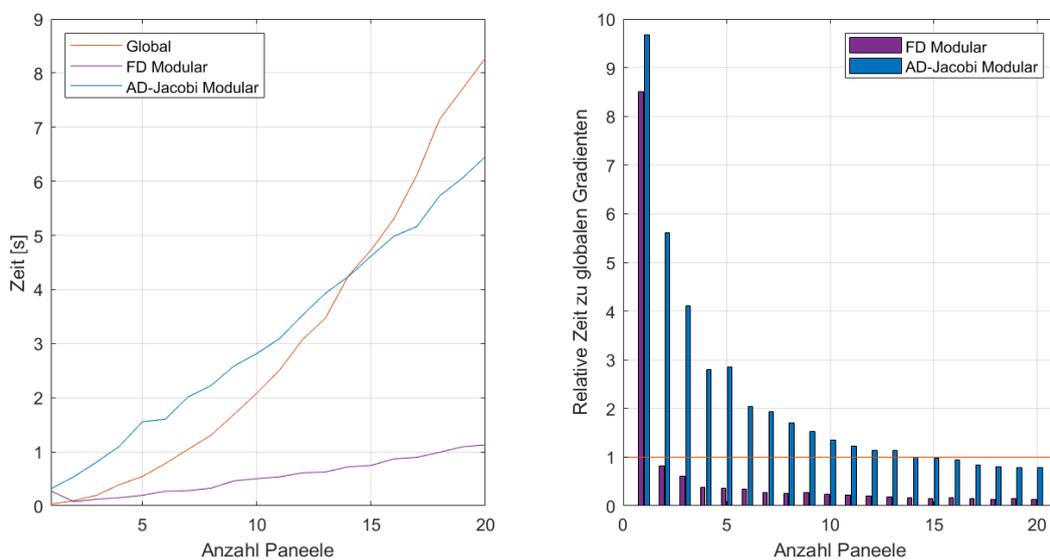


Abbildung 27: Vergleich der Rechenzeiten zwischen globalen und modularen Gradienten (Panel nicht versteift)

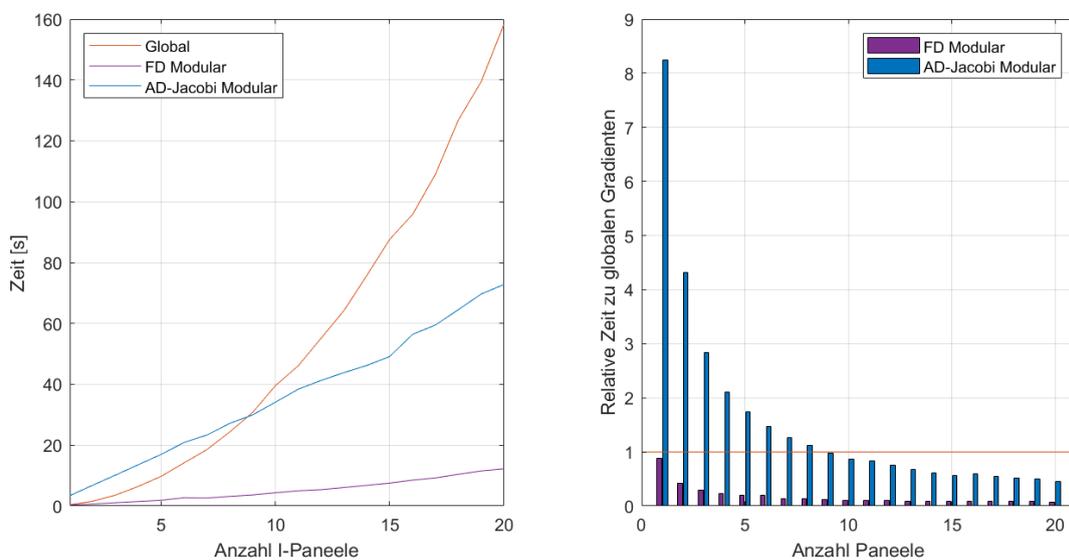


Abbildung 28: Vergleich der Rechenzeiten zwischen globalen und modularen Gradienten (I-Panel)

In den obigen Abbildungen sind die Rechenzeiten für die Restriktionsgradienten für ein nicht versteiftes Paneel-Set und ein I-Paneel-Set dargestellt. Die Berechnungen wurden einmal für die globale und modulare Methode durchgeführt. Die partiellen Gradienten wurden bei der modularen Methode einmal mit den in Tabelle 2 aufgeführten Methoden und einmal mit der Jacobi-Methode statt den finiten Differenzen durchgeführt. Erstere ist als FD-Modular im Grafen ersichtlich und letztere als AD-Modular. Die Gradienten wurden hier nach der Hautdicke, Stringerdicke, LP und Stringehöhe berechnet.

Sofort ist der zeitliche Vorteil der modularen Gradienten im Gegensatz zu den modularen Gradienten zu erkennen. Rechts neben dem Graphen, welcher die absolute Rechenzeit anzeigt, sind die relativen Rechenzeiten von den beiden modularen Methoden im Gegensatz zu den globalen Methoden gezeigt. Liegen die Balken unter der Linie bedeutet dies, dass ab dieser Paneelanzahl die modulare Methode schneller ist als die globale Methode.

Die globale Methode besitzt eine quadratische Skalierung des Rechenaufwands mit Steigerung der Paneelanzahl, da hier nicht nur die Anzahl der Restriktionen mit der Anzahl der Paneele ansteigt, sondern auch die Anzahl der Parameter, die pro Restriktion ausgewertet werden. Bei der modularen Methode erhöht sich mit der Anzahl der Paneele lediglich die Anzahl der Restriktionen und die Anzahl der Parameter bleibt pro Restriktion unverändert. Daher die lineare Skalierung des Verlaufs.

Die modulare Berechnung mit der Jacobi-Methode ist zunächst langsamer als die globalen finiten Differenzen. Durch den Vorteil der modularen Differenzen steigt die Rechenzeit dennoch nicht so stark an wie die globalen Differenzen. Dadurch gibt es eine Anzahl an Paneelen ab der auch die Berechnung der Gradienten mit der Jacobi-Methode schneller ist als die globalen finiten Differenzen.

Durch die Implementierung der modularen Gradienten mit der Kettenregel konnte die Geschwindigkeit gegenüber den globalen Gradienten verbessert werden. Hierbei ist die Methode der finiten Differenzen effektiver als die automatischen Differenzen.

4.2 Genauigkeit

Im letzten Kapitel wurde die Geschwindigkeit der finiten Differenzen den der automatischen Differentiation gegenübergestellt. In diesem Kapitel wird nun validiert, ob die Werte der berechneten Gradienten der verschiedenen Methoden übereinstimmen. Es werden die Gradienten der globalen finiten Differenzen, der modularen finiten Differenzen und der modular automatischen Differenzen berechnet. Es sind Abweichungen zu erwarten, da es sich bei der automatischen Differentiation um eine exakte Lösung handelt und bei den finiten Differenzen um eine numerische Näherung der

Gradienten. Um zu überprüfen, wie weit die berechneten Gradienten der beiden Methoden auseinander liegen, werden für das LP-Paneel-Set, welche schon für die Tests im vorherigen Kapitel verwendet wurde, jeweils die partiellen Gradienten mit finiten Differenzen und automatischer Differentiation berechnet. Da sich die Berechnungsmethoden der automatischen Differentiation nur im Rechenaufwand unterscheiden und das Ergebnis dieser beiden Methoden stets identisch ist, wird es keinen Unterschied geben, welche der Methoden hier verwendet wird. Wurden die Gradienten berechnet, werden diese miteinander verglichen und der maximal auftretende relative Fehler in diesem Paneel-Set bestimmt. Um möglichst viele Vergleichswerte zu erhalten werden die Gradienten nach allen globalen Parametern gebildet.

Es wird davon ausgegangen, dass die global mit automatischer Differentiation berechneten Gradienten das korrekte Ergebnis liefern. Daher dienen diese Gradienten als Referenz für die anderen Gradienten. Der Vergleich der automatisch berechneten Gradienten dient dazu, um zu überprüfen, ob die Kettenregel richtig aufgestellt wurde.

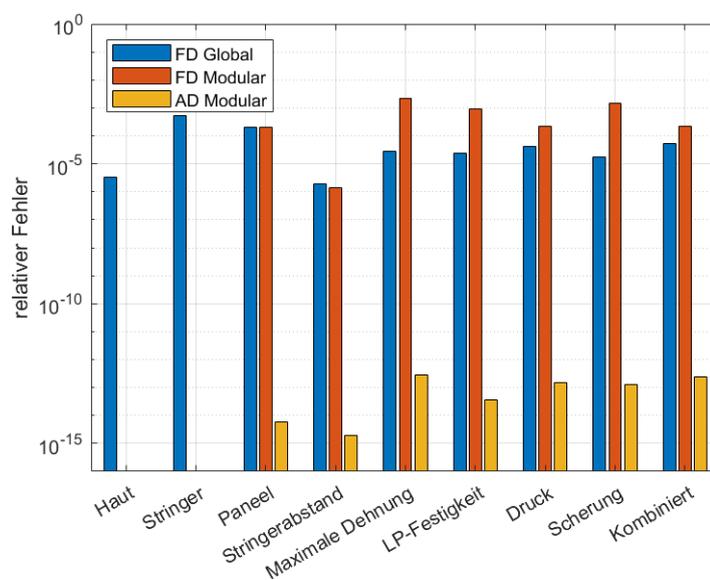


Abbildung 29: Relative Abweichungen von finiten Differenzen ggü. automatischer Differentiation

In Abbildung ist der maximal aufgetretene relative Fehler beim Vergleich der Gradienten dargestellt. Für die Haut- und Stringergradienten wurden ausschließlich die globalen finiten Differenzen mit den modularen automatischen Differenzen verglichen, da dort keine Gradienten über die Kettenregel berechnet werden. Es wurde eine logarithmische Skala für die Ordinate gewählt, damit die Werte der relativen Fehler der modular automatisch berechneten Gradienten sichtbar sind. Bei den hier berechneten Gradienten handelt es sich um die Gradienten der ABD-Matrizen von Haut, Stringer, Paneel und Stringerabstand. Zusätzlich wurden die Gradienten von den Festigkeits- und

Stabilitätskriterien berechnet. Zur besseren Übersicht wurde sich beim Vergleich der Restriktionsgradienten auf die für die Optimierung relevantesten Kriterien beschränkt. Dazu gehören zwei Festigkeitskriterien. Beim ersten wird das Bauteil bei maximal auftretender Dehnung des Lastfalls auf Versagen überprüft. Das zweite Festigkeitskriterium, welches hier auf Genauigkeit getestet wurde, evaluiert das Versagen des Bauteils speziell für LP-Paneele. Die letzten drei Kriterien sind Stabilitätskriterien und überprüfen das Bauteil auf Beulen durch Druck, Scherung und einer kombinierten Druck-Scher-Beanspruchung.

Der Fehler der zwischen den global und modular automatisch berechneten Gradienten liegt in der Größenordnung $1e^{-14}$. Durch den kleinen Wert kann angenommen werden, dass die Gradienten der globalen und modularen automatischen Differentiation exakt übereinstimmen. Somit kann bestätigt werden, dass die Gradienten für die LP-Paneele korrekt über die Kettenregel berechnet werden. Der relative Fehler, der bei den finiten Differenzen aufgetreten ist, liegt je nach Typ des Gradienten zwischen $1e^{-2}$ und $1e^{-6}$.

Werden die global berechneten Gradienten mit finiten Differenzen mit den modular berechneten Gradienten mit finiten Differenzen verglichen, ist erkennbar, dass die Genauigkeit der modular berechneten Gradienten kleiner ist. Die Größe des Unterschiedes zwischen den relativen Fehlern liegt zwischen Faktor 100 und 1000. Es ist also ein deutlicher Unterschied zu erkennen. Der Grund dafür liegt darin, dass bei jeder Berechnung der partiellen Gradienten mit finiten Differenzen ein Fehler entsteht. Bei Aufsummierung der partiellen Gradienten werden ebenfalls die entstehenden Fehler aufsummiert. Das erklärt den geringen Unterschied der beiden Methoden bei den Paneel- und Stringerabstand-ABD Gradienten, da dort weniger partielle Gradienten in der Kettenregel berechnet werden. Möglich ist auch, dass sich die Fehler der partiell berechneten Gradienten gegenseitig mehr oder minder aufheben. So kann es auch vorkommen, dass die modulare Berechnung etwas genauer ist als die globale Berechnung der Gradienten. Dies ist bei der Stringerabstand-ABD-Matrix der Fall.

5 Anwendung

Durch die Untersuchungen im letzten Kapitel ist der Genauigkeitsunterschied zwischen den mit finiten Differenzen berechneten und den mit automatischer Differentiation berechneten Gradienten zu erkennen. Ebenfalls hat sich ergeben, dass der relative Fehler durch die modularen finiten Differenzen gegenüber den globalen ansteigen kann. Vor allem war dies bei den Gradienten der Kriterien ersichtlich.

In diesem Kapitel wird eine weitere Untersuchung durchgeführt, in der überprüft wird, welchen Einfluss die Genauigkeit der Gradienten auf den Optimierungsprozess hat. Dafür wird derselbe Optimierungsprozess mehrmals durchlaufen, wobei die Methode zur Ermittlung der Gradienten bei jedem Durchlauf variiert wird. Um zu klären, ob die Genauigkeit der Gradienten einen Einfluss auf die Endergebnisse der Optimierung hat, werden diese miteinander verglichen. Weiterhin soll die Konvergenzgeschwindigkeit der Optimierungen untersucht werden. Es ist möglich, dass die Methoden eine unterschiedliche Anzahl an Iterationen benötigen, um auf ein Optimum zu schließen. Dadurch kann es zu einem zusätzlichen Geschwindigkeitsvorteil zu dem kommen, der schon durch die modulare Berechnung existiert. Ebenfalls ist es möglich, dass der Optimierungsprozess bei einigen Methoden nicht konvergiert und somit zu keinem Ergebnis kommt.

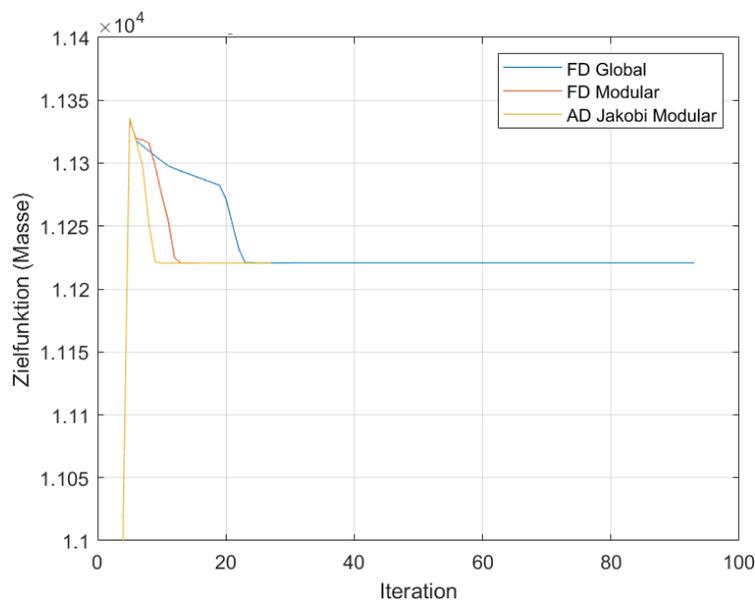


Abbildung 30: Konvergenzverlauf globaler und modularer Gradienten (Paneel nicht versteift)

Für diese Optimierung wurde ein Paneel-Set aus 20 nicht versteiften Paneelen verwendet. Als Zielfunktion wird die Masse des gesamten Paneel-Sets minimiert. Die Abbruchkriterien wurden hier so gewählt, dass bei Massenänderungen oder Parameteränderungen zwischen den Iterationen unter $1e^{-6}$ die Optimierung beendet wird.

Zunächst ergibt sich bei jeder Optimierung mit den unterschiedlichen Methoden dieselbe Masse als Endergebnis. Wichtig ist hier aber wie schnell die Methoden konvergieren. Die Optimierung mit den global berechneten Gradienten konvergierte nach 93 Iterationen. Die modularen finiten Differenzen konvergierten nach 20 Iterationen, mit automatischen Differenzen nach 27 Iterationen. Durch die modulare Berechnung war es somit möglich, die Anzahl der nötigen Iterationen deutlich gegenüber den globalen Gradienten zu reduzieren und somit einen zusätzlichen Geschwindigkeitsvorteil zu erzielen.

Die Rechenzeit wird durch die modularen Gradienten, also durch zwei Faktoren verbessert. Zum einen benötigt der Optimierungsprozess um zu konvergieren dadurch weniger Iterationen, zum anderen wird pro Iteration weniger Zeit für die Berechnung der Gradienten benötigt. Aus Abbildung 27 geht hervor, dass auch mit den automatischen Differenzen ab einer Anzahl von 14 Paneelen weniger Zeit pro Iteration benötigt wird als mit den globalen finiten Differenzen. Durch diese beiden Faktoren konnte in diesem Fall die Optimierungszeit verringert werden. Bei den modularen automatischen Differenzen waren es ca. 76 % Verringerung der benötigten Zeit und durch die modularen finiten Differenzen sogar 97 %. Anzumerken ist, dass auch hier ein Dummy-Solver eingesetzt wurde. Wie schon erwähnt, ändert sich die Ableitungsmethode des FE-Solvers durch die modularen Gradienten nicht. In einem richtigen Optimierungsprozess mit einem echten FE-Solver würde der Geschwindigkeitsvorteil geringer ausfallen.

Auffällig bei den globalen Gradienten ist, dass diese nur ein paar Iterationen mehr benötigten als die modularen Gradienten, um auf das Endergebnis zu konvergieren. Im Anschluss laufen aber noch viele Iterationen, bei denen die Zielfunktion kaum verändert wird, bis eines der Abbruchkriterien erreicht wird.

Derselbe Test wurde ebenfalls für ein Set von versteiften Paneelen durchgeführt, um dort das Konvergenzverhalten der drei Methoden zu untersuchen.

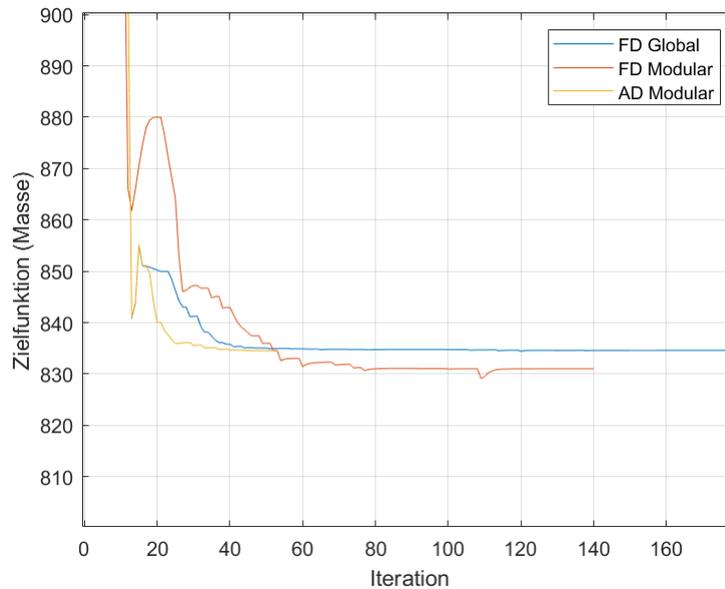


Abbildung 31: Konvergenzverlauf globaler und modularer Gradienten (I-Panel)

Für versteifte Paneele scheint die Methode der automatischen Differentiation am schnellsten zu konvergieren. Das Endergebnis ist dasselbe wie bei der herkömmlichen Methode der finiten globalen Gradienten. Das Ergebnis der finiten modularen Gradienten weicht etwas von den anderen beiden ab. Ob es bei jedem Optimierungsproblem zu einem besseren Optimum kommt, wie bei diesem Versuch, ist hieraus nicht erkennbar. Denkbar wäre ebenfalls, dass das konvergierte Optimum, welches mit den finiten modularen Differenzen bestimmt wurde, über dem der anderen beiden Methoden liegen kann. Der Grund für die Abweichung ist hier der durch die Gradientenberechnung aufgetretene Fehler. Dies könnte sich aber auch negativ auf das Ergebnis auswirken.

Durch die automatische modulare Methode gegenüber den finiten globalen Differenzen konnte die Zeit um 48 % reduziert werden mit den finiten modularen Gradienten um 77 %.

6 Fazit

6.1 Zusammenfassung

Das Ziel dieser Arbeit war es, neue Methoden der Gradientenberechnung für die Software *lightworks* des DLR zu implementieren, um den Rechenaufwand beim Optimierungsprozess zu minimieren. Im herkömmlichen Optimierungsprozess wurden die Gradienten der Restriktionsfunktionen direkt nach den globalen Parametern abgeleitet. Dabei wurde die Methode der finiten Differenzen verwendet. Zusätzlich zur globalen Gradientenbestimmung wurde in dieser Arbeit die modulare Gradientenbestimmung implementiert. Hierbei werden die totalen Gradienten der Restriktionsfunktionen in partielle Gradienten aufgeteilt. Ist nun nach den totalen Gradienten gefragt, werden zunächst die partiellen Gradienten berechnet und mit der Kettenregel zu den totalen Gradienten aufaddiert. Zusätzlich zu der Berechnung über finiten Differenzen wurde die Berechnung mit automatischer Differentiation implementiert. Dafür wurde die Python-Bibliothek *autograd* verwendet. Aus dieser Bibliothek wurden drei verschiedene Funktionen in *lightworks* implementiert um Gradienten zu berechnen. Zusätzlich war es bei einigen partiellen Gradienten möglich diese analytisch zu berechnen.

In einem nächsten Schritt mussten für die modulare Methode die Kettenregeln aufgestellt werden. Dafür wurde der Optimierungsprozess in einzelne Module eingeteilt. Für jedes Modul war es möglich die dazugehörigen partiellen Gradienten zu definieren. Aus den partiellen Gradienten wurden dann die Kettenregeln für das Strukturmodell, den FE-Solver und den Restriktionsprozessor aufgestellt.

Im Anschluss wurden die implementierten Methoden auf Geschwindigkeit miteinander verglichen. Aus den Geschwindigkeitsversuchen ergab sich, dass die Gradientenberechnung mit finiten Differenzen in den meisten Fällen schneller war als über automatischer Differentiation. Bei den partiellen Gradienten, bei denen die analytische Methode angewendet wurde, war diese die ausnahmslos schnellste. Aus diesem Versuch war es dann möglich die effizienteste Methode für jeden partiellen Gradienten zu bestimmen. Außerdem wurde die Genauigkeit der implementierten Methoden in einem weiteren Versuch erprobt. Durch diesen Versuch konnte bestätigt werden, dass die Kettenregeln richtig aufgestellt wurden. Es war ebenfalls erkennbar, dass durch die modular berechneten finiten Differenzen im Gegensatz zu den global berechneten Gradienten ein zusätzlicher Fehler auftrat.

Zum Schluss wurde der Nutzen der neu implementierten Methoden in einem letzten Versuch analysiert. Dafür wurde eine Ersatz-Optimierung einmal mit den globalen finiten Differenzen und mit den modularen Gradienten durchgeführt. Die Ergebnisse und die Konvergenzverläufe dieser Optimierungen wurden miteinander verglichen. Aus dem Versuch war der Vorteil ersichtlich den die modular berechneten Gradienten mit sich bringen. Durch diese Methoden waren deutlich weniger

Berechnungen notwendig und dadurch wurde die benötigte Laufzeit der Optimierung enorm reduziert. Je mehr Paneele in der Optimierung verwendet werden, desto mehr war der Vorteil zu bemerken. Außerdem benötigte der Optimierer weniger Iterationen mit den modularen Gradienten als mit den globalen. Bei versteiften Paneelen ist aufgefallen, dass durch die modularen finiten Differenzen der Optimierer auf ein anderes Minimum konvergiert. In diesem Fall war das Ergebnis etwas besser als mit den anderen Methoden, wobei trotzdem alle Restriktionen erfüllt wurden. Darüber, ob es bei jeder Optimierung der Fall sein wird, ist noch keine Aussage zu treffen.

Insgesamt war es möglich durch die Implementierung der modularen Berechnung die Effizienz des Optimierungsprozess zu steigern.

6.2 Ausblick

Aus den Versuchen ergab sich, dass die automatischen Differenzen ineffektiver sind als die finiten Differenzen. Auch war auffällig, dass der Rechenaufwand der Jacobi-Methode mit Anstieg der abgeleiteten Parameter ansteigt. Da diese Methode nach dem Prinzip der Rückwärtsberechnung agiert, ist dieses Verhalten ungewöhnlich. Es sollte im Anschluss an diese Arbeit die Implementierung der automatischen Differentiation effizienter gestaltet werden. In der jetzigen Implementierung wird beim Berechnen der Gradienten über die Paneele im Strukturmodell iteriert und für jedes einzelne die Gradienten bestimmt. Eine möglicherweise effizientere Lösung wäre es das gesamte Parameter-Set aller Paneele in einem Durchlauf zu berechnen, indem diese als Matrix in die Jacobi-Funktion von *autograd* übergeben werden.

Eine andere Möglichkeit die Performance der automatischen Differentiation zu verbessern ist, andere Bibliotheken anzuwenden. Die Entwickler von *autograd* arbeiten an einer verbesserten Version von *autograd* namens *jax*. Durch *jax* ist es unter anderem möglich die Gradientenberechnung zu parallelisieren. Dadurch ist eine Verbesserung der Rechenzeit gegenüber der bisherigen sequentiellen Berechnung der Gradienten zu erwarten. In *lightworks* ist die „just in time“-Kompilierung für einige Funktionen, welche auch abgeleitet werden müssen, implementiert. Solche Funktionen haben eine verbesserte Laufzeit gegenüber dem normalen Python-Code. Da es mit *autograd* nicht möglich war diese Funktionen abzuleiten, musste die „just in time“-Kompilierung deaktiviert werden. Mit *jax* ist es aber möglich solchen Programmcode abzuleiten [18].

In der jetzigen neuen Implementierung werden die Gradienten für jede Iteration neu berechnet. Die Gradienten von linearen Funktionen sind allerdings konstant und verändern sich somit während des Optimierungsprozesses nicht. Ein Beispiel für eine solche Funktion ist der LP-Gültigkeitsbereich. Diese

muss während der Optimierung nur einmal berechnet werden und bleibt dann für jede Iteration konstant. Dadurch ist eine weitere Steigerung der Effizienz möglich.

In dieser Arbeit wurde ausschließlich mit einem Dummy-FE-Solver gerechnet. Es gilt noch zu untersuchen, ob die modularen Gradienten auch mit einem echten FE-Solver korrekt berechnet werden. Die in dem Versuch aus Kapitel 5 angegebenen Laufzeitverbesserungen durch die modularen Gradienten gegenüber den globalen Gradienten gelten nur für den eingesetzten Dummy-FE-Solver. Wie stark sich die Laufzeit mit einem echten FE-Solver verbessert ist noch zu klären.

Es wurden bisher ausschließlich Gradienten der Restriktionsfunktionen mit den neuen Methoden berechnet. Die Methoden können aber auch auf die Zielfunktion angewendet werden, sodass diese auch über automatische Differenzen berechnet wird und nicht wie bisher mit finiten Differenzen.

Zuletzt können die bisher verwendeten globalen Parameter erweitert werden, sodass es in der Optimierung größere Handlungsmöglichkeiten gibt, um die Flügelmasse evtl. noch weiter zu optimieren. Zum Beispiel kann in Zukunft auch die Geometrie der Paneele in *lightworks* verändert werden. Möglicherweise ist es dafür notwendig zusätzliche Restriktionen zu definieren, um die Geometrie der Paneele zu beschränken. Für diese Parameter müssen ebenfalls die Kettenregeln aus Kapitel 3.1 noch erweitert werden.

Literaturverzeichnis

- [1] DLR. (o.J.). *Common Language for Aircraft Design*. Abgerufen 07.01.2012, von <https://cpacs.de/#contents>.
- [2] Schürmann, H. (2007). *Konstruieren mit Faser-Kunststoff-Verbunden* (2.Auflage). Berlin: Springer.
- [3] Tsai, S. W. & Pagano, N. J. (1968). *Invariant Properties of Composite Materials*.
- [4] Dähne, S. & Hühne C. (2018). *Gradient Based Structural Optimization of a Stringer Stiffened Composite Wing Box with Variable Stringer Orientation*. Springer International Publishing.
- [5] Schumacher, A. (2013). *Optimierung mechanischer Strukturen: Grundlagen und industrielle Anwendungen* (2. Aufl.). Springer Vieweg.
- [6] Kuhn, H. W. & Tucker A. W. (1951). *Nonlinear Programming*.
- [7] Breuer, P. B. (2018). *Commercial Aircraft Composite Technology*. Springer International Publishing.
- [8] Vidyashankar, B. R. & Krishna Murty, A. V. (2001). *Analysis of Laminates with Ply Drops*. In *Composites Science and Technology* 61 (S. 749-758).
- [9] Niu, C & Niu, M. C. Y. (1992). *Composite Airframe Structures: Practical Design Information and Data*.
- [10] Bloomfield, M. W., Diaconu, C. G. & Weaver, P. M. (2009). *On Feasible regions of Lamination Parameters for Lay-up Optimization of Laminated Composites*. In *Proceedings of the Royal Society* (S. 1123-1143).
- [11] Wiedemann, J. (2007). *Leichtbau: Elemente und Konstruktion*. Springer-Verlag.
- [12] Martins, J. R. R. A., Poon, M. K. (2005). *On Structural Optimization Using Constraint Aggregation*. In: *6th World Congress in Structural and Multidisciplinary Optimization*.
- [13] Dennis, Jr., J. E. & Schnabel R. B. (1996). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. SIAM.
- [14] Rall, L. B. (1981). *Automatic Differentiation: Techniques and Applications*. Springer-Verlag.
- [15] HIPS-GitHub. (2019, Juni 29). *Autograd Tutorial*. Abgerufen 16.12.2020, von <https://github.com/HIPS/autograd/blob/master/docs/tutorial.md>.

-
- [16] Nemeth, M. P. (2011). *A Treatise on Equivalent-Plate Stiffnesses for Stiffened Laminated-Composite Plates and Plate-Like Lattices*. In: *NASA Technical Paper*. National Aeronautics and Space Administration.
- [17] IASB (2009). *Handbuch Strukturberechnung*.
- [18] Google-GitHub (2021, April 08). *Jax: Autograd and XLA*. Aufgerufen 05.05.2012 von <https://github.com/google/jax>.

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit eigenständig und ohne fremde Hilfe angefertigt habe. Textpassagen, die wörtlich oder dem Sinn nach auf Publikationen oder Vorträgen anderer Autoren beruhen, sind als solche kenntlich gemacht.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Braunschweig, 07.05.2021

Lennart Tönjes