



# Numerische Simulation von Spulenwicklungen

explizit Wickelkopfschwingungen an spulengewickelten  
Hydrogeneratoren

## Diplomarbeit

Zur Erlangung des akademischen Grades

Diplom-Ingenieur

an der

Technischen Universität Graz

Fakultät für Maschinenbau und Wirtschaftswissenschaften

Institut für Thermische Turbomaschinen und Maschinendynamik

Vorgelegt von: Manuel **LEIBETSEDER**

Betreut von: Dipl.-Ing. Dr.techn. Andreas **MARN**

Dipl.-Ing. Bernd **SCHLEGL**

Graz, am 05. November 2013

## Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, den 05. November 2013

## Danksagung

Am Ende eines Studiums ist es Zeit ein herzliches Danke an all jene Personen zu richten, die mich in dieser Zeit begleitet und unterstützt haben. Alle hier zu nennen wäre unmöglich, daher möchte ich hier nur jene Personen, die mir unmittelbar bei der Erstellung dieser Arbeit behilflich waren kurz erwähnen.

Besonderer Dank gilt meinem Betreuer Dipl. Ing. Dr. techn. Andreas Marn der mich seit Jänner 2013 nicht nur fachlich ausgezeichnet betreut hat, sondern mir auch bei persönlichen Fragen zur Berufswahl eine große Hilfe war. Ebenso danke ich Dipl. Ing. Dr. techn. Bernd Schlegl für die stets gute Zusammenarbeit und Betreuung der Arbeit bis Ende 2012.

Den Ansprechpersonen von der Firma ANDRITZ HYDRO GmbH Dipl. Ing. Dr. techn. Bernhard Bachner und Dipl. Ing. Dr. techn. Fritz Neumayer gilt ein großes Danke für die fachliche Unterstützung in Detailfragen, sowie die unkomplizierte Art der Kooperation.

Keine unmittelbare Hilfe aber dennoch sehr wichtig für die Arbeit war die finanzielle aber insbesondere persönliche Unterstützung meiner Eltern, die es mir erst ermöglicht haben zu studieren und mir stets Rückhalt und Ansprache bieten.

## Kurzfassung

Generatoren werden im Betrieb durch die periodischen elektromagnetischen Kräfte ständig zu Schwingungen angeregt. Im Bereich des Wickelkopfs ist die Abstützung dieser Kräfte konstruktionsbedingt schwierig auszuführen, weshalb es hier innerhalb des Generators zu den größten Amplituden kommt. Im Störfall können diese Kräfte auf ein Vielfaches, der im Betrieb wirkenden Kräfte, ansteigen. Ist nun die Anregungsfrequenz im Bereich einer Eigenfrequenz des Wickelkopfs, so kommt es zu Resonanz und es werden die Amplituden schnell zu groß. Eine mögliche Folge daraus kann sein, dass die Isolierung der Leiterstäbe beschädigt wird und es zum Ausfall der Maschine kommt. Um dies zu vermeiden ist es von Vorteil, das Schwingungsverhalten bereits in der Konstruktionsphase mittels FE-Simulation vorzuberechnen und nötigenfalls schon frühzeitig Gegenmaßnahmen zu setzen. Ein, dieser Arbeit vorausgehendes, Forschungsprojekt von ANDRITZ HYDRO GmbH in Zusammenarbeit mit dem Institut für Thermische Turbomaschinen und Maschinendynamik an der TU Graz untersuchte die Wickelkopfschwingungen für stabgewickelte Generatoren. Ergebnis dieser Arbeit ist ein Berechnungsprogramm für die Eigenfrequenzen von stabgewickelten Wickelköpfen. Im Rahmen dieser Arbeit wurde dieses Berechnungsprogramm für spulengewickelte Generatoren in großen Teilen neu geschrieben.

## Abstract

During operation generators are forced to oscillations due to electromagnetic forces. Especially in the end windings of the generator it is difficult to reinforce the conductors, so that the amplitudes can increase to dangerous values, if the frequency of the electromagnetic forces is similar to one of the eigen frequencies of the end windings. This could damage the insulation of the conductors and lead to severe damage. To avoid this, it is of advantage to simulate the oscillations and calculate the eigen frequencies with FE-methods during the construction period and to set counteractions if it's necessary before the generator will be built. A research project from ANDRITZ HYDRO GmbH in cooperation with the Institute for Thermal Turbomachinery and Machine Dynamics of the Technical University Graz was initialized, to examine the oscillations of bar-winding heads. The result of this project was a FE-calculation tool for the eigen frequencies of a bar-winding head. The main purpose of this thesis is to adapt the existing routine for coil-winded generators.

---

## Nomenklatur

$p$  ... .. Polpaarzahl, die halbe Anzahl der Pole

$n_{syn}$  ... .. Synchrondrehzahl in U/s

$f$  ... .. Netzfrequenz in Hz

$m$  ... .. Masse in kg

$k$  ... .. Federsteifigkeit in N/m

$\varepsilon$  ... .. Dehnung

$\ell$  ... .. Elementlänge in m

$E$  ... .. Elastizitätsmodul in MPa

$A$  ... .. Querschnittsfläche in m<sup>2</sup>

$\sigma$  ... .. Zugspannung in MPa

$u$  ... .. Knotenpunktverschiebung in m

$\dot{u}$  ... .. Knotengeschwindigkeit in m/s

$\ddot{u}$  ... .. Knotenbeschleunigung in m/s<sup>2</sup>

$[\mathbf{M}_e]$  ... .. Elementmassenmatrix

$[\mathbf{K}_e]$  ... .. Elementsteifigkeitsmatrix

$\underline{U}$  ... .. Verschiebungsvektor

$\dot{\underline{U}}$  ... .. Geschwindigkeitsvektor

$\ddot{\underline{U}}$  ... .. Beschleunigungsvektor

$\hat{\underline{U}}$  ... .. Verschiebungsamplitudenvektor des harm. Schwingungsansatzes

$\omega$  ... .. Eigenkreisfrequenz in 1/s

$i$  ... .. Imaginäre Einheit

$[M]$  ..... *Massenmatrix*

$[C]$  ..... *Dämpfungsmatrix*

$[K]$  ..... *Steifigkeitsmatrix*

$[K_F]$  ..... *Steifigkeitsmatrix extern angreifender Federkräfte*

$H_{ij}(\omega)$  ..... *Übertragungs- oder Vergrößerungsfunktion*

$F_j(\omega)$  ..... *Eingangsspektrum*

$X_i(\omega)$  ..... *Ausgangssignale der Beschleunigungssensoren*

---

# Inhaltsangabe

Eidesstattliche Erklärung.....	II
Danksagung .....	III
Kurzfassung .....	IV
Abstract .....	V
Nomenklatur .....	VI
Inhaltsangabe.....	VIII
<b>1 Einleitung .....</b>	<b>1</b>
1.1 Motivation .....	1
1.2 Aufgabenstellung.....	2
<b>2 Grundlagen .....</b>	<b>3</b>
2.1 Aufbau eines Generators .....	3
2.2 Ausgangssituation der Arbeit und Randbedingungen .....	14
2.3 Grundlagen einer rechnerischen Modalanalyse .....	16
2.4 Grundlagen einer experimentellen Modalanalyse (FFT) .....	23
<b>3 Ausführung und Arbeitsweise des Programms.....</b>	<b>27</b>
3.1 Software ANSYS Mechanical APDL.....	27
3.2 Anforderung an die vorgegebene Geometrie.....	30
3.3 Anforderungen an das Netz.....	33
3.4 Knotennummerierung.....	37
3.5 Geometriaufbereitung.....	41



3.6	Modellaufbau und Netzdetails .....	48
3.7	Vereinfachungen.....	60
3.8	Ergebnisse der rechnerischen Modalanalyse .....	62
4	Zusammenfassung.....	73
5	Abbildungsverzeichnis.....	74
6	Tabellenverzeichnis .....	77
7	Literaturverzeichnis .....	78
8	Anhang.....	80
8.1	Programmbeschreibung „Spulenmodell“ .....	80
8.2	Programmbeschreibung „Gewoelbekeile“ .....	118
8.3	Auswertung der Parameterstudie .....	131

# 1 Einleitung

## 1.1 Motivation

Die Gewinnung elektrischer Energie wird, mit Ausnahme der Photovoltaik- und Brennstoffzellentechnik, immer über den Zwischenschritt der mechanischen Energie erfolgen. Zur Erzeugung von mechanischer Energie aus der Primärenergie gibt es unzählige Möglichkeiten, die Umwandlung der mechanischen Energie in elektrische Energie erfolgt aber immer durch einen Generator. Die Weiterentwicklung dieser Maschinen besonders in der Steigerung der Zuverlässigkeit sowie der Optimierung des Wirkungsgrades und des Leistungsgewichtes sind heute Schwerpunkte in der Entwicklung. Der Ausfall eines Generators bedeutet neben den immensen Reparaturkosten auch noch das Fehlen der elektrischen Energie und einen Wegfall der Einnahmen für die gesamte Dauer der Reparatur. Die genaue Vorausberechnung der Beanspruchungen, insbesondere im Störfall ist hierbei mitunter von entscheidender Bedeutung. [1]

Die hohen Belastungen im Störfall können Bauteile des Generators zu Schwingungen anregen, was in weiterer Folge zu Lärmentwicklung und schlimmstenfalls zum Schadensfall und Ausfall der Maschine führen kann. Eines dieser hoch belasteten Bauteile ist der Wickelkopf, also jener Teil der Statorwicklung außerhalb des Blechpaketes. Die meisten Probleme traten bisher bei den sogenannten Turbogeneratoren auf, sodass bis dato das Hauptaugenmerk auf deren Berechnung gelegt wurde. Einige wissenschaftliche Arbeiten wurden bereits im Vorfeld an der TU Graz zu diesem Thema verfasst. (vgl. [2], [3], [4] )

Alle bisher untersuchten Generatoren (hauptsächlich Turbogeneratoren) hatten Stabwicklungen. Für Wickelköpfe des zweiten Wicklungstyps, der Spulenwicklung, war bisher keine Berechnung beziehungsweise Simulation in dieser Art möglich. Der Wunsch nach einem Berechnungsprogramm für ebendiesen Generatortyp wurde laut, weshalb dieses Projekt im Rahmen eines Forschungsprojektes [2] der Firma ANDRITZ HYDRO GmbH mit dem Institut für Thermische Turbomaschinen und Maschinendynamik gestartet wurde. Zur Erklärung der Begriffe „Stabwicklung“ und „Spulenwicklung“ muss auf die Kapitel am Anfang dieser Arbeit verwiesen werden. Die heute herkömmliche Simulation von gesamten CAD-Modellen mittels automatisch

vernetzenden FE-Programmen ist aufgrund der hohen Komplexität der Geometrie der Wickelköpfe nicht zielführend. Es war nötig ein einfaches Modell der komplexen Geometrie zu erzeugen, um mit vertretbarem Aufwand diese berechnen zu können.

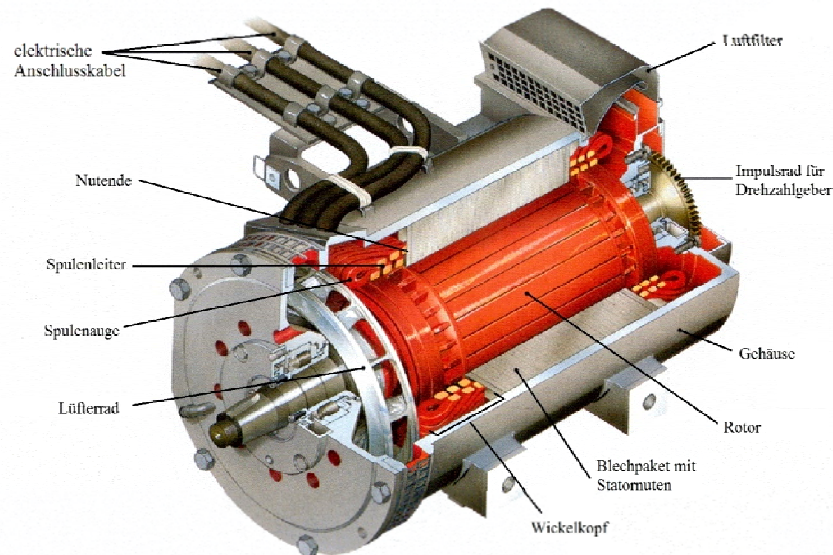
Aufgabe dieser Arbeit war ursprünglich die Weiterentwicklung eines bereits bestehenden Berechnungsprogramms, was sich aber im Laufe der Arbeit als eine nahezu Neuentwicklung des zugrundeliegenden Programmcodes entwickelt hat. Mit Hilfe des, im Zuge dieser Arbeit entstandenen, Berechnungsprogramms ist man nun in der Lage, Wickelköpfe spulengewickelter Generatoren zu berechnen und kann somit bereits in der Designphase mögliche spätere Probleme beim Betrieb der Maschine erkennen und diesen entgegensteuern. Dieses Wissen hilft somit nicht nur dem Generatorhersteller sondern insbesondere auch dem Betreiber durch die bereits Eingangs geforderte, gesteigerte Zuverlässigkeit der Maschine. (vgl. [3])

## 1.2 Aufgabenstellung

Die ursprüngliche Aufgabe dieser Arbeit bestand darin, ein Pendant, des im Zuge der Doktorarbeit von Herrn Dipl.-Ing. Dr. techn. Bernd Schlegl entstandenen, Berechnungsprogramms für stabgewickelte Generatorwickelköpfe, für spulengewickelte Wickelköpfe zu erstellen. Da Stabwicklungen und Spulenwicklungen einen grundlegend verschiedenen Aufbau haben, muss der Großteil des bestehenden Programms in seiner vorliegenden Form neu geschrieben werden. Eine Anpassung des Programms wäre wenig zweckmäßig, da die Spulenwicklungen eine gänzlich andere Handhabung der vorgegebenen Geometrie und Art der Vernetzung erfordern. Das Berechnungsprogramm bekommt eine Parasolid-Datei als Input, zerlegt diese selbständig in zwei Spulenteile und führt die Geometrie auf die Form einer Stabgeometrie zurück, danach werden ebenfalls automatisch nicht benötigte Geometrieteile erkannt und gelöscht bevor der, speziell an Spulenwicklungen angepasste, Algorithmus die Geometrie vernetzt. Durch die Anwendung auf einen speziellen Generator und den Vergleich der Simulationsergebnisse mit denen der experimentellen Schwingungsanalyse dieses Generators wird es möglich den Einfluss der noch teilweise unbekanntem Materialparameter zu studieren und gibt in weiterer Folge die Möglichkeit einer Vorausberechnung des Schwingungsverhaltens noch vor dem Bau der Maschine.

## 2 Grundlagen

### 2.1 Aufbau eines Generators

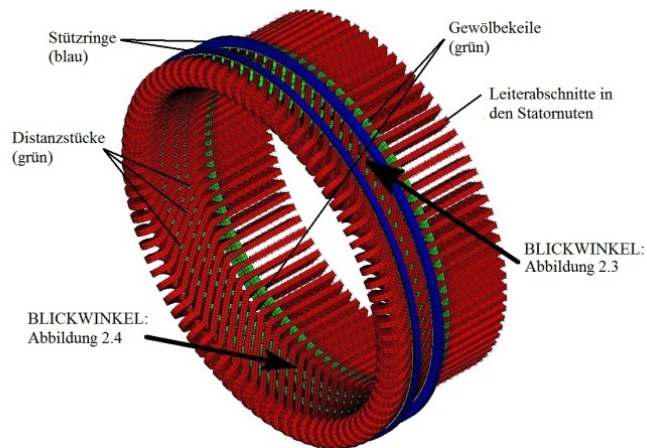


**Abbildung 2.1: Aufbau einer spulengewickelten Drehstrommaschine [5]**

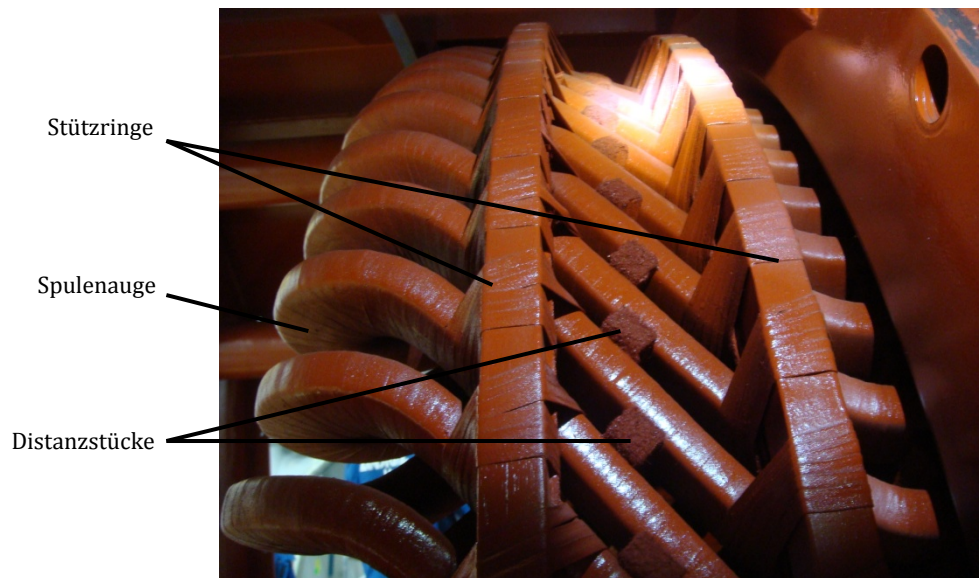
In Abbildung 2.1 ist der Teilschnitt einer spulengewickelten Drehstrommaschine dargestellt. Diese Arbeit befasst sich ausschließlich mit dem Wickelkopf, also jenen Teilen der Spulenleiter, welche außerhalb des Blechpaketes liegen.

#### 2.1.1 Der Wickelkopfaufbau spulengewickelter Generatoren

Die Abbildung 2.2 zeigt schematisch den Aufbau des Wickelkopfs und die Blickwinkel unter welchen die folgenden Abbildungen aufgenommen wurden. Das Blechpaket ist nicht dargestellt, dafür aber ein Stück der, in den Nuten des Blechpaketes verlaufenden, geraden Leiterabschnitte.

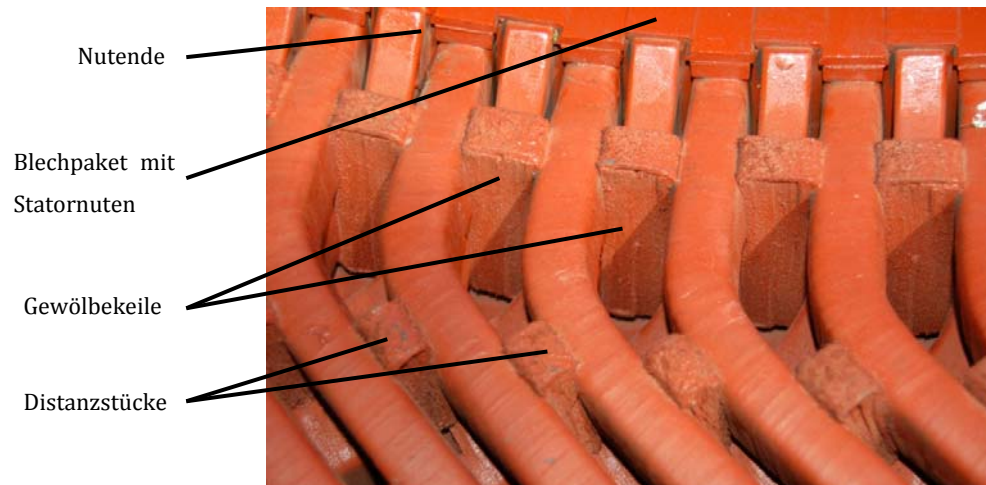


**Abbildung 2.2: Wickelkopf schematisch mit Blickwinkel der folgenden Abb.**



**Abbildung 2.3: Wickelkopfaufbau eines spulengewickelten Generators [6]**

Abbildung 2.3 ist eine Ansicht von außen auf den Wickelkopf ohne Gehäuse. Rechts des Bildes ist das Blechpaket mit den Statornuten, in welchen die Leiter liegen. Die abgebildeten, nicht eingebetteten Leiter im Bereich des Wickelkopfs sind mit sogenannten Distanzstücken, Stützringen und Gewölbekeilen abgestützt. Die Abbildung 2.4 zeigt das Nutende, wo die Leiter das Blechpaket verlassen die davor liegenden Gewölbekeile und die erste, in diesem Falle, 3 Distanzstückreihen des Wickelkopfs.



**Abbildung 2.4: Austritt der Leiter aus dem Blechkpaket am Nutende [6]**

### 2.1.2 Unterscheidung zwischen Turbo- und Hydrogenerator

Während Turbogeneratoren meist von Gas- und Dampfturbinen angetrieben werden, welche bei hohen Drehzahlen (max. 3000 oder 3600 U/min) laufen, werden die sogenannten Hydrogeneratoren oder Schenkelpolmaschinen von Wasserturbinen mit wesentlich kleineren Drehzahlen (im Extremfall unter 70 U/min bei Kaplan turbinen) angetrieben. Laut „Gieseke, Mosonyi: Wasserkraftanlagen“ [1] gelten Generatoren mit Polpaarzahl 1 oder 2 als Turbogeneratoren, alle Generatoren mit höherer Polpaarzahl als Hydrogeneratoren. Der Zusammenhang zwischen Polpaarzahl, Netzfrequenz und Synchrondrehzahl ist wie folgt:

$$p = \frac{f}{n_{syn}}$$

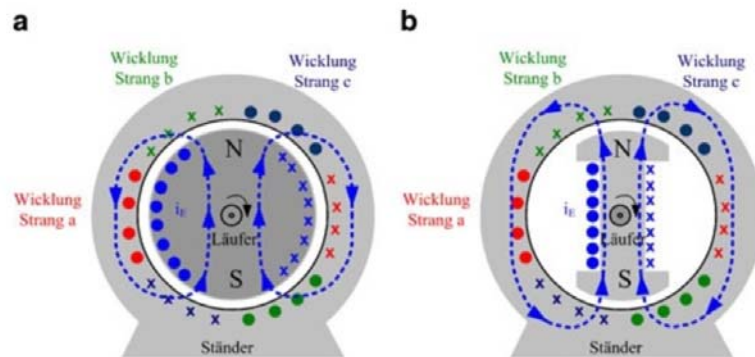
$p$  ... .. Polpaarzahl, also die halbe Anzahl der Pole

$n_{syn}$  ... Synchrondrehzahl der Maschine in U/s

$f$  ... .. Netzfrequenz in Hz

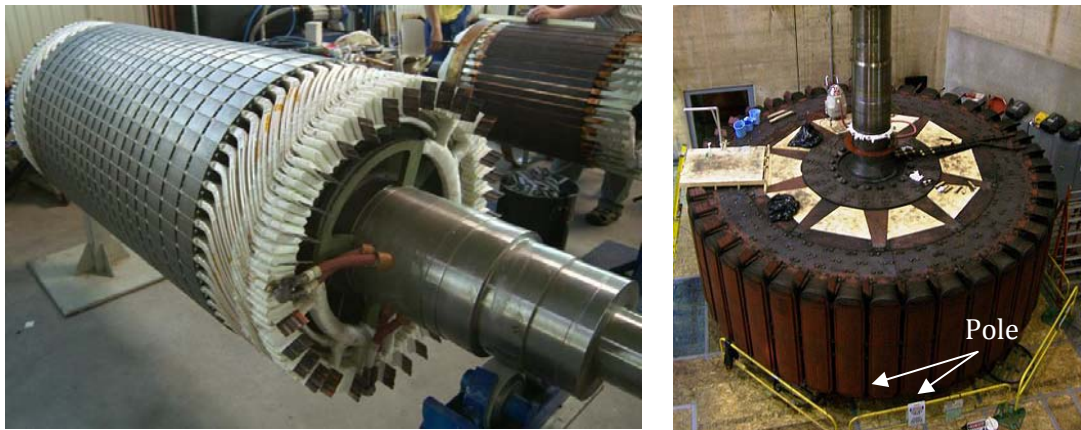
Unter der Annahme einer unveränderlichen Netzfrequenz von 50 oder 60 Hz, bestimmt somit die Polpaarzahl allein die Synchrondrehzahl des Generators. „Teigelkötter: Energieeffiziente elektrische Antriebe“ [7], siehe Abbildung 2.5, bezeichnet Maschinen mit Vollpolläufer als Turbomaschinen, die andere Bauweise mit ausgeprägten Polen am

Läufer als Schenkelpolmaschinen, welche hauptsächlich als Generatoren in Wasserkraftanlagen, also als Hydrogenerator, eingesetzt werden.



**Abbildung 2.5: Ausführungen von Synchronmaschinen [7]:  
(a) Vollpolläufer oder Turboläufer, (b) Schenkelpolläufer**

Die höheren Polpaarzahlen bei Hydrogeneratoren bedingen einen größeren Durchmesser und erfordern mehr Statornuten. Die im Gegenzug kürzere Bauform, welche bei Einbau mit vertikaler Welle von Vorteil ist, hat aber den Nachteil, dass weniger Länge zum Aufbau der Spannung im Leiter zur Verfügung steht und deshalb ein und derselbe Leiter mehrmals durch eine Statornut geführt werden muss. In diesem Fall muss auf Spulenwicklungen zurückgegriffen werden. Näheres dazu im folgenden Kapitel. Die beiden Fotos in Abbildung 2.6 zeigen den Aufbau von Vollpol- oder Turboläufer und Schenkelpolläufer. Hier zeigt sich die wesentlich kürzere Bauweise bei verhältnismäßig großem Durchmesser der Schenkelpol- oder Hydrogeneratoren im Vergleich zum langen, schlanken Läufer des Turbogenerators.

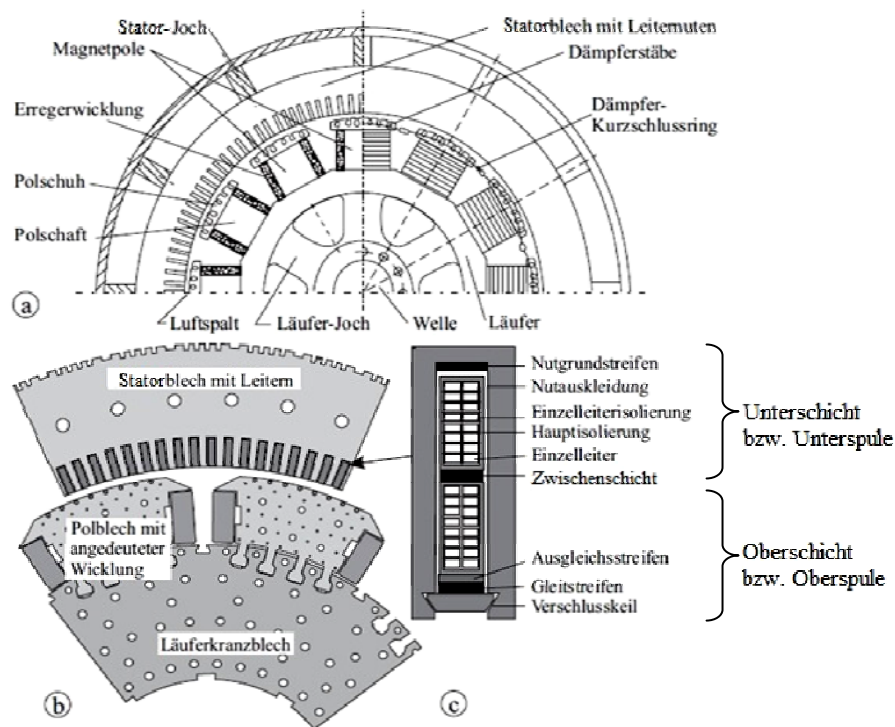


**Abbildung 2.6: Läuferausführungen:**

**links: Vollpolläufer oder Turboläufer [8], rechts: Schenkelpolläufer eines Hydrogenerators beim Hoover-Damm [9]**

Abbildung 2.7 zeigt den schematischen Querschnitt durch einen Hydrogenerator mit Polpaarzahl  $p=6$ . Im Schnitt (a) sind die insgesamt 12 Pole angedeutet und das Statorblechpaket mit den Leiternuten dargestellt. Schnitt (c) zeigt den Schnitt durch eine Statornut mit zwei Leitern, „Gieseke, Mosonyi: Wasserkraftanlagen“ [1] bezeichnet den achsnäheren Leiter als Oberschicht, den darunterliegenden als Unterschicht. In dieser Arbeit werden ebenfalls die Bezeichnungen Ober- und Unterstab bei Stabwicklungen beziehungsweise Ober- und Unterspule verwendet. Die Begriffe Ober- und Unterspule sind lediglich zur Unterscheidung, des in der Nut oben bzw. unten liegenden Teils, des durchgehenden Spulenloops eingeführt worden. Wenn in den folgenden Kapiteln das Auftrennen oder Teilen der Spule angesprochen wird, so ist dies rein in der Simulation zur Vereinfachung der Berechnung so ausgeführt, der Spulenloop im realen Generator ist und bleibt in einem Stück erhalten.

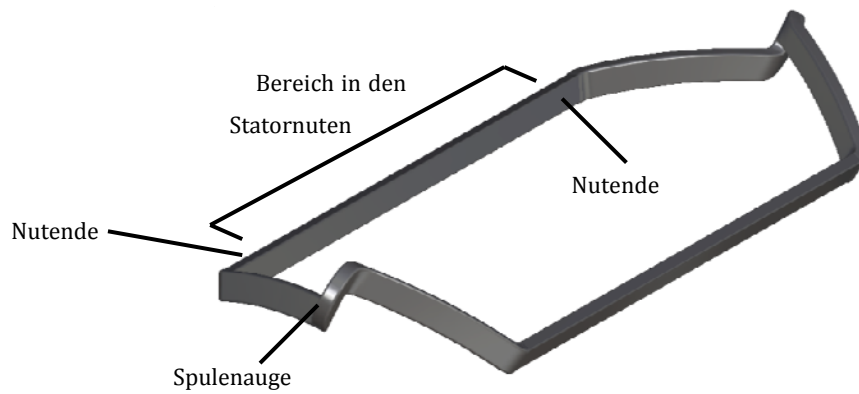




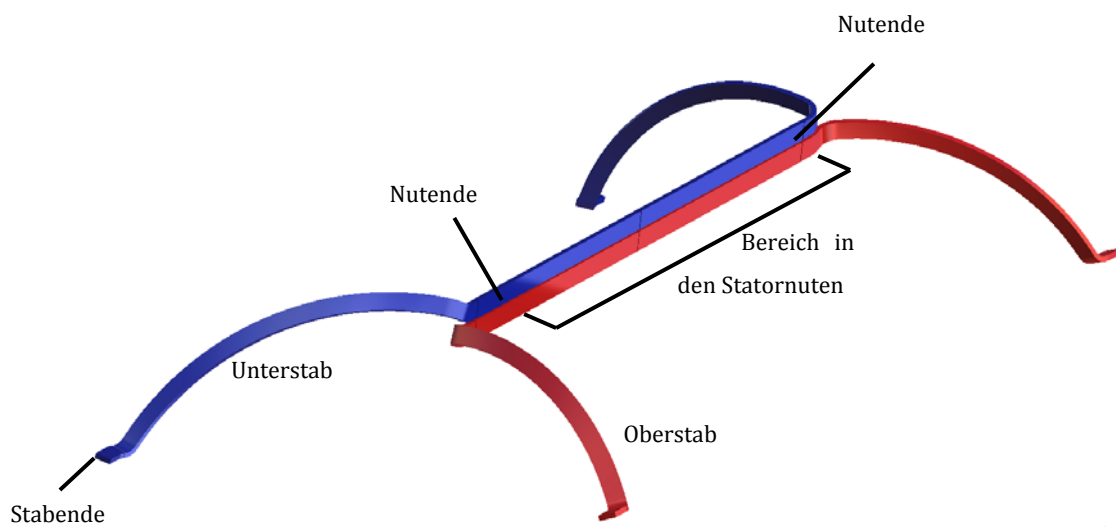
**Abbildung 2.7: Schematischer Querschnitt durch einen Hydrogenerator [1]:**  
**a) durch eine 12-polige Synchronmaschine mit Einzelpolen (Bezeichnung charakteristischer Bauteile); b) Detail der Blechpakete; c) Querschnitt einer Zweischichtleiterwicklung mit Ober- und Unterschicht**

### 2.1.3 Unterscheidung zwischen Stab- und Spulenwicklung

Der grundlegende äußere, und für die Simulation wesentliche Unterschied zwischen Stab- und Spulenwicklungen ist die Verbindungsstelle zwischen Ober- und Unterschicht. Dort wo bei Stabwicklungen der Ober- und Unterstab mit separaten Laschen zusammengelötet wird, geht bei Spulenwicklungen die Spule in einem Stück durch und bildet eine geschlossene Schleife. Diese Stelle wo die Leiter in engem Radius um  $180^\circ$  umgebogen werden, nennt man das Spulenaugen.

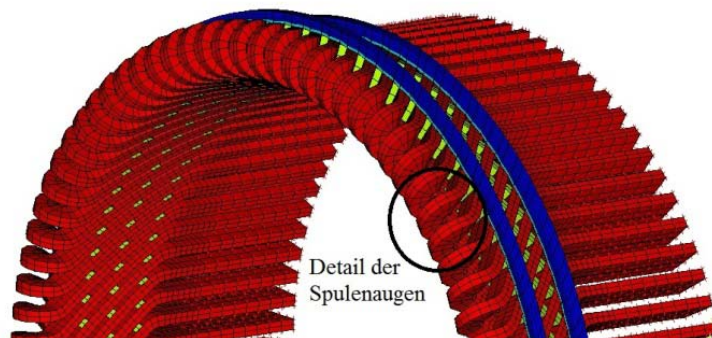


**Abbildung 2.8: Geschlossene Spulenschleife**



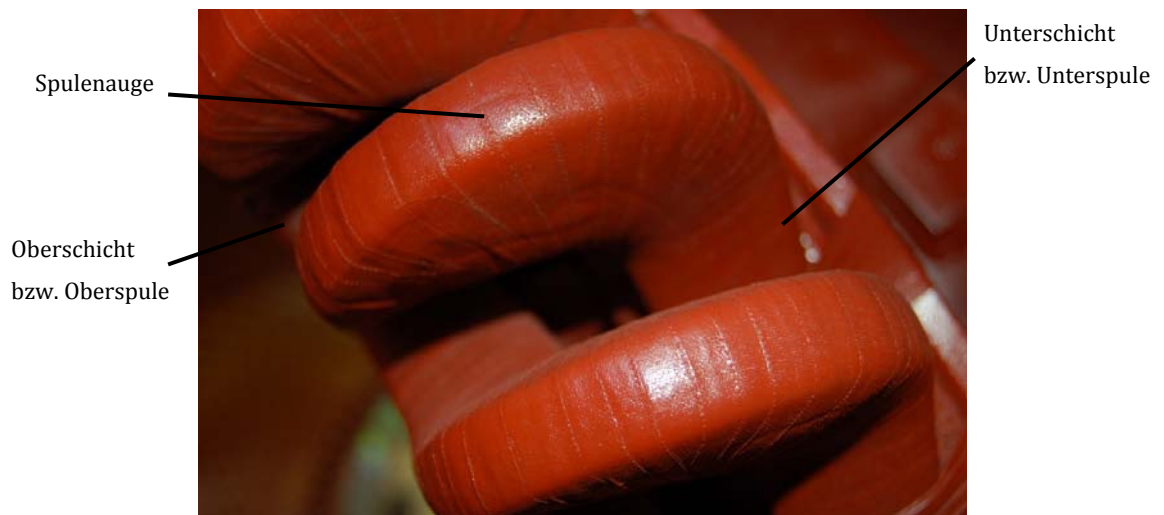
**Abbildung 2.9: Die zwei separaten Stäbe einer Stabwicklung**

Die Abbildung 2.8 und Abbildung 2.9 zeigen das äußere Erscheinungsbild von Spulen- und Stabgeometrie ohne Blechpaket. Alle Teile außerhalb des „Bereiches in den Statornuten“ bilden den Wickelkopf.



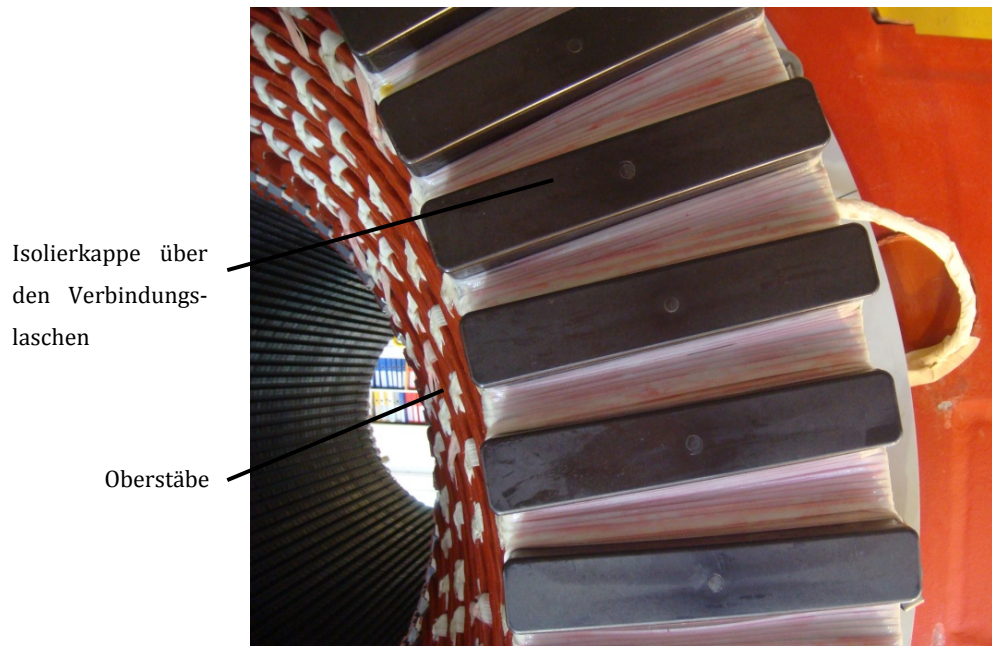
**Abbildung 2.10: Wickelkopf einer Spulenwicklung schematisch**

Die Abbildung 2.10 bildet den Wickelkopf zur Hälfte ab, die Abbildung 2.11 zeigt den oben eingekreisten Bereich als Detailfoto. Darauf sind zwei der Spulenaugen zu erkennen. Die Leiter sowie die Isolierung gehen an dieser Stelle durch. Der Übergang bzw. die Grenze zwischen Ober- und Unterspule ist nicht durch ein Bauteil definiert sondern wird erst in der Modellierung an der günstigsten Stelle festgelegt.



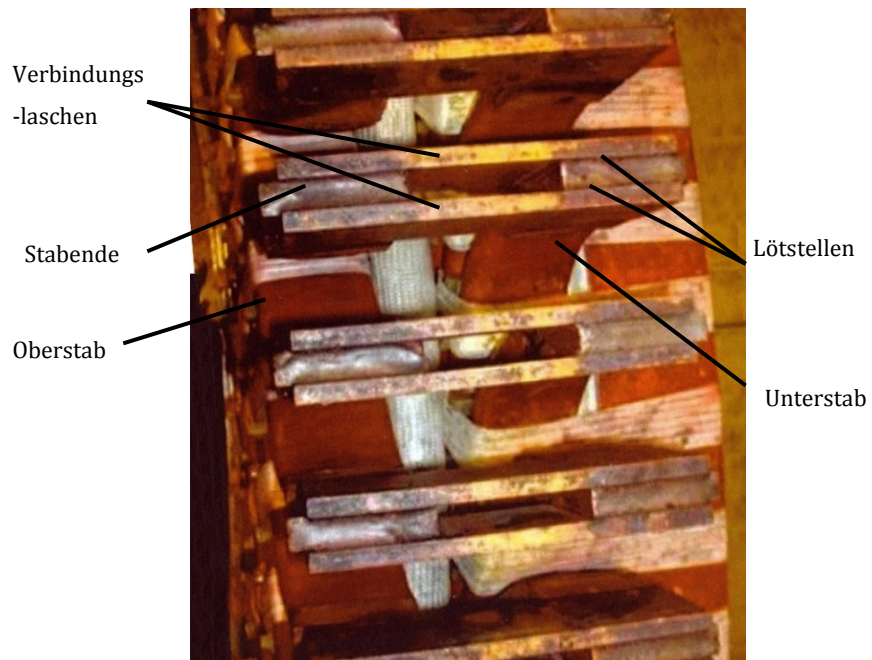
**Abbildung 2.11: Detailfoto der Spulenaugen [6]**

Bei den, in den vorangegangenen Projekten modellierten, Stabwicklungen sieht diese Stelle am Wickelkopf wie in Abbildung 2.12 aus. Der stark gekrümmte Leiterabschnitt fällt weg, dafür sind an dessen Stelle Verbindungslaschen angelötet.



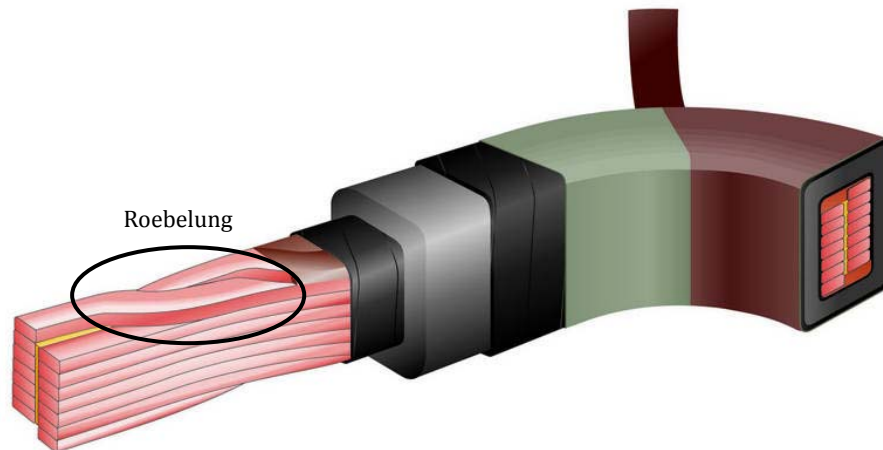
**Abbildung 2.12: Verbindungs-laschen an Stabwicklungen isoliert [6]**

Die selbe Ansicht nur ohne Isolierkappen zeigt die Abbildung 2.13. Die beiden Stäbe stehen am Ende des Wickelkopfs nach vorne weg und werden an deren Ende beidseitig mit Laschen verlötet.

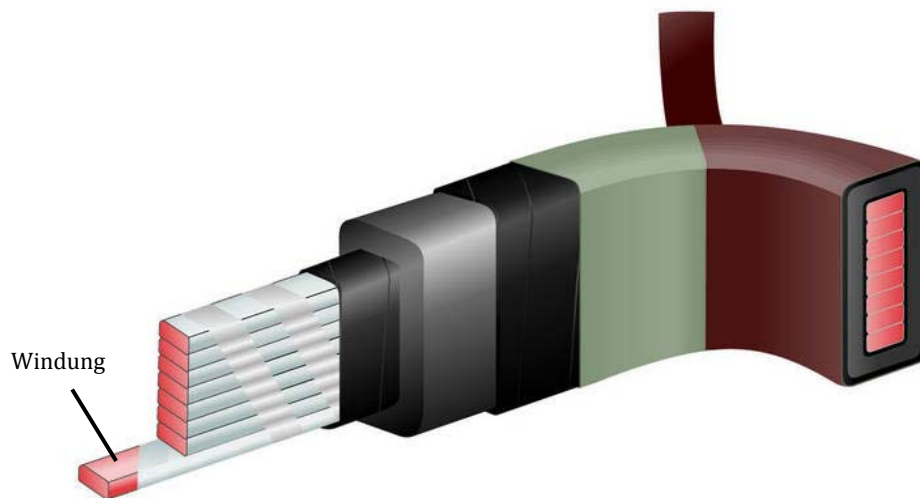


**Abbildung 2.13: Verbindungs-laschen an Stabwicklungen nicht isoliert [6]**

Nicht nur äußerlich unterscheiden sich Stab- und Spulenwicklungen, sondern auch im Aufbau der Leiter an sich. Während die Stäbe bei Stabwicklungen aus sogenannten Roebelstäben bestehen, bei denen einzelne Kupferleiter voneinander isoliert und verdreht sind, sind bei Spulenwicklungen die einzelnen Kupferleiter nicht verdreht (oder „geroebelt“) sondern laufen, ebenfalls voneinander isoliert, geordnet parallel nebeneinander her. Vergleiche dazu Abbildung 2.14 mit Abbildung 2.15.



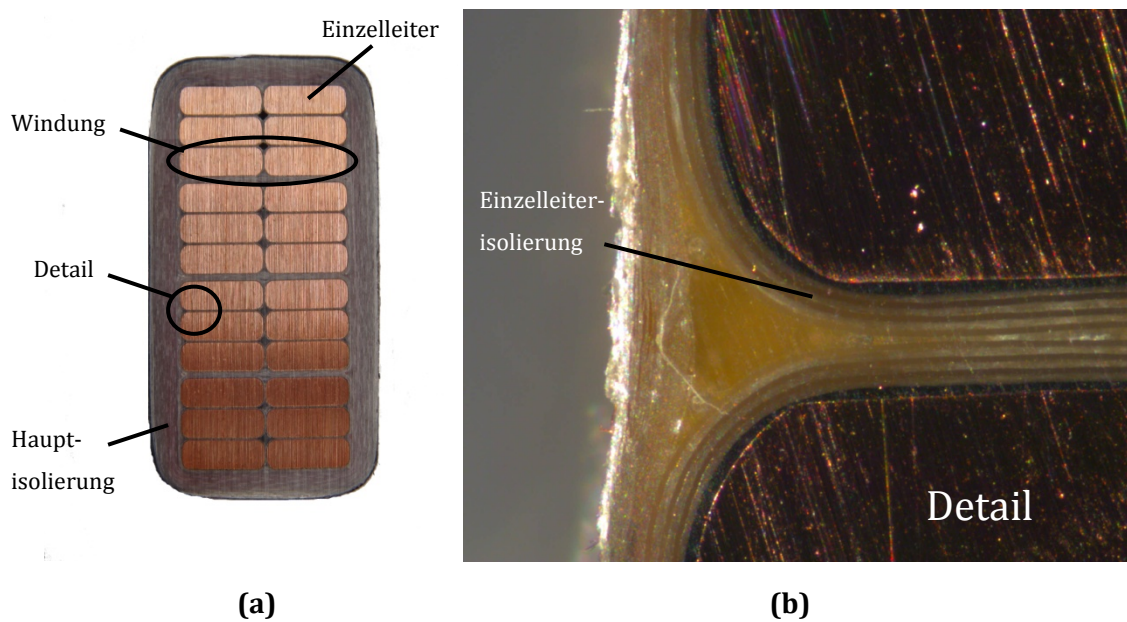
**Abbildung 2.14: Aufbau eines Roebelstabes [10]**



**Abbildung 2.15: Spulenaufbau [10]**

Anders als bei Wicklungsstäben, welche einen Anfang und ein Ende haben, sind Spulen dem äußeren Erscheinungsbild nach geschlossene Schleifen oder Loops. Die Abbildung 2.16 (a) zeigt den Aufbau eines Spulenquerschnitts. Zwei Einzelleiter nebeneinander sind zu einer Windung zusammengefasst und isoliert. In diesem Fall läuft diese

Windung 12 Mal in der Spulenschleife im Kreis. Diese 12 Windungen sind außen nochmals mit einer Hauptisolierung isoliert. Würde man im Betrieb die Spannung der einzelnen Windungen messen können, so würde man bei Spulenwicklungen im Querschnitt in den verschiedenen Windungen unterschiedliche Spannungen messen. Bei Stabwicklungen im Vergleich wäre über den Querschnitt die Spannung gleich. Die Isolierung der einzelnen Windungen voneinander muss daher, wie in Abbildung 2.16 (b) ersichtlich, entsprechend stärker sein als jene bei Stabwicklungen, die dort nur zur Vermeidung von unerwünschten Wirbelströmen dient.



**Abbildung 2.16: Leiteranordnung in der Spule [6]:  
a) gesamter Querschnitt , b) Detail der Einzelleiterisolierung**

## 2.2 Ausgangssituation der Arbeit und Randbedingungen

Wie bereits in der Aufgabenstellung beschrieben, ist die Ausgangssituation für diese Arbeit ein bereits bestehendes Finite-Elemente-Berechnungsprogramm für stabgewickelte Generatoren. Abbildung 2.17 zeigt den Aufbau beziehungsweise den Ablauf des gesamten Berechnungsprogramms. Die rot gekennzeichneten Kästchen sind dabei gänzlich neu programmiert, „Gelb markiert“ bedeutet eine grundlegende Überarbeitung des Programmteils und grün keine oder eine nur minimale Überarbeitung.

Das Einlesen der technischen Parameter und der Materialeigenschaften bleibt sinnhafterweise gleich, da es dem Benutzer erlaubt, die selbe Eingabemaske wie zur Simulation von Stabwicklungen zu verwenden. Das „Programm Spulenmodell“ ist im Vergleich zum vorherigen Stabmodell für Stabwicklungen um einiges komplexer und besteht nun aus rund doppelt so vielen Zeilen, sodass es nun etwa ein Drittel des gesamten Programmcodes ausmacht. Die beiden Unterprogramme „Nutende“ und „Federanbindungsknoten sperren“, welche gelb markiert sind, wurden überarbeitet. Das „Rotationssymmetrische Mustern“, welches nur aus 6 Zeilen besteht, bleibt gleich. Neu hinzugekommen ist der Programmteil „Verbinden der aufgetrennten Spulenteile an den Stoßstellen“. Dieser ersetzt die Modellierung der Kupferverbindungsflaschen bei Stabwicklungen. Die beiden Anbauteile, Distanzpakete und Stützringe, können direkt übernommen werden. Für die Modellierung der, nur bei Spulenwicklungen vorkommenden, Gewölbekeile ist aber ein gänzlich neues Unterprogramm geschrieben worden. Die beiden letzten Programmteile „Nachbearbeitung“ und „Modalanalyse“ machen in Summe nur etwa 10 Zeilen aus und können wiederverwendet werden.

Das vorherige Berechnungsprogramm, sowie auch das neue Programm für die Spulenwicklungen mussten auf Wunsch der Firma ANDRITZ HYDRO GmbH in ANSYS Mechanical APDL programmiert werden, da dieses Programm bereits in Verwendung ist und es im Gegensatz zu neueren Versionen mehr Eingriffsmöglichkeiten auf die Arbeitsweise des Programmes bietet.

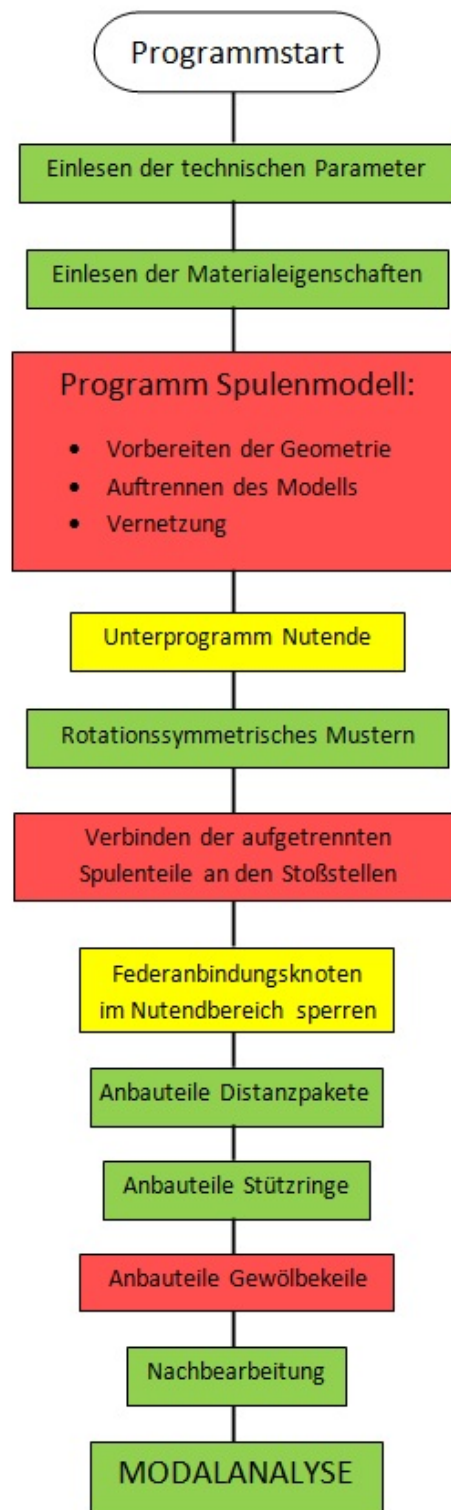


Abbildung 2.17: Aufbau der Programmstruktur



## 2.3 Grundlagen einer rechnerischen Modalanalyse

### 2.3.1 Bewegungsgleichung in Matrizenform

Die Bewegungsgleichung eines Einmassenschwingers kann, wie aus der Maschinendynamik bekannt, folgendermaßen angeschrieben werden.

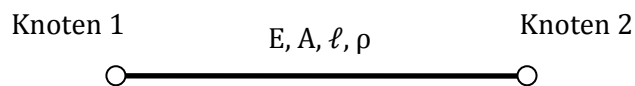
$$m \cdot \ddot{u} = \sum F_i$$

Setzt man für die äußeren Kräfte, eine der Auslenkung  $u$  proportionale Federkraft  $-k \cdot u$  ein, so erhält man die Bewegungsgleichung eines einfachen schwingungsfähigen Systems. Andere Kräfte werden vorerst beabsichtigt noch außer Acht gelassen. Auf sie wird aber am Ende des Kapitels noch eingegangen. Es folgt daraus die Bewegungsgleichung nach d'Alembert:

$$m \cdot \ddot{u} + k \cdot u = 0$$

$$\text{Massenkraft} + \text{Federkraft} = 0$$

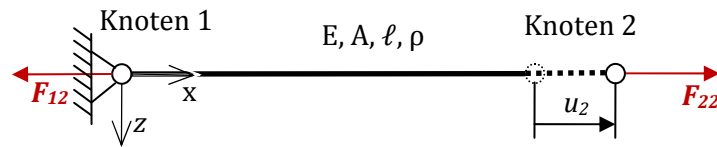
Die beiden Terme beschreiben die Massenkraft und die Federkraft des Systems. Diese Formel gilt nur für eine Punktmasse, also wenn alle Kräfte im Schwerpunkt wirken. Um aber auch komplexere Geometrien als einfachere, schwingungsfähige Systeme betrachten zu können, muss man das System der Punktmasse erweitern und durch ein zumindest eindimensional ausgedehntes Element ersetzen. Zur Erklärung soll das Stabelement (Abbildung 2.18) dienen, welches man sich als einen, zwischen zwei Knoten aufgespannten, Zug- Druckstab vorstellen kann.



**Abbildung 2.18: Das Stabelement**

Will man diesen Stab beschreiben, so muss man zumindest die Zustände in allen beiden Knoten berücksichtigen. Die Knoten sind aber durch das elastische Element gekoppelt. Die Kraftwirkungen in den Knoten sind somit abhängig voneinander.

*Anmerkung zu den Kräften:* Der erste Index beschreibt den Kraftangriffspunkt (Knotennummer) der zweite Index, zufolge welcher Verschiebung die Kraft zustande kommt.



**Abbildung 2.19: Verschiebung und Kraftwirkung**

Stellt man sich nun den Knoten 1 eingespannt vor und prägt dem Knoten 2 eine Verschiebung in positiver x-Richtung auf, so wirkt der Stab als Feder und es wirkt eine Kraft zufolge der Verschiebung (Abbildung 2.19). Den Zusammenhang zwischen Kraft und Verschiebung, also der Steifigkeit des Zugstabes, errechnet man aus dem Hooke'schen Gesetz folgendermaßen.

$$\varepsilon = \frac{\partial u}{\partial x} \cong \frac{\Delta u}{\Delta x} = \frac{u_2 - u_1}{\ell}$$

Da laut Definition  $u_1 = 0$  ist, kann folgender Zusammenhang angeschrieben werden

$$\sigma = E \cdot \varepsilon = E \cdot \left(\frac{u_2}{\ell}\right)$$

$$\sigma = \frac{F_{22}}{A}$$

$$F_{22} = \left(\frac{E \cdot A}{\ell}\right) \cdot u_2$$

Da eine Kraft in Knoten 2 eine Reaktionskraft in Knoten1 bedingt, welche betragsmäßig gleich ist aber entgegengesetzt wirkt, folgt nun.

$$F_{12} = -F_{22} = -\left(\frac{E \cdot A}{\ell}\right) \cdot u_2$$

Die Knotenkräfte lassen sich auch zufolge einer Verschiebung des Knoten 1 bei eingespannt gedachtem Knoten 2 ermitteln.

$$F_{21} = -\left(\frac{E \cdot A}{\ell}\right) \cdot u_1$$

$$F_{11} = \left(\frac{E \cdot A}{\ell}\right) \cdot u_1$$

Die Gleichungen können nun zusammengefasst (superponiert) werden zu

$$F_1 = F_{11} + F_{12} = \left(\frac{E \cdot A}{\ell}\right) \cdot u_1 + -\left(\frac{E \cdot A}{\ell}\right) \cdot u_2$$

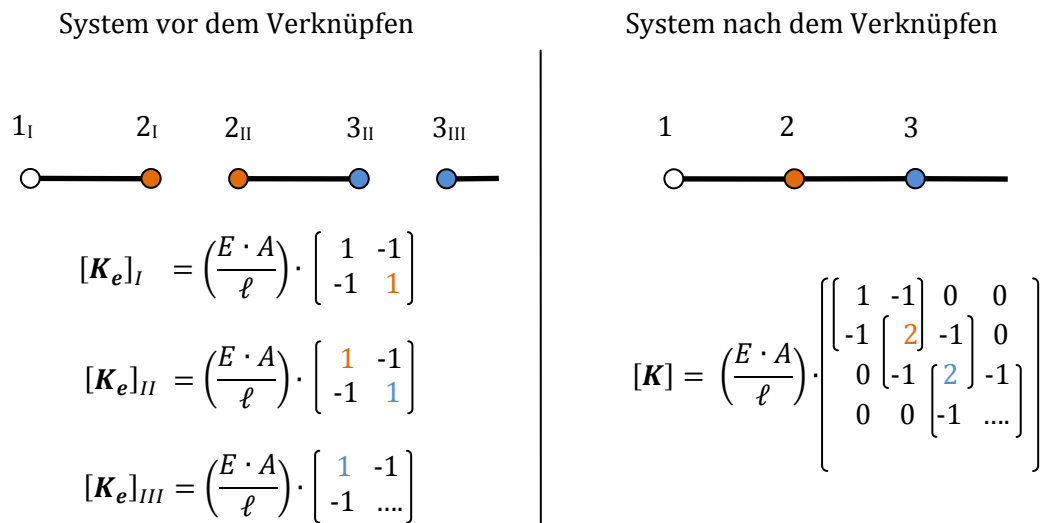
$$F_2 = F_{21} + F_{22} = -\left(\frac{E \cdot A}{\ell}\right) \cdot u_1 + \left(\frac{E \cdot A}{\ell}\right) \cdot u_2$$

beziehungsweise noch eleganter in Matrizenform angeschrieben werden.

$$\begin{aligned} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} &= \begin{bmatrix} F_{11} + F_{12} \\ F_{21} + F_{22} \end{bmatrix} = \underbrace{\left(\frac{E \cdot A}{\ell}\right) \cdot \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}}_{[\mathbf{K}_e]} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\ \underline{F}_e &= \underline{F}_{e1} + \underline{F}_{e2} = [\mathbf{K}_e] \cdot \underline{U}_e \end{aligned}$$

Die erhaltene Steifigkeitsmatrix  $[\mathbf{K}_e]$  beschreibt nun die Steifigkeit eines Stabelements. Man kann aber noch die Struktur der Formel für die Federkraft  $F = k \cdot u$  erkennen. Wobei lediglich die Einzelkraft  $F$  durch den Kraftvektor  $\underline{F}_e$ , die Federsteifigkeit  $k$  durch die Steifigkeitsmatrix  $[\mathbf{K}_e]$  und die Verschiebung  $u$  durch den Verschiebungsvektor  $\underline{U}_e$  ersetzt wurden.

Bei einer Anordnung von zwei oder mehreren Stabelementen hintereinander, wirken in den Knoten zwischen den Elementen jeweils Kräfte beider Elemente. Durch das Verknüpfen der Matrizen werden die Kraftkomponenten gleichen Angriffspunktes (Knoten) zufolge gleicher Verschiebungen *addiert*. Dies erkennt man, in der einfachen Steifigkeitsmatrix aus Abbildung 2.20, in den Diagonalelementen.



**Abbildung 2.20: Verknüpfung der Elemente und Matrizen**

Da der Eintrag in der zweiten Zeile und zweiten Spalte der ersten Elementsteifigkeitsmatrix  $[K_e]_I$  eine Kraft im Knoten 2 zufolge einer Verschiebung im Knoten 1 beschreibt, genauso wie der erste Eintrag in der zweiten Elementsteifigkeitsmatrix  $[K_e]_{II}$ , so können diese beiden Einträge in der gesamten Steifigkeitsmatrix  $[K]$  schließlich zusammengefasst werden. Beschreibt die Matrix mehrere Zustände pro Knoten, bestehen die zu verknüpfenden Einträge wiederum aus Matrizen.

Analog kann bei den Massenkräften vorgegangen werden. Hier erhält man die Massenmatrix  $[M]$ , welche mit dem Beschleunigungsvektor  $\underline{\ddot{U}}$  multipliziert, die Massenkräfte beschreibt.

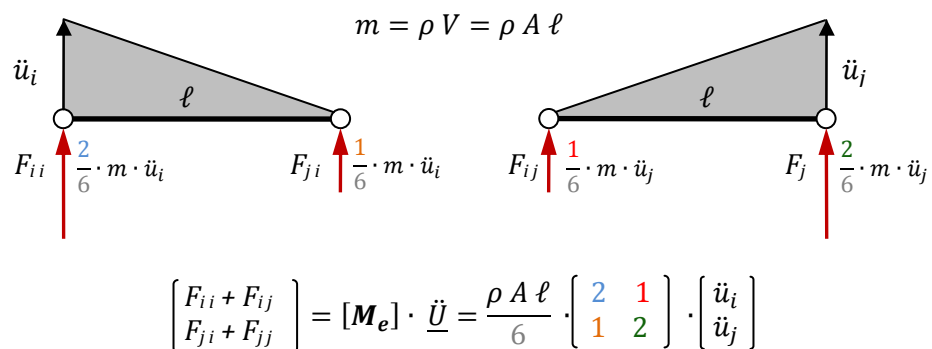
Es folgt aus der einfachen Bewegungsgleichung einer Punktmasse

$$m \cdot \ddot{u} + k \cdot u = 0$$

eine Bewegungsgleichung in Matrizenform für die Finite Elemente Methode

$$[M] \cdot \underline{\ddot{U}} + [K] \cdot \underline{U} = 0$$

Einschub: Eine nicht ganz korrekte aber anschauliche Erklärung der Massenmatrix liefert die Betrachtung eines Trägers auf zwei Stützen. Da es praktisch keinen Balken nur mit Massenkräften aber ohne Steifigkeit gibt, umgeht man diesen Umstand indem man sich die Bewegung um  $90^\circ$  verdreht vorstellt. Da sich am Betrag und Angriffsort der Beschleunigung nichts ändert, sind auch die angreifenden Massenkräfte dem Betrage nach gleich, nur wirken diese in eine Richtung in der die Steifigkeitskräfte wenig Einfluss haben. Wird dieser Träger nun, wie die Formfunktion  $h_i$  andeutet, im Knoten  $i$  mit  $\ddot{u}_i$  beschleunigt, so wird die, über den Balken gleichmäßig verteilte Masse entsprechend der Formfunktion beschleunigt (linkes Bild in Abbildung 2.21). Die Formfunktion kann also als dreiecksförmige Belastung auf den Balken interpretiert werden, welche dem Betrag nach  $\frac{1}{2} \cdot m \cdot \ddot{u}_i$  sein muss. In Abbildung 2.21 ist diese Last als graue Fläche dargestellt.



**Abbildung 2.21: Skizze zum Gedankenmodell für die Massenmatrix**

Die Auflagerreaktionen eines, mit einer Dreieckslast belasteten, Trägers sind bekanntlich mit  $\frac{2}{3}$  auf das Lager unter der maximalen Last und mit  $\frac{1}{3}$  auf das Lager ohne Last aufzuteilen. Da die erste Spalte der Matrix mit der Beschleunigung des ersten Knoten multipliziert wird – also den eben betrachteten Fall darstellt – und die Zeilen die Knoten- bzw. Lagerkräfte abbildet, erklärt sich weshalb bei einer Beschleunigung des Knotens  $i$ ,  $\frac{2}{6}$  der Massenkraft auf den Knoten  $i$  und  $\frac{1}{6}$  auf den Knoten  $j$  wirkt. Bei Beschleunigung des Knoten  $j$  teilen sich die Kräfte, wie in der rechten Skizze gezeichnet, umgekehrt auf. Dieses Gedankenmodell eignet sich nicht zum Erklären der Massenmatrix, sondern soll lediglich als anschaulicheres Modell zum besseren Verständnis beitragen.

Optional bietet sich noch an, ein geschwindigkeitsproportionales Dämpfungsverhalten des Elementwerkstoffes in einer Dämpfungsmatrix  $[C]$  zu beschreiben. Multipliziert mit dem Geschwindigkeitsvektor  $\underline{\dot{U}}$  ergibt dies den Dämpfungskraftvektor welcher ggf. ebenfalls zur linken Seite der Gleichung addiert wird. Die linke Seite der Gleichung beschreibt nun alle im Element entstandenen Kräfte zufolge einer Bewegung in den Knoten.

Da in einem System aus finiten Elementen aber auch noch weitere externe Kräfte an den Knoten angreifen können, werden diese im hier als äußerer Kraftvektor  $\underline{F}$  auf der rechten Seite der Bewegungsgleichung berücksichtigt.

Schlussendlich kann man die allgemeine Bewegungsgleichung in Matrizenform, eines Finite Elemente Systems, folgendermaßen anschreiben.

$$[M] \cdot \underline{\ddot{U}} + [C] \cdot \underline{\dot{U}} + [K] \cdot \underline{U} = \underline{F}$$

### 2.3.2 Lösen der Bewegungsgleichung und Interpretation

Vereinfacht man die Gleichung von oben und streicht den Dämpfungsterm  $[C] \cdot \underline{\dot{U}}$  wieder heraus und zieht die äußeren Kräfte, falls vorhanden und von den Knotenpunktsverschiebungen abhängig, auf die linke Seite. Das heißt also für  $\underline{F} = -[K_F] \cdot \underline{U}$  eingesetzt werden kann, dann sieht die Bewegungsgleichung wie folgt aus:

$$[M] \cdot \underline{\ddot{U}} + \{ [K] + [K_F] \} \cdot \underline{U} = 0$$

Es handelt sich jetzt um eine harmonische Differentialgleichung 2. Ordnung, in welche der harmonische Schwingungsansatz eingesetzt werden kann:

$$\underline{U}(t) = \underline{\hat{U}} \cdot e^{i\omega t}$$

$$\underline{\dot{U}}(t) = i \cdot \omega \cdot \underline{\hat{U}} \cdot e^{i\omega t}$$

$$\underline{\ddot{U}}(t) = -\omega^2 \cdot \underline{\hat{U}} \cdot e^{i\omega t}$$

$$[M] \cdot -\omega^2 \cdot \underline{\hat{U}} \cdot e^{i\omega t} + \{ [K] + [K_F] \} \cdot \underline{\hat{U}} \cdot e^{i\omega t} = 0$$

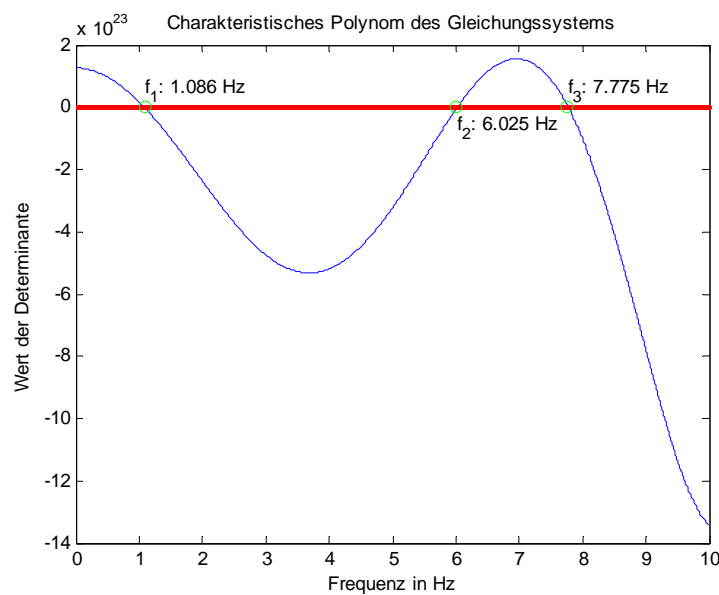
Es kann nun  $\underline{\hat{U}} \cdot e^{i\omega t}$  heraus gekürzt werden und es bleibt folgender Ausdruck stehen

$$- [M] \cdot \omega^2 + \{ [K] + [K_F] \} = 0$$

Das Gleichungssystem besitzt dann eine Lösung, wenn die Determinante des linken Teiles verschwindet. Berechnet man nun die Determinante in Abhängigkeit der Eigenkreisfrequenz  $\omega$  beziehungsweise der Frequenz  $f = \frac{\omega}{2\pi}$ , so ist das die Funktion des charakteristischen Polynoms, dessen ausgewerteten Nullstellen die Eigenfrequenzen des Systems darstellen.

$$\det\{-[M] \cdot \omega^2 + [K] + [K_F]\} = 0$$

Die Abbildung 2.22 zeigt den Determinantenwert über der Frequenz für ein einfaches System mit 5 Knoten. Die Abbildung soll nur zeigen, dass das System genau da seine Eigenfrequenzen besitzt, wo das charakteristische Polynom (blaue Kurve) seine Nullstellen (Schnittpunkte mit der roten Linie) hat. Die Berechnung des selben Systems mit ANSYS liefert nahezu die gleichen Ergebnisse. Für eine detailliertere Ausführung der Berechnung, den dazugehörigen MATLAB-Code und die Simulation mit ANSYS Mechanical APDL muss auf [11] verwiesen werden.



**Abbildung 2.22: Determinantenwert über der Frequenz**

Für die Herleitung komplexerer Elemente muss auf das Skriptum aus „Höherer Festigkeitslehre und FE-Methoden“ [12] verwiesen werden.

## 2.4 Grundlagen einer experimentellen Modalanalyse (FFT)

### 2.4.1 Durchführung

Ähnlich wie das Anschlagen einer Stimmgabel, so kann man sich die experimentelle Frequenzanalyse vorstellen. Dazu wird das zu untersuchende Bauteil wie in Abbildung 2.23 entweder mit einem Impulshammer angeschlagen oder durch einen Shaker angeregt.

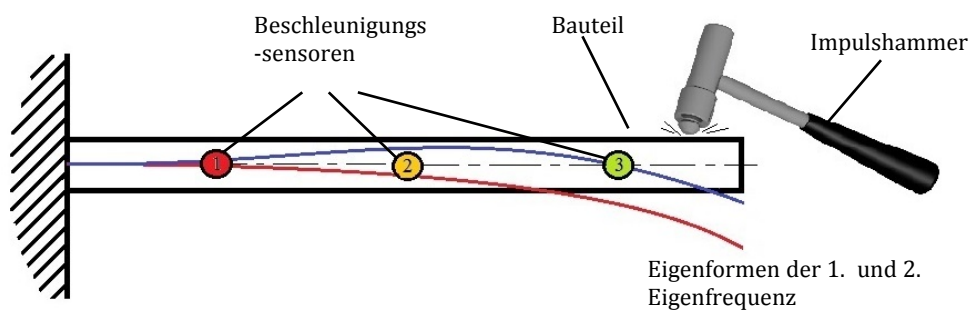


Abbildung 2.23: Schema der experimentellen Modalanalyse

Das System kann, wie in Abbildung 2.24 dargestellt, als sogenanntes „Black Box Model“ betrachtet werden. Mit dem Signal der Anregung  $F_j(\omega)$  als Eingangsgröße und den Signalen der Beschleunigungssensoren  $X_i(\omega)$  als Ausgangsgrößen können nun die Übertragungs- oder Vergrößerungsfunktionen  $H_{ij}(\omega)$  für jeden Beschleunigungssensor errechnet werden.

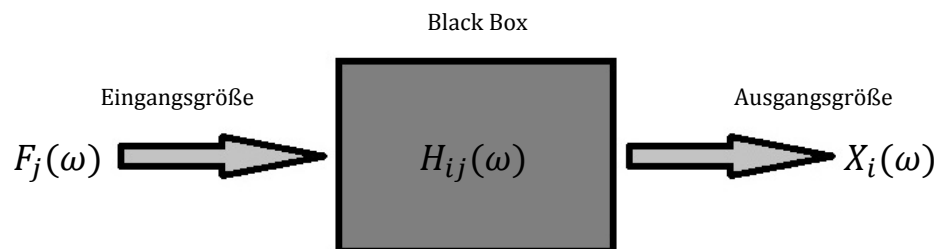


Abbildung 2.24: Black-Box Modell



Es folgt daraus der Zusammenhang:

$$H_{ij}(\omega) = \frac{X_i(\omega)}{F_j(\omega)}$$

### 2.4.1 Ergebnisse und Interpretation

$H_{ij}(\omega)$  ist eine komplexe Funktion und kann somit in Amplitude und Phasenlage oder in Real- und Imaginärteil zerlegt werden. Bei schwacher Dämpfung, welche für eine Modalanalyse vorausgesetzt werden muss, hat der Imaginärteil von  $H_{ij}(\omega)$  bei den Eigenfrequenzen des Systems seine Maxima. Die folgende Abbildung 2.25 zeigt schematisch den Imaginärteil der Vergrößerungsfunktion  $H_{ij}(\omega)$  über der Eigenkreisfrequenz  $\omega$  für das System in Abbildung 2.23. Die drei Kurven stellen die Werte der drei Beschleunigungssensoren dar. Aus diesem Diagramm zeigt sich, wie wichtig eine zweckmäßige Positionierung der Beschleunigungssensoren ist. Allein mit dem Beschleunigungssensor Nr. 3 (grüne Kurve) wäre die 2. Eigenfrequenz nur sehr schwer herauszulesen. Zudem kann mit diesen Werten auf die zugehörigen Eigenformen der eruierten Eigenfrequenzen zurückgeschlossen werden.

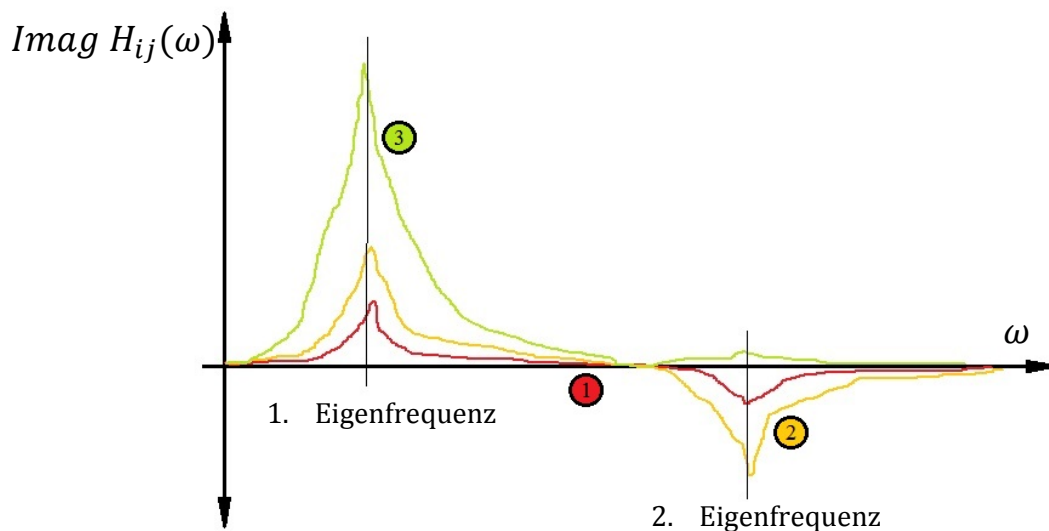
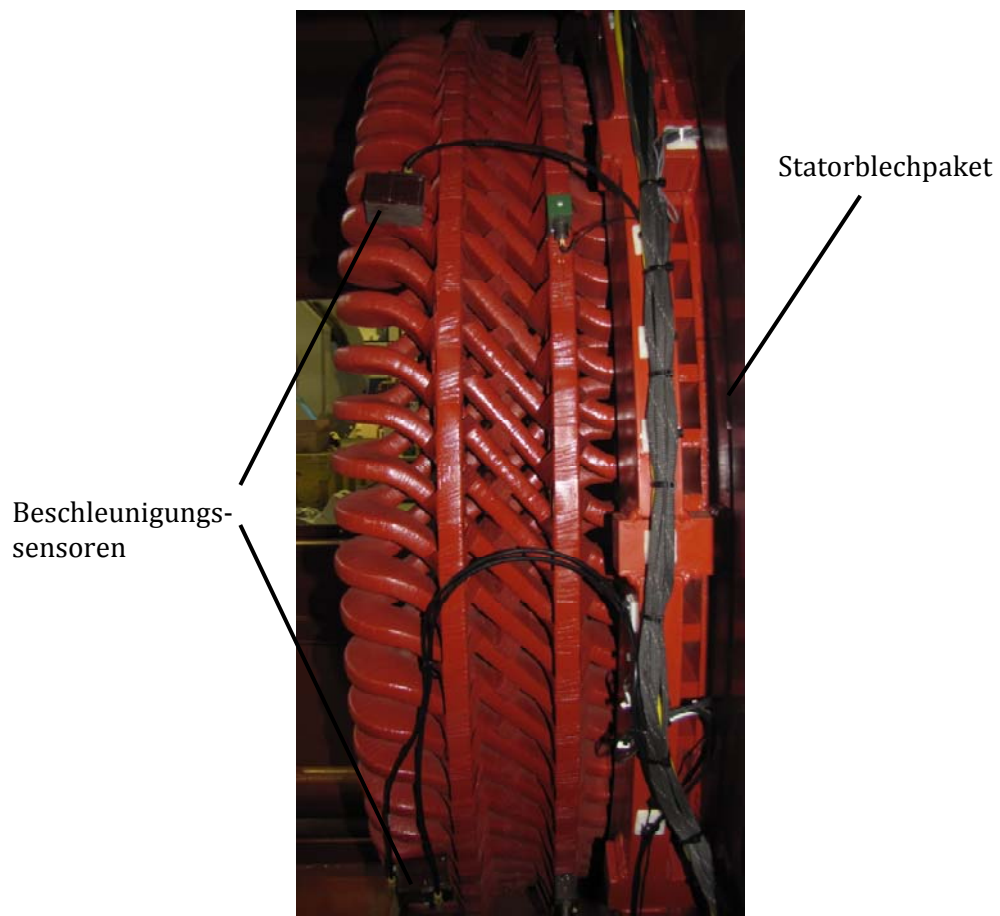


Abbildung 2.25: Imaginärteil der Vergrößerungsfunktion schematisch

Für den untersuchten Generatorwickelkopf, sieht der Versuchsaufbau wie in Abbildung 2.26 dargestellt aus.



**Abbildung 2.26: Messung an einem spulengewickelten Wickelkopf [6]**

Die Ergebnisse dieser Messung sind in Tabelle 2.1 zusammengefasst. Es wurden dabei nur jene Eigenfrequenzen ausgewählt und angeführt, welche für den Vergleich mit der Simulation geeignet sind, also vergleichbare Eigenformen aufweisen.

**Tabelle 2.1: Ergebnisse der Messung**

Nummer	Eigenfrequenz Mittelwert
1	166 Hz
2	177 Hz
3	189 Hz
4	219 Hz
5	256 Hz
6	289 Hz

Die Auswertung der verschiedenen Aufnehmer liefert mehrere Werte für eine Eigenfrequenz. Die Werte in der Tabelle sind Mittelwerte dieser verschiedenen Frequenzen. Neben den Tabellenwerten gibt es zu jedem einzelnen Frequenzwert eine animierte Darstellung der dazugehörigen Eigenform. Die Abbildung 2.27 zeigt stellvertretend einen Schnappschuss einer der Animationen für die erste Eigenfrequenz. Diese animierten Eigenformen sind mit den animierten Eigenformen aus der ANSYS-Simulation sehr gut vergleichbar, was die Zuordnung der Messergebnisse zu den Simulationsergebnissen erheblich vereinfacht.

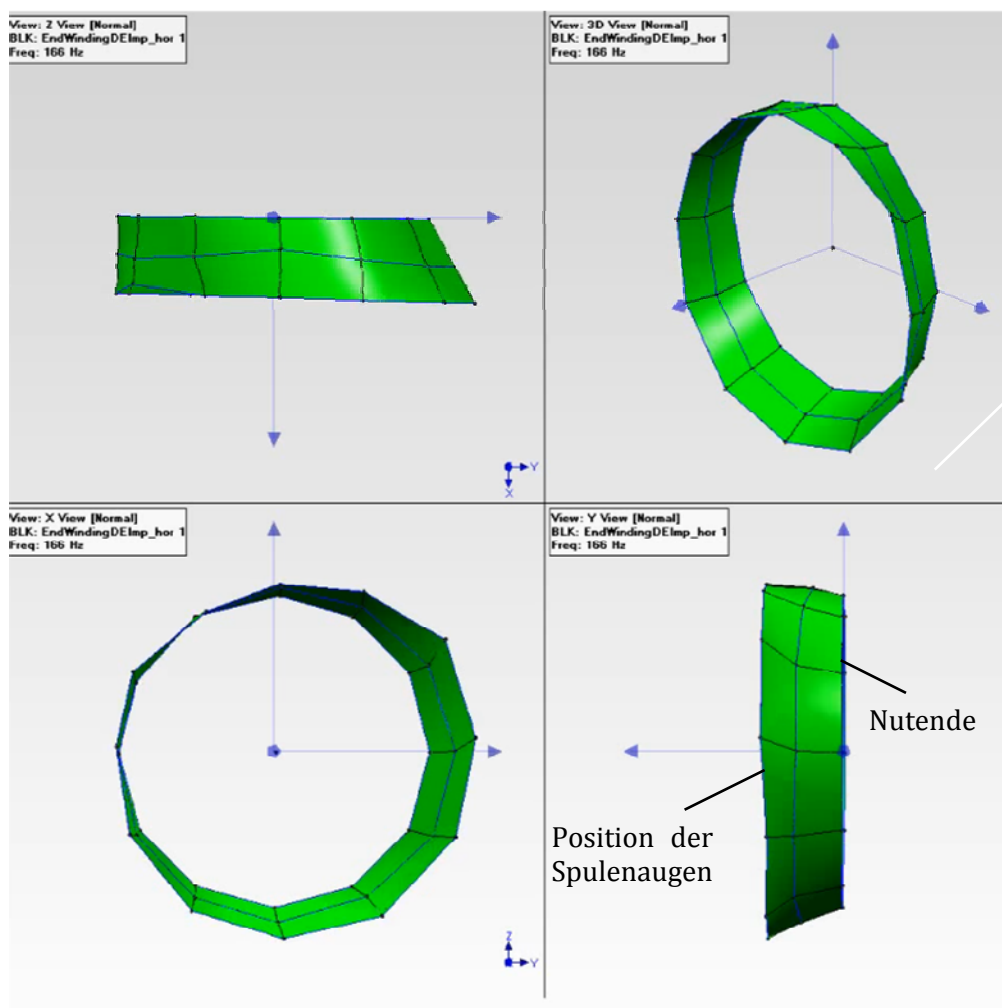


Abbildung 2.27: Schnappschuss einer animierten Eigenform [6]

## 3 Ausführung und Arbeitsweise des Programms

### 3.1 Software ANSYS Mechanical APDL

Zur Lösung von technischen Problemstellungen liefert das Programm ANSYS zahlreiche Möglichkeiten. Für nahezu alle Arten von technischen Aufgaben in der Struktur- und Fluidmechanik, Elektrotechnik, Akustik und Thermodynamik gibt es Lösungsmethoden. Die Modalanalyse zur Ermittlung der Eigenfrequenzen kann dabei mit zwei unterschiedlichen Varianten des Programms, der ANSYS Workbench sowie ANSYS Mechanical APDL, ausgeführt werden.

ANSYS Workbench zeichnet sich durch eine bedienerfreundliche Oberfläche aus, gibt aber dem Anwender weniger Möglichkeiten auf die Prozesse und die Vernetzung einzugreifen als dies noch bei ANSYS Mechanical APDL der Fall ist. APDL bedeutet ANSYS Parametric Design Language und stellt seinerseits eine eigene Programmiersprache dar.

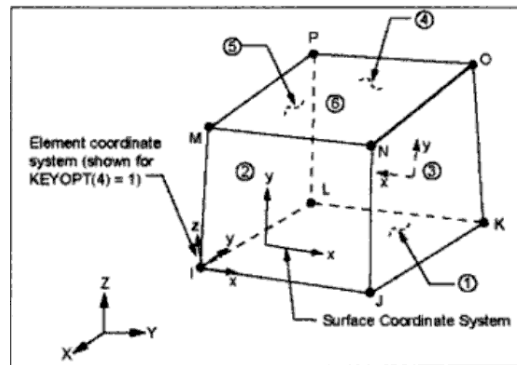
Das Berechnungsprogramm für spulengewickelte Wickelköpfe wurde, wie bereits das Berechnungsprogramm für stabgewickelte Generatorwickelköpfe und von der Firma ANDRITZ gefordert, mit ANSYS Mechanical APDL erstellt. [13], [3]

#### 3.1.1 Verwendeter Lösungsalgorithmus

Für die Modalanalyse in ANSYS Mechanical APDL gibt es einige Lösungsmethoden, die sogenannten „mode extraction methods“, zur Auswahl. Für die Berechnung der Wickelköpfe kommen 2 Methoden in Frage. Zum einen die Block Lanczos-Methode und die QR-Damped Methode. Im Zuge der Simulation erwies sich die Block Lanczos-Methode als zuverlässig und stabil. Vergleiche der Ergebnisse mit der QR-Damped Methode ergaben keine Abweichungen in den Ergebnissen, weshalb die Lösung nach wie vor mit der Block Lanczos-Methode ausgeführt wird.

### 3.1.2 SOLID 45; Volumenelement

„Für allgemeine 3-dimensionale Bauteile sind Volumenelemente zu verwenden. Damit kann die Geometrie direkt so, wie sie in den heute üblichen 3D-CAD-Programmen erstellt wird, übernommen und in Elemente aufgeteilt werden.“



**Abbildung 3.1: Skizze des SOLID 45 Volumenelements [13]**

Die Ergebnisse dieser Elemente sind am besten, wenn die Längen der Elementkanten etwa gleich (also würfelförmig) sind. Dies bedeutet, dass für dünnwandige Bauteile oder dünne und schlanke Strukturen entweder sehr viele Elemente mit einer feinen Aufteilung des Netzes in Kauf genommen werden müssen oder die Ergebnisgenauigkeit eingeschränkt ist. Aus der Erfahrung sind Kantenlängenverhältnisse von 1:10 durchaus üblich und im Ingenieuralltag als akzeptabel zu bezeichnen. Bei weiter davon abweichenden Verhältnissen sollte genauer nachgedacht werden, ob die Lasten, die Spannungen und insbesondere die Gradienten und die Ergebnisse (Verschiebungen oder Spannungen oder Auflagerreaktionen) dadurch beeinträchtigt werden. Hierzu können auch Variantenuntersuchungen dienen.“ Siehe [14] S.150.

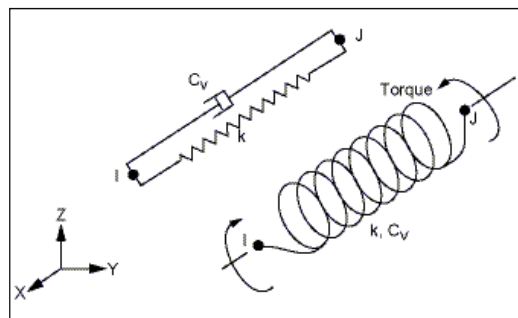
Die relevanten Eigenschaften des Elementtyps sind:

- Jeweils 3 Freiheitsgrade an den Knoten I, J, K, L, M, N, O und P:  $u_x, u_y, u_z$
- Geometrie 3-dimensional, geeignet für allgemeine, unregelmäßige Bauteile
- Beanspruchung: Normalspannungen und Schubspannungen in allen drei Raumrichtungen (ebenso Dehnungen in allen 3 Raumrichtungen)
- Berechnungsaufwand relativ hoch
- Trilineare Ansatzfunktion für Verschiebung in drei Richtungen
- Anwendungsgebiet bei Bauteilen mit vergleichbar gleich großen Abmessungen in allen drei Raumrichtungen

Vergleiche mit [13].

### 3.1.3 COMBIN 14; Feder-Dämpfer-Element

Dieses masselose Element, welches in Abbildung 3.2 skizziert ist, kann sowohl als Längs- als auch als Torsions-Feder-Dämpfereinheit in 1D-, 2D- und 3D-Anwendungen angewendet werden. Es besteht aus zwei Knotenpunkten und kann Längsdehnungen und Verdrehungen in bzw. um seine Längsachse aufnehmen, jedoch keine Biegung.



**Abbildung 3.2: Skizze des COMBIN 14 Elements [13]**

Die Federkraft ist linear proportional zur Dehnung bzw. zur Verdrehung. Sollte ein nichtlineares Verhalten der Feder gefordert sein, muss auf das COMBIN 40 Element verwiesen werden. Die Dämpfung kann entweder geschwindigkeitsproportional mit der ersten Dämpfungskonstante CV1 oder durch einschalten der zweiten Dämpfungskonstante CV2 auch nichtlinear erfolgen. Durch setzen der KEYOPT's können die entsprechenden Funktionen ausgewählt werden. Als Output gibt das Element eine entsprechende Kraft oder ein Moment als Antwort auf eine Verschiebung zurück.

Die relevanten Eigenschaften des Elementtyps sind:

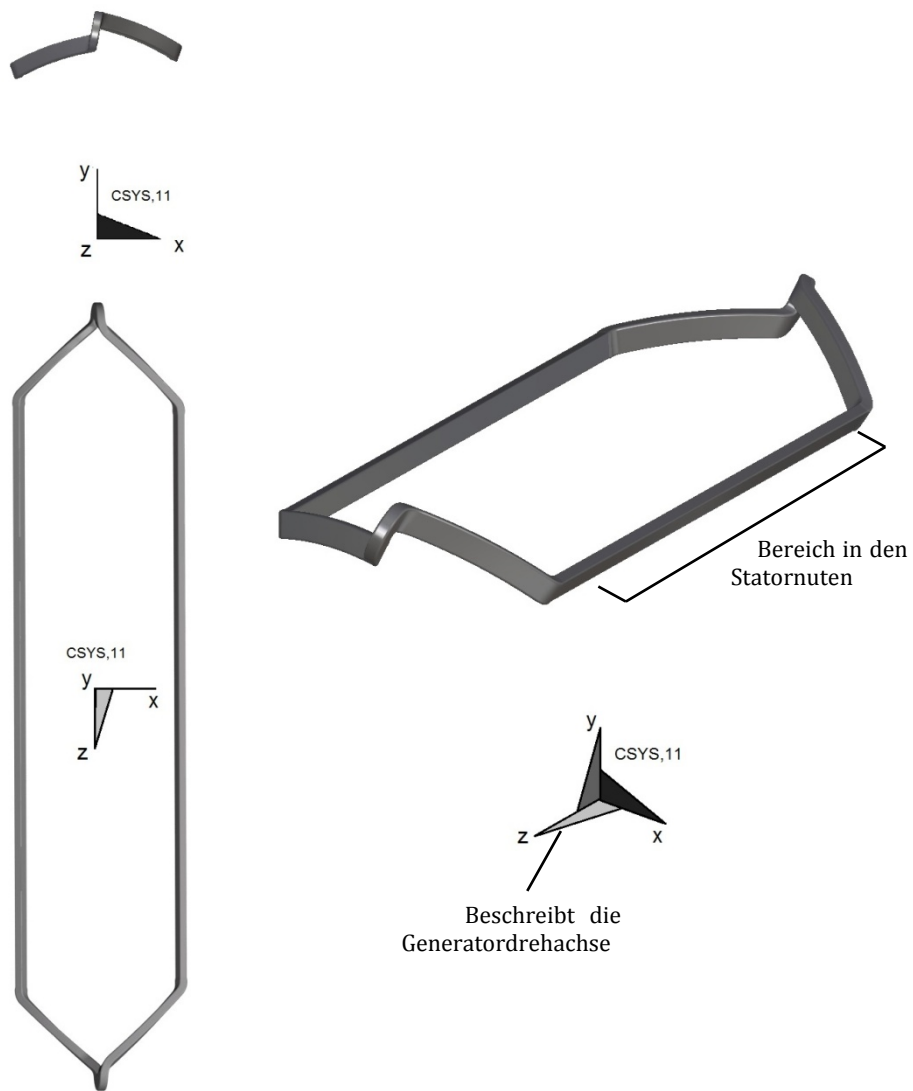
- Die Längsfeder „arbeitet“ nur über die Länge ihrer Ausdehnung
- Das Element erlaubt nur eine gleichmäßige Spannung in der Feder
- Die Feder- oder Dämpfungseigenschaft kann durch nullsetzen der Federsteifigkeit  $K$  oder der Dämpfungskonstanten  $CV$  weggeschaltet werden.
- Wenn  $CV2$  nicht gleich Null ist, dann reagiert das Element nichtlinear und erfordert eine iterative Lösung
- Für die Modalanalyse muss die Dämpfungseigenschaften ausgeschaltet werden,  $CV1$  und  $CV2$  müssen Null sein

Vergleiche mit Literaturquelle [13].

## 3.2 Anforderung an die vorgegebene Geometrie

Die Eingabe der Geometriedaten in ANSYS erfolgt mittels einer Parasolid-Datei (X\_T-Datei (.x\_t)), welche bereits im Raum so positioniert sein muss, dass eine beliebige Achse des, der Geometrie zugehörigen rechtwinkligen, kartesischen Koordinatensystems die Drehachse des Generators beschreibt und in Achsrichtung seinen Ursprung in der Mitte des Blechpaketes hat.

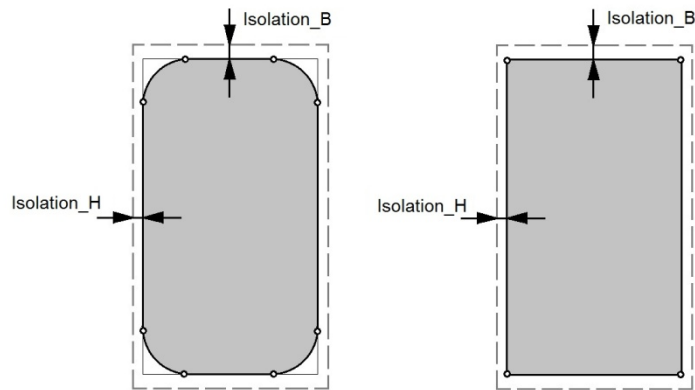
Abbildung 3.3 zeigt die Geometrie im Raum mit dem Koordinatensystem  $CSYS,11$ , welches gleich am Anfang des Programmes erzeugt wird und gegebenenfalls angepasst werden muss, sodass es im Bezug zur Geometrie so zu liegen kommt, wie im Bild ersichtlich.



**Abbildung 3.3: Die Geometrie im Raum mit Koordinatensystem CSYS,11**

Die im Programm verarbeiteten Entitäten sind ausschließlich Keypoints (KP) und Linien (LINE) alles andere (Knoten, Flächen, Volumina) wird gleich zu Beginn gelöscht. Es wird davon ausgegangen, dass der näherungsweise rechteckige Querschnitt der Spule an jeder seiner 4 (abgerundeten) Ecken von gleich vielen Linien (Konturlinien bzw. Konturlinienzüge genannt) begrenzt wird. Abbildung 3.4 zeigt welche Ausgangssituationen möglich sind.



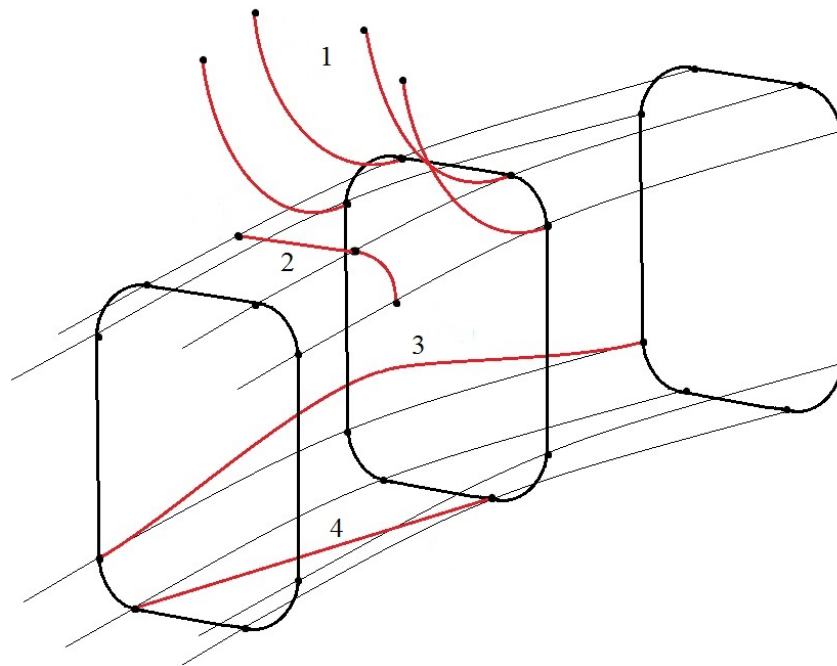


**Abbildung 3.4: Mögliche Ausgangssituationen für den Leiterquerschnitt**

Das Linke der beiden Querschnitte zeigt einen Stabaufbau mit 8 Konturlinien, wie bei der aktuellen Parasolid-Datei gegeben. **Isolierung B** an der Breitseite und **Isolierung H** an der hohen Seite sind im aktuellen Fall beide Null. Der rechte Querschnitt wäre eine Geometrie mit nur 4 Konturlinien. Auch diese Art von Ausgangsgeometrie wäre denkbar. Wichtig ist eben, dass beide Varianten in einer Parasolid-Datei nicht gemischt vorkommen.

Abbildung 3.5 zeigt schematisch mit den dicken roten Linien jene 4 Fälle die Probleme verursachen. Die dünnen Schwarzen Linien sind die Konturlinien, die dick schwarz eingezeichneten Linien die erlaubten Querlinien, welche das Programm selbständig löscht.

1. Linien mit offenem Ende, welche vom Konturlinienzug abzweigen
2. Eine zusätzliche Unterbrechung von einzelnen Konturlinien mit oder ohne zwischenliegenden Querlinien
3. Abzweigende Linie mit geschlossenem Ende
4. Linien zwischen nacheinander folgenden Unterbrechungen und unterschiedlichen Konturlinien

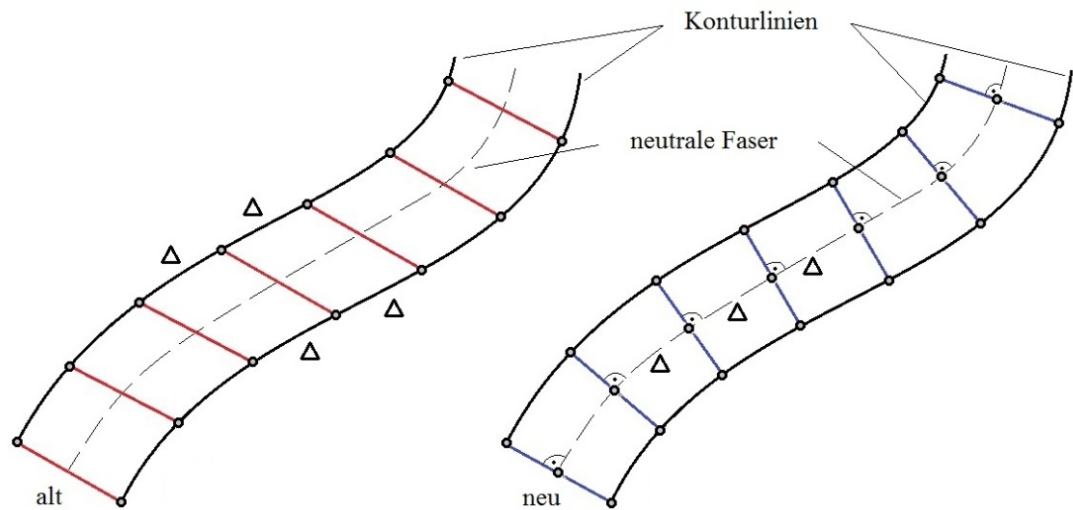


**Abbildung 3.5: Erlaubte und nicht erlaubte Linien zwischen den Konturlinienzügen**

### 3.3 Anforderungen an das Netz

Alle Anbauteile wie Stützringe, Distanzstücke, Gewölbekeile, etc. benötigen ein Netz mit 2 mal 2 Elementen im Stabquerschnitt und genau definierte Knotennummern zwischen beziehungsweise an den Eckpunkten der Elemente.

Der kritischste Punkt beim Netzaufbau ist das Spulenaug. Der Vernetzungsalgorithmus für die Stabwicklungen kann den stark gekrümmten Teil des Spulenauges nicht vernetzen, weshalb hier ein grundsätzlich anderes Verfahren angewendet werden muss. Näheres dazu unter Kapitel 3.6 Modellaufbau und Netzdetails.



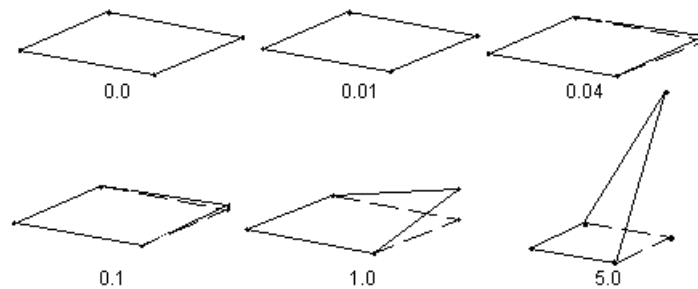
**Abbildung 3.6: Alter und neuer Algorithmus**

Das Hauptaugenmerk bei der Netzgenerierung ist auf die frühe Vermeidung von Netzfehlern durch möglichst sinnvolles Positionieren der Knoten (NODE) gerichtet. Dies kann nur dadurch erreicht werden indem man nicht wie in Abbildung 3.6 (alt) die Konturlinien der Rohgeometrie in gleich viele Stücke teilt, sondern wie in Abbildung 3.6 (neu) ersichtlich, ausgehend von einer neutralen Faser die Geometrie unterteilt. Bei den, in den Spulenaugen auftretenden, starken Krümmungen versagt hier der alte Algorithmus, da die Elemente zu stark verzerrt werden und folglich Fehlermeldungen auftreten. Der, bei dieser Arbeit entstandene, neue Algorithmus kommt hingegen auch mit stärkeren Krümmungen zurecht, da sich das Netz der Geometrie besser anpasst.

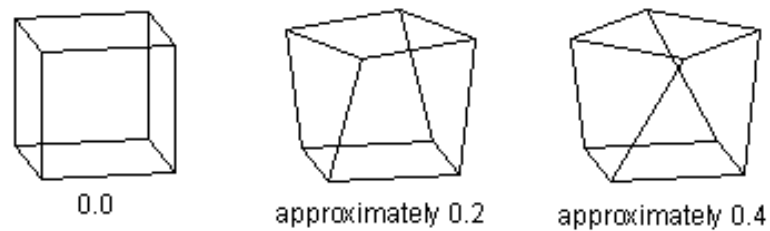
Folgende geometrische Anforderungen an die Elemente können bei schlechter Netzqualität nicht erfüllt werden. Um dem Anwender bei der Verbesserung der Netzqualität zu unterstützen, werden diese Kriterien und die Abhilfemaßnahmen bei deren Nichteinhaltung kurz erläutert.

**1. Warping Factor (Verzerrungsfaktor) ist zu groß:**

Das bedeutet, dass das Element zu stark von seiner idealen Form abweicht. Abbildung 3.7 und Abbildung 3.8 sollen dies anhand von Flächen- und Volumenelementen veranschaulichen.



**Abbildung 3.7: Warping Factor bei ebenen Flächen [13]**



**Abbildung 3.8: Warping Factor bei einem Würfel bzw. 8-Knoten Element [13]**

Abhilfe: Durch geschickte Teilung der Geometrie die gesamten Verzerrungen auf möglichst alle Elemente gleichmäßig aufteilen (zum Beispiel durch Einziehen von Hilfsebenen oder Working Planes), anstatt eine Reihe nahezu würfelförmiger Elemente zu generieren und dem letzten Element fast die ganzen Verzerrungen aufzubürden. Dieses Kriterium machte die Anwendung des alten Algorithmus auf die Spulengeometrie unmöglich und erforderte die Neuentwicklung des jetzt verwendeten neuen Teilungsalgorithmus.

**2. Aspect Ratio (Seitenverhältnis) ist zu groß:**

Für eine gute Qualität der Elemente ist ein Seitenverhältnis von unter 10 anzustreben, weswegen die Teilung im Stab sinnhaft gewählt werden muss, sodass die Elemente möglichst würfelförmig sind (siehe Abbildung 3.9: „links“).

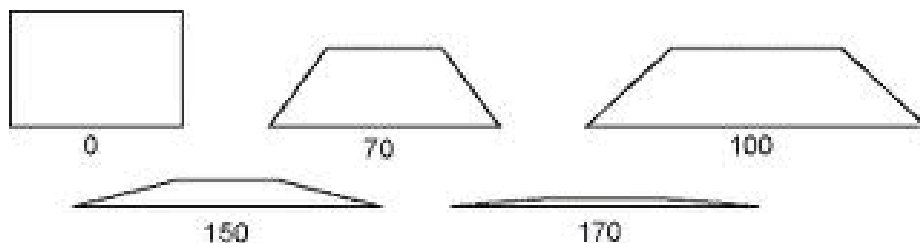


**Abbildung 3.9: Seitenverhältnis bei Rechtecken und Quadern [13]**

Abhilfe: Falls möglich, einziehen zusätzlicher Elementgrenzen. Dieses Kriterium verhindert die zu starke Verfeinerung des Netzes im Bereich des Spulenauges. Hier können aufgrund der Vernetzungsvorschrift für die Leiter keine zusätzlichen Elementgrenzen eingezogen werden. Die Abhilfe dafür ist die Einführung von zwei Leiterabschnitten, sodass die anderen Bereiche unabhängig des Netzes im Spulenaug vernetzt werden können.

**3. Angle Deviation (Winkelabweichung) ist zu groß:**

Wenn die Winkelabweichung zweier idealerweise paralleler Kanten mehr als 150° beträgt, wie in Abbildung 3.10 ersichtlich, dann gibt ANSYS eine Fehlermeldung aus.



**Abbildung 3.10: Winkelabweichung von idealerweise parallelen Kanten [13]**

Abhilfe: siehe Punkt 1

#### 4. Corner Angle (Eckenwinkel) ist zu groß:

ANSYS überprüft auch die Eckenwinkel der Elemente. Abbildung 3.11 zeigt wie Elemente, bei von  $90^\circ$  abweichenden Eckenwinkel, aussehen. Bei Eckenwinkel über  $155^\circ$  würde ANSYS eine Fehlermeldung ausgeben, es ist aber während des Programmierens nie aufgetreten.

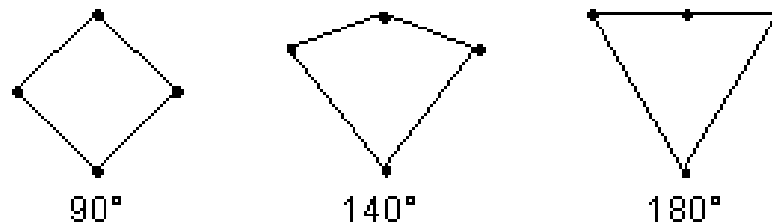


Abbildung 3.11: Eckenwinkel [13]

### 3.4 Knotennummerierung

Die Nummerierung der Knoten (NODE) ist sehr wichtig für die Funktion der danach aufgerufenen Unterprogramme, da diese mit der, bis dahin modellierten, Geometrie über die Knoten gekoppelt werden. Weicht die Nummerierung der Knoten ab, so kann das darauffolgende Unterprogramm die benötigten Kopplungsknoten nicht finden und bricht ab.

Die Einer- und Zehnerstelle bei der Nummerierung der Knoten ist je nach Unterprogramm unterschiedlich festgelegt. Herauszuheben sind hier das Stab-beziehungsweise das Spulenmodell welche diese Nummern für die Netzteilung nach der Leiterachse systematisch verwenden. Die Nummer **xx01** steht dabei für das Nutende die Nummer **xxTeilung+1** für die letzten Knoten am Stabende beziehungsweise für die Knoten an der Teilungsstelle am Spulenaug. Das Unterprogramm Nutendbereich verwendet die Nummern ebenfalls systematisch für die Knoten in der Verlängerung des Stabes hinter dem Nutende im Blechpaket, sowie die feststehenden Knoten rundherum an denen die fiktiven Federn für die elastische Bettung ‚aufgehängt‘ werden. Die Übersichtsdarstellungen Abbildung 3.12 und Abbildung 3.13 zeigen die Position des Nutendes im Bezug zu den beiden Stäben beziehungsweise Spulenteilen und die Nummerierung der Knoten im Leiterabschnitt hinter dem Nutende.

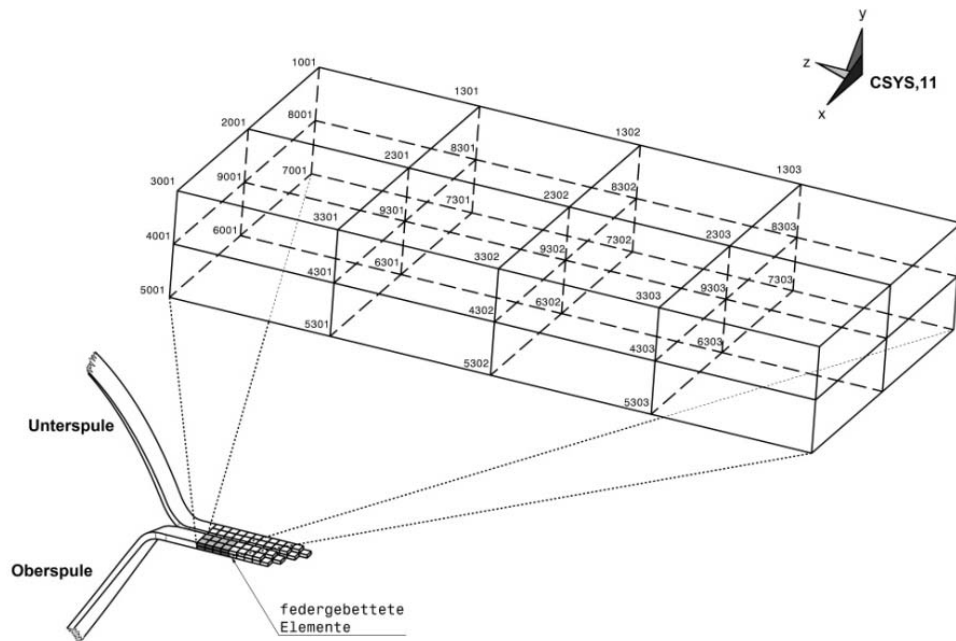


Abbildung 3.12: Position des Nutendes im Bezug zu den Leitern

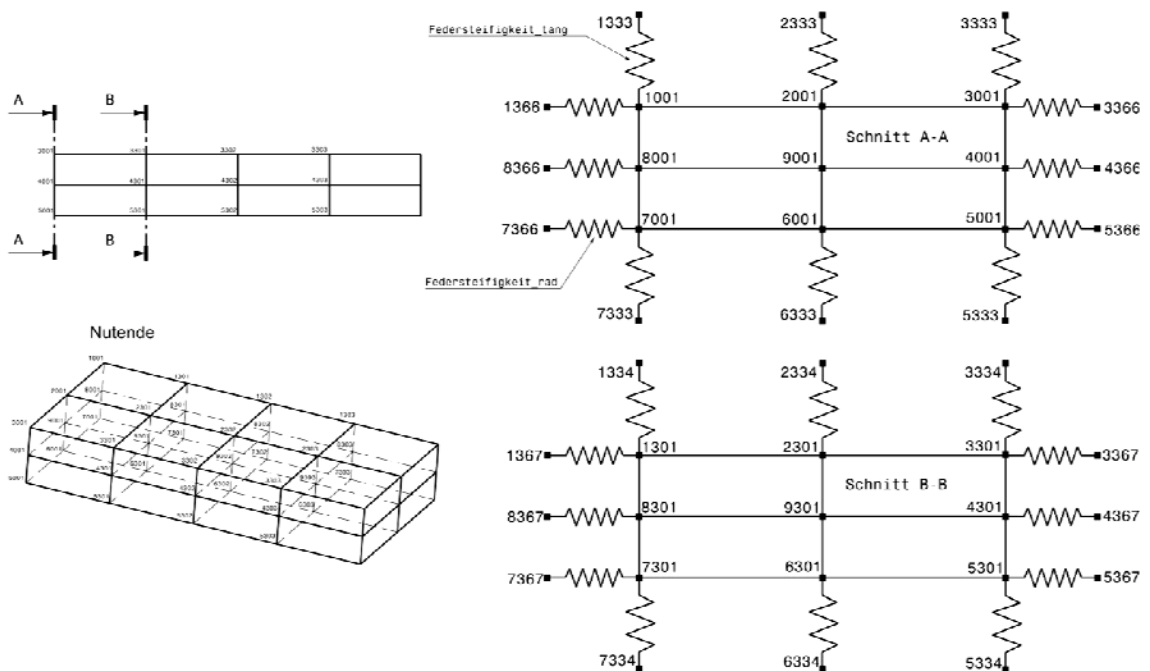


Abbildung 3.13: Federanbindung und Knotennummerierung hinter dem Nutende

Der Schnitt A-A liegt direkt am Nutende, der Schnitt B-B zwischen dem ersten und zweiten Element im Blechpaket. In den Schnitten sieht man das System nach dem die

Knoten nummeriert sind. Die Aufteilung in radiale und tangentiale Federn bedingt, dass nicht mehr als 32 zusätzliche Elemente hinter dem Nutende simuliert werden können. Bei einer Studie im Zuge von [11] erwiesen sich 8 bis maximal 16 zusätzliche Elemente als ausreichend, weshalb das Limit von 32 Elementen praktisch keine Einschränkung bedeutet.

Nachstehende Tabelle zeigt die Bedeutung der Hunderterstelle mit den Knotennummierungen und den dazugehörigen Komponenten bei Stab- und Spulenwicklungen. Die Knotennummern der verschiedenen Komponenten sind an der Hunderterstelle zu unterscheiden.

Die Bezeichnung erfolgt folgendermaßen:

1347 gehört z. B. zur 3xx- Gruppe und ist folglich ein Knoten im Nutende. Wie in Abbildung 3.13 ersichtlich muss dieser Knoten ein Ankerknoten für eine tangentiale Feder an der ersten Konturlinie zwischen 14. und 15. Element (falls es dieses gibt) hinter dem Nutende sein. Tabelle 3.1 zeigt die Nummerierung im Vergleich.

**Tabelle 3.1: Knotennummerierung**

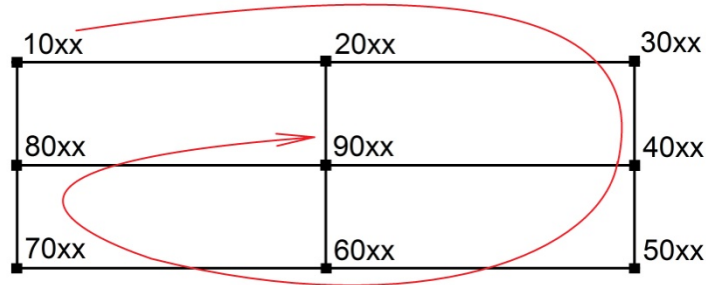
Knotennummer	Stabwicklung (bar)	Spulenwicklung (coil)
0xx	Wicklungsstab	Spulenteil
1xx	Stützring außen	Stützring außen
2xx	-	-
3xx	Nutende	Nutende
4xx	Verbindungsflaschen	-
5xx	Stützring innen	Stützring innen
6xx	Distanzring	
7xx	Distanzstück Oberstab (a)	Distanzstück Oberstab (a)
8xx	Distanzstück Unterstab (i)	Distanzstück Unterstab (i)
9xx	-	Gewölbekeile

Die freien Plätze bzw. Hunderterstellen wären noch für weitere Anbauteile nutzbar.

Die Tausenderstelle in der Knotennummerierung hat insbesondere eine Bedeutung beim Netz im Leiter. Hier werden die 4 Konturlinien an den Ecken der Geometrie, sowie jeweils eine Linie, also insgesamt 4 Linien dazwischen und eine Linie in der neutralen Faser, in Summe 9 Linien durch die Tausenderstelle in der Nummerierung unterschieden.



Folgende Abbildung 3.14 zeigt schematisch, beim Blick auf den Leiterquerschnitt gegen die **CSYS,11**-Z-Achse die Durchnummerierung der Konturlinien. Zwischen den Knoten erkennt man die Struktur des Netzes mit 2 x 2 Elementen im Querschnitt.



**Abbildung 3.14: Nummerierung der Konturlinien im Leiterquerschnitt**

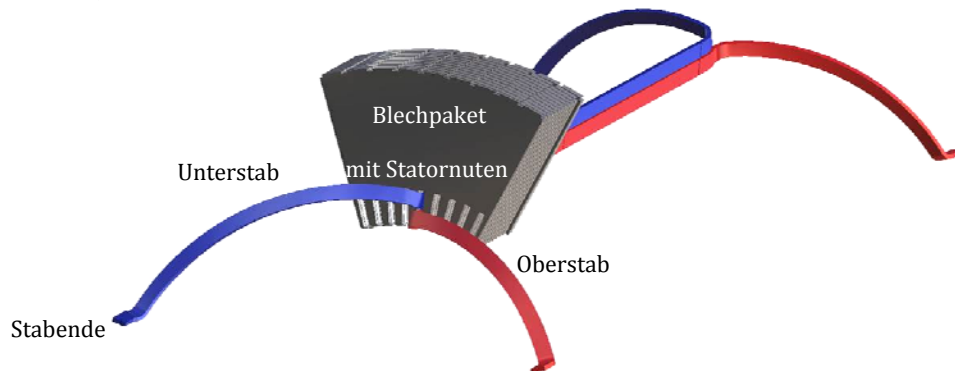
Um beim Mustern um die Drehachse keine Kollisionen in der Nummerierung zu bekommen, wurde die Konstante **off\_Stb** eingeführt. Diese bewirkt, dass jeweils zwischen zwei Statornuten die Knotennummern um **off\_Stb = 10000** höher werden.

Für die Unterscheidung der beiden Stäbe (Ober- und Unterstab) beziehungsweise der beiden Spulenteile (Ober- und Unterspule) ist **off\_OS\_US** eingeführt worden. **off\_OS\_US** ist das Produkt aus der Anzahl aller Statornuten und **off\_Stb**, sodass die größte Knotennummer in allen Oberspulenteilen immer kleiner als die kleinste Unterspulenknotennummer bleibt.

Jede Abweichung von diesen Nummerierungsschemata bewirkt, dass die weiteren, auf diesem System aufbauenden Unterprogramme nicht ordnungsgemäß oder gar nicht funktionieren. Die Nummerierung der Knoten steigt also in den Einer- und Zehnerstellen immer vom Nutende aus an, die Hunderterstellen beschreiben die Komponenten, die Tausenderstellen die jeweilige Konturlinie und alle höheren Stellen sind da um Kollisionen zwischen den unterschiedlichen Nuten und zwischen Ober- und Unterspule zu vermeiden.

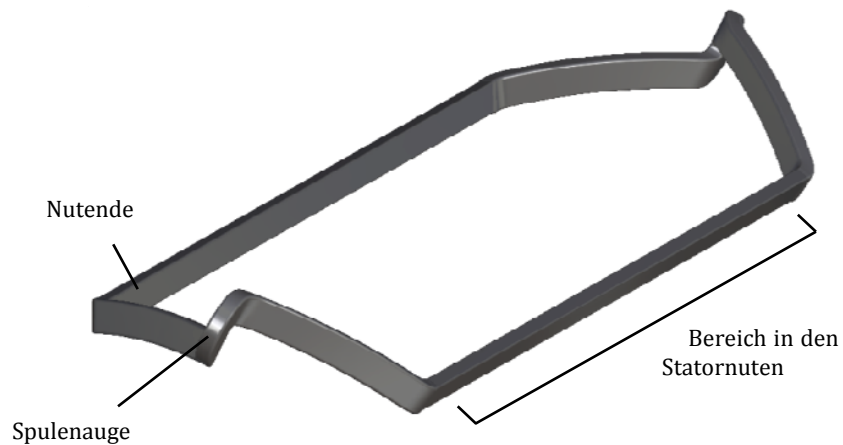
### 3.5 Geometrieaufbereitung

So wie in Abbildung 3.15 gezeigt, sieht die Geometrie der Wicklungsstäbe für das Programm für die Stabwicklungen aus. Jeweils 2 Stäbe, ein Ober- und ein Unterstab liegen in jeder Statornut. Die in Form gebogenen Stababschnitte außerhalb der Nut bilden den Wickelkopf an dessen vorderen Ende die Stabenden verlötet werden.



**Abbildung 3.15: Stabgeometrie mit Blechpaket**

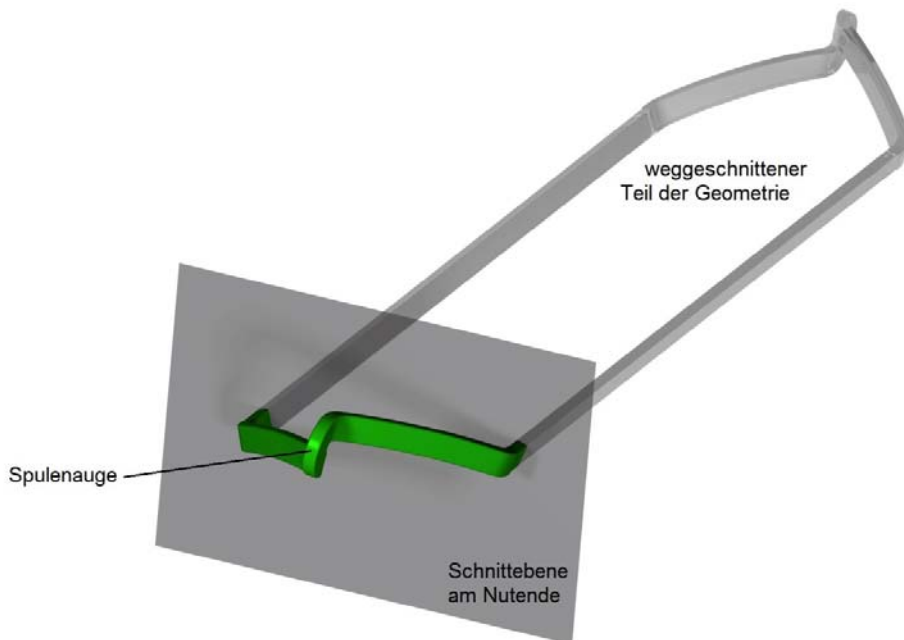
Nun muss auch die Spulengeometrie in eine ähnliche Form gebracht werden.



**Abbildung 3.16: Spulengeometrie**

Abbildung 3.16 zeigt die gesamte Spulengeometrie mit den wichtigsten Positionen.

Da nur jener Teil der Geometrie für die Modellierung benötigt wird, welcher außerhalb der Statornuten liegt, wird zunächst alles hinter dem Nutende weggeschnitten.



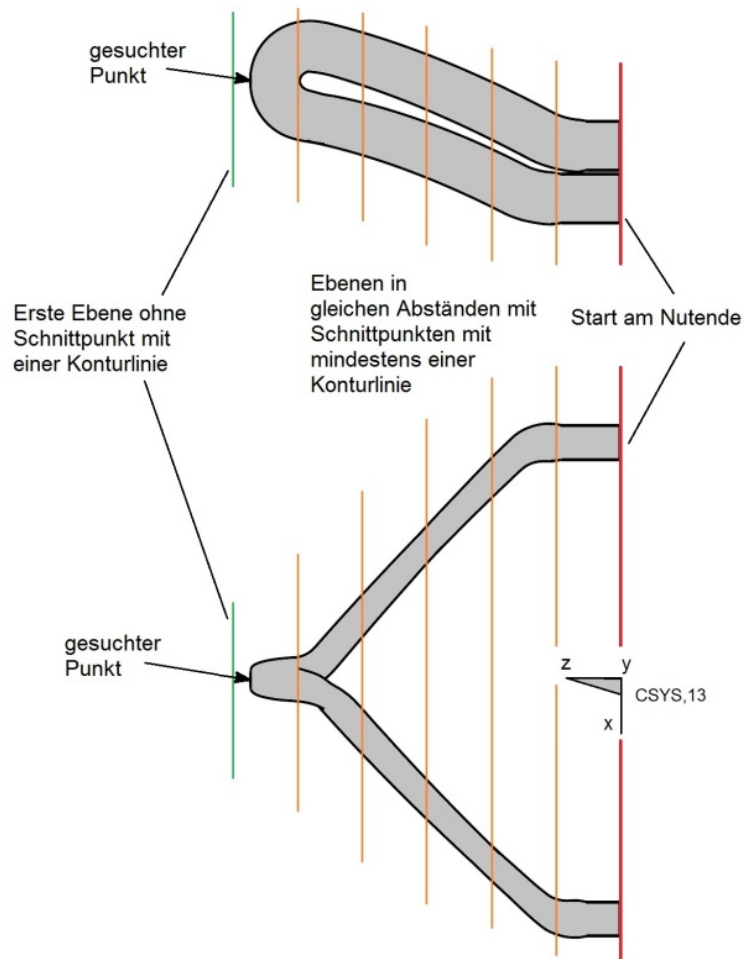
**Abbildung 3.17: Wegschneiden nicht gebrauchter Geometrieteile**

Die Abbildung 3.17 zeigt die transparent eingezeichnete Schnittebene am Nutende, der grüne Teil vor der Ebene wird weiterverwendet, der dahinterliegende Teil weggeschnitten. Die Position des Nutendes muss manuell eingegeben werden.

Nachdem die nicht benötigten Teile weggeschnitten wurden, muss die Geometrie am Spulenauge aufgetrennt werden, damit aus einer durchgehenden Geometrie, zwei getrennte Spulenteile werden können.

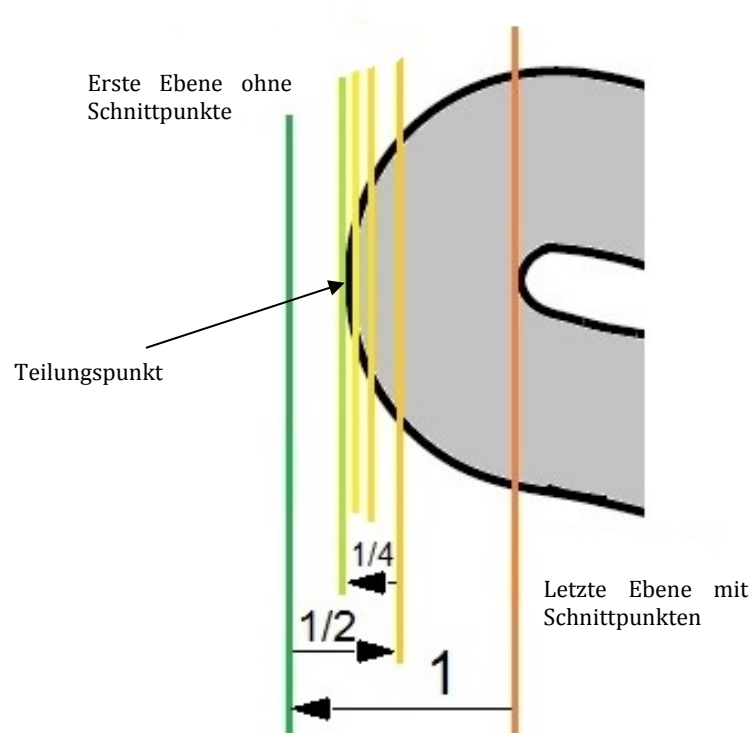
Dazu muss der exponierteste Punkt, also die Mitte des Spulenauges, gesucht werden. Das Problem dabei war, dass außer den 8 Konturlinien, keine anderen Entitäten auf diese Position hindeuteten. Es musste also ein Verfahren gefunden werden, das aus den 8 Konturlinien einen Punkt generiert, welcher mit ausreichender Genauigkeit in der Mitte des Spulenauges sitzt.

Die Abbildung 3.18 veranschaulicht den ersten Teil des Verfahrens zum Suchen des Teilungspunktes. Es wird ausgehend vom Nutende die Geometrie in gleichen Schritten aufgeteilt. Solange eine Schnittebene mindestens einen Schnittpunkt mit einer Konturlinie hat, wird ein neuer Schnitt erzeugt. Findet man aber keinen Schnittpunkt mehr, so wird abgebrochen und mit dem Bisektionsverfahren begonnen.



**Abbildung 3.18: Schrittweises Aufschneiden**

Beim Bisektionsverfahren (Abbildung 3.19) wird der Bereich zwischen der letzten Ebene mit Schnittpunkten und der ersten Ebene ohne Schnittpunkte mit den Konturlinien betrachtet. Dabei wird bei jedem Schritt eine Ebene (Working Plane) um die halbe vorige Schrittweite in CSYS,13 Z-Richtung verschoben. Ob diese Ebene in Richtung Nutende oder in entgegengesetzte Richtung verschoben wird hängt davon ab, ob beim vorigen Schnitt Schnittpunkte gefunden wurden oder nicht. Ist die Schrittweite unter einen Schwellwert (epsilon) gesunken, wird automatisch abgebrochen.



**Abbildung 3.19: Bisektionsverfahren**

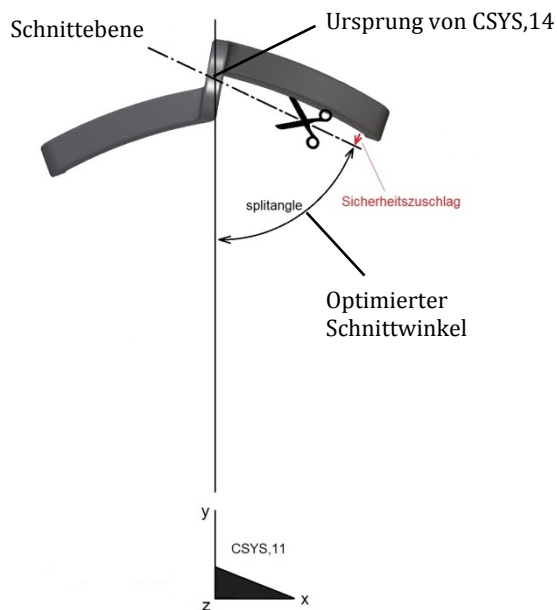
Sobald die Position in Z-Richtung gefunden wurde, wird die Geometrie erneut in Schritten in Richtung Nutende geschnitten und alle X- und Y-Koordinaten der gefundenen Schnittpunkte gemittelt. Die Mittelwerte der X- und Y-Koordinaten sind sodann die Koordinaten des gesuchten Teilungspunktes.

Auf diesen Teilungspunkt wird dann das Koordinatensystem CSYS,14 gelegt, dessen X-Y-Ebene die Schnittebene beschreibt. Die Z-Achse von CSYS,14 wird abhängig davon, welcher Teil der Spule im folgenden Schritt bearbeitet wird, auf die gegenüberliegende Seite gedreht, sodass alle Entitäten mit positiven Z-Koordinaten gelöscht werden können. Am Einfachsten wäre es, wie in Abbildung 3.20 ersichtlich, das Spulenauge mit einer Ebene mit fixem Winkel (z.B.  $45^\circ$ ) zu teilen. Der Nachteil dabei ist aber, dass die Schnittebene dementsprechend steil gewählt werden muss, um verschiedene Spulengeometrien sicher in zwei Teile zu teilen. Die Elemente an der Teilungsebene, die später bei der Vernetzung erzeugt werden, bekommen dort aber eine umso ungünstigere Form, je steiler der Schnittwinkel wird.



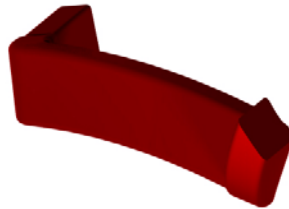
**Abbildung 3.20: Auftrennen am Spulenauge (Konzept 1)**

Die Lösung für dieses Problem zeigt Abbildung 3.21. Der Schnittwinkel wird über die Schnittpunkte zwischen Geometrie und der Schnittebene am Nutende bestimmt und passt sich jeder beliebigen Geometrie so an, dass der Schnittwinkel vom Programm automatisch so flach als möglich gewählt wird. Der berechnete Schnittwinkel ist somit am nächsten dem Ideal eines Schnittes normal auf die Stabachse.

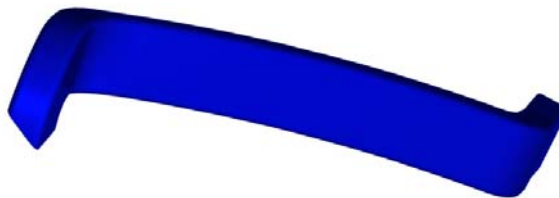


**Abbildung 3.21: Auftrennen am Spulenauge (Konzept 2)**

Nachdem die Konturlinien aufgetrennt wurden und alles außer dem entsprechenden Spulenteil gelöscht wurde, bleibt abhängig vom Programmaufruf entweder die Oberspule (Abbildung 3.22) beziehungsweise die Unterspule (Abbildung 3.23) übrig.

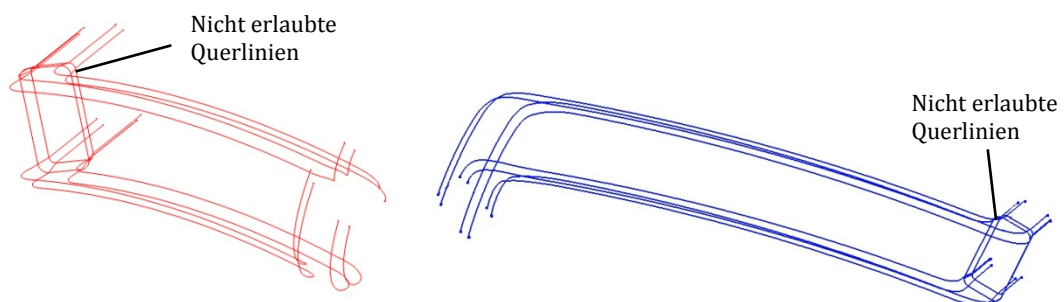


**Abbildung 3.22: Oberspule**



**Abbildung 3.23: Unterspule**

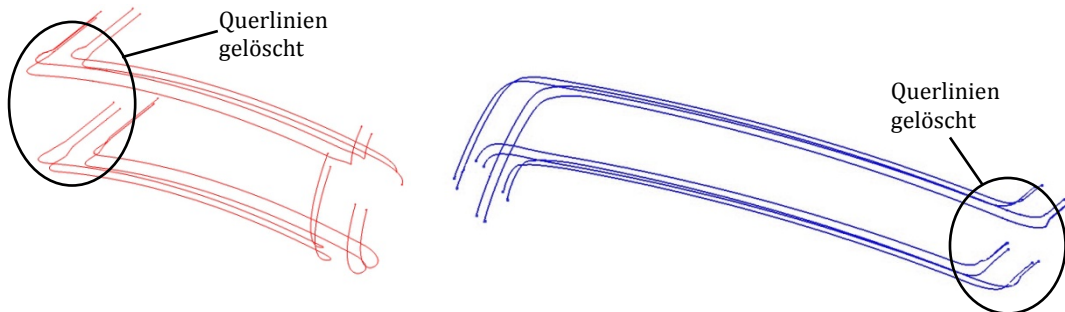
Das tatsächliche Endergebnis sieht ähnlich, wie in Abbildung 3.22 und Abbildung 3.23 gezeigt, aus. Im Modell bestehen die Geometrien nur aus Keypoints (KP) und Linien (LINE) wie in Abbildung 3.24 dargestellt.



**Abbildung 3.24: Spulenteile wireframe**

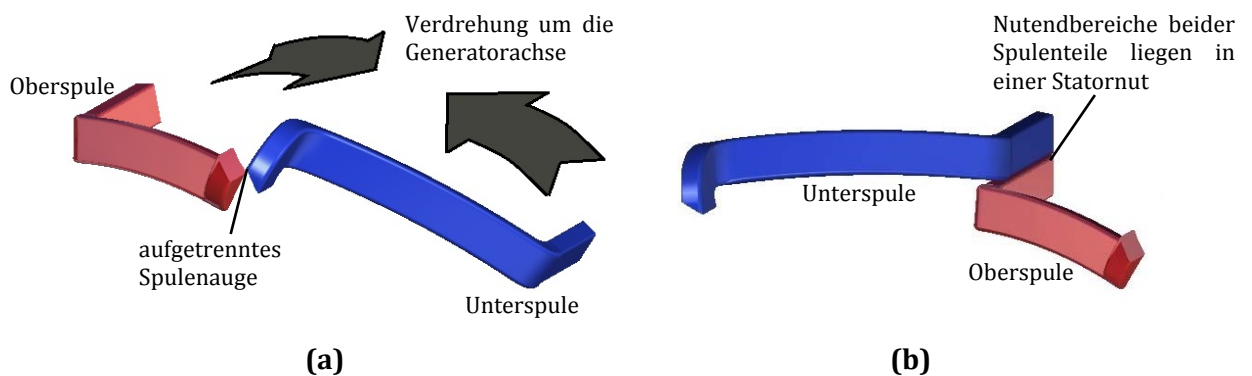
Alle Linien, welche nicht die Konturen beschreiben müssen noch vor dem Vernetzen eliminiert werden. Im Programm werden diese als Querlinien beschrieben.

Abbildung 3.25 zeigt die Geometrie nach dem Löschen der Querlinien und Verschmelzen der Linienzüge, sodass im vorliegenden Fall pro Spulenteil nur mehr 16 Keypoints und 8 Linien übrig bleiben.



**Abbildung 3.25: Spulenteile wireframe ohne Querlinien**

Am Ende der Geometriaufbereitung werden die beiden Spulenteile so um die Generatorachse gedreht, dass die Nutenden von Ober- und Unterspule übereinander liegen. Abbildung 3.26 zeigt dies anhand zweier Skizzen. Die linke Skizze beschreibt den Zustand (a) vor dem Verdrehen, die rechte Skizze den Zustand (b) danach. Die Spulenteile im Zustand (b) gleichen im Aufbau nun den, aus einer Statornut auskragenden, Stäben einer Stabwicklung. Vergleiche dazu mit Abbildung 3.15.



**Abbildung 3.26: Verdrehen der Spulenteile:**

**a) Geometrie vor dem Verdrehen, b) Spulenteile liegen am Nutende in einer Statornut**

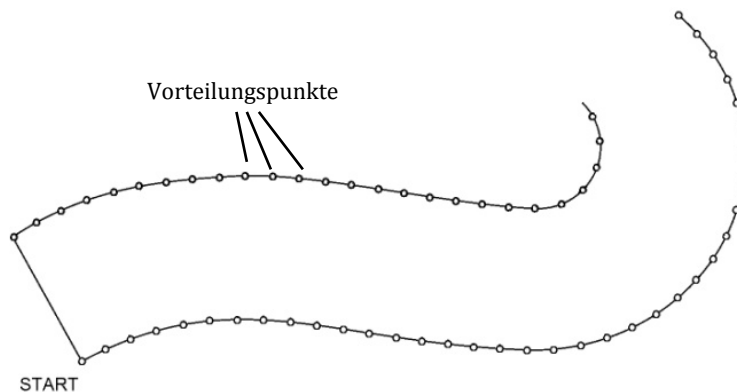


## 3.6 Modellaufbau und Netzdetails

### 3.6.1 Das Stabmodell

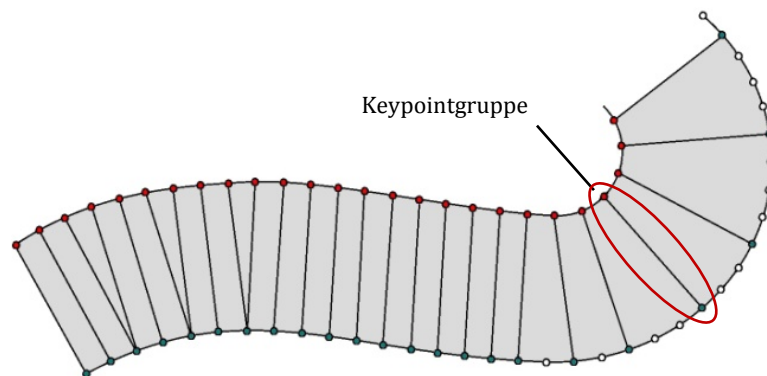
In diesem Kapitel wird beschrieben, wie das Netz in den Spulenteilen erzeugt wird. Eine besondere Herausforderung dabei war, ein in der Struktur den Stabwicklungen identisches Netz aufzubauen, aber auf die besonderen Anforderungen der Spulengeometrie einzugehen. Als Hilfestellung für Programmanpassungen in der Zukunft sind auch nicht zum Ziel führende Konzepte und mögliche Fehlerquellen erläutert.

Ausgehend von einem Spulenteil (Ober- oder Unterspule) wird zunächst die Anzahl der Konturlinien durch Zählen der Keypoints am Nutende ermittelt. Diese Konturlinien werden vorgeteilt, das heißt in gleich viele, gleich große Stücke geteilt. Abbildung 3.27 zeigt schematisch das Vorteilen von 2, von in Wirklichkeit 8, Konturlinien.



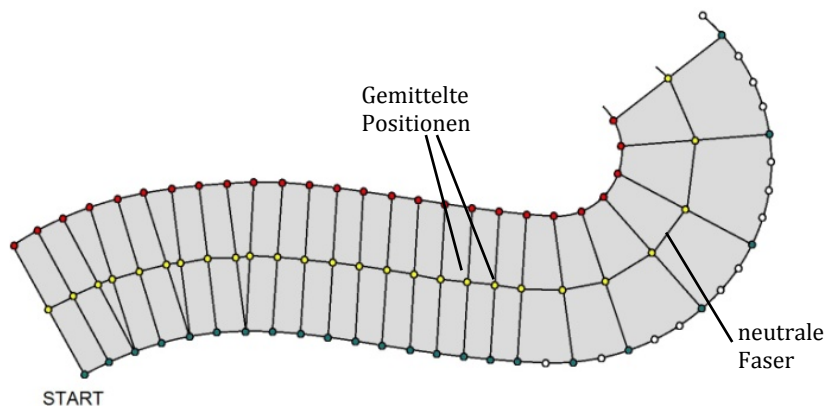
**Abbildung 3.27: Vorteilen der Konturlinien**

Daraufhin werden ausgehend von einer Konturlinie jeweils die nächstgelegenen Vorteilungspunkte auf der nächsten Konturlinie gesucht und gruppiert. Abbildung 3.28 zeigt dies anhand des schon oben angeführten Schemas. Die Linien zwischen den rot und grün markierten Keypoints sollen die Zusammengehörigkeit dieser in einer Keypointgruppe darstellen. Die Gruppen haben bereits die Eigenschaft, dass sie näherungsweise auf einer Ebene normal auf die Stabachse liegen.



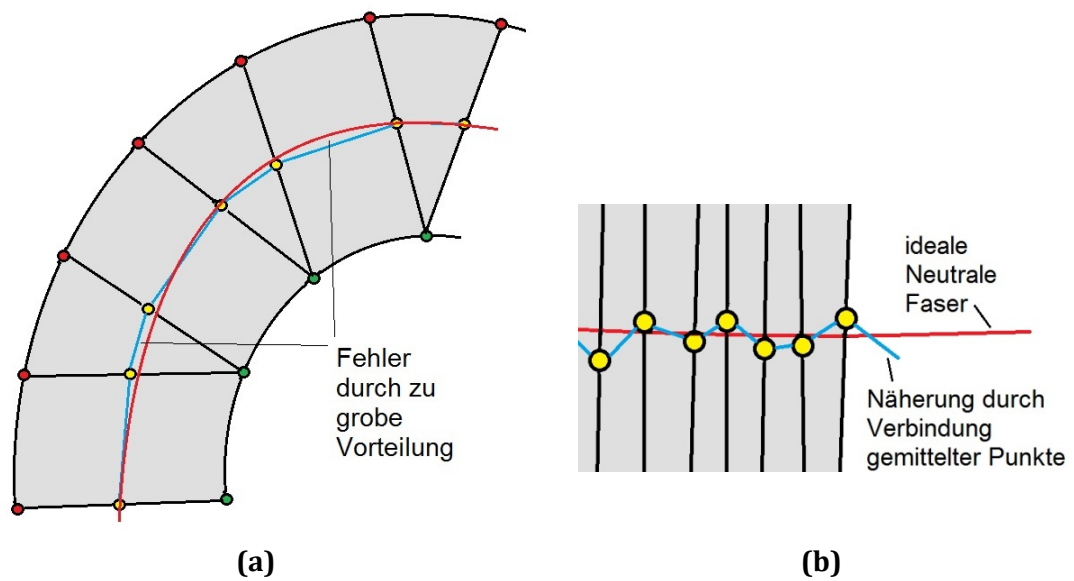
**Abbildung 3.28: Gruppieren der Vorteilungspunkte**

Nun werden die Koordinaten der Keypointgruppen gemittelt und an der Position der mittleren Koordinaten ein neuer Keypoint generiert. Die Verbindungslinien dieser Keypoints stellt nun näherungsweise die neutrale Faser der Geometrie dar. Die Abbildung 3.29 zeigt diesen Vorgang schematisch.



**Abbildung 3.29: Erzeugen einer neutralen Faser**

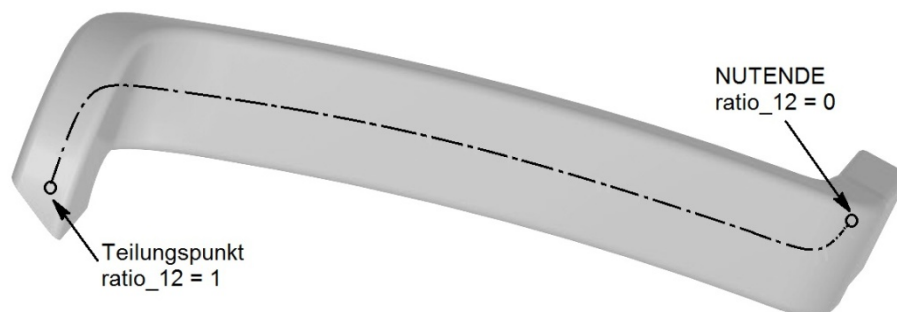
Der Parameter „Vorteilung“ bestimmt die Genauigkeit der neutralen Faser. Wird diese zu grob gewählt, so stimmt die Näherung mit geraden Linienzügen nur bedingt. Aber auch eine zu fein gewählte Vorteilung kann zu Problemen führen, da dann die einzelnen Keypoints so nahe beieinander liegen, dass schon durch kleine Ungenauigkeiten die lokale Krümmung der neutralen Faser übermäßig ansteigen kann. Die (a) zeigt die Fehler bei zu grober Vorteilung, die Abbildung 3.30 (b) die Fehler bei zu feiner Vorteilung.



**Abbildung 3.30: Fehler durch ungünstig gewählte Verteilung:**  
**a) zu grobe Verteilung, b) zu feine Verteilung**

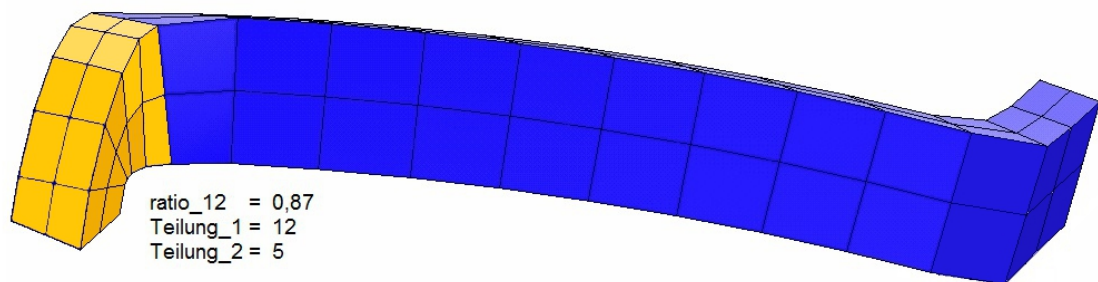
Die obigen Abbildungen zeigen nun deutlich, weshalb eine automatische Krümmungserkennung nur schwer möglich ist. Zu Beginn des Projektes wäre eine krümmungsabhängige Elementteilung angedacht worden. Dies würde aber, abgesehen vom oben genannten Grund, die Aufteilung beider Spulenteile in gleich viele Elemente der Länge nach wesentlich erschweren. Eine gleiche Elementanzahl der Länge nach ist auch bei den Stabwicklungen verwendet worden und stellt sicher, dass die weiteren Unterprogramme für die Anbauteile problemlos funktionieren.

Nachdem nun die neutrale Faser erzeugt ist, werden alle Hilfsentitäten gelöscht, sodass nur 8 Konturlinien und eine durchgehende neutrale Faser übrig bleiben.



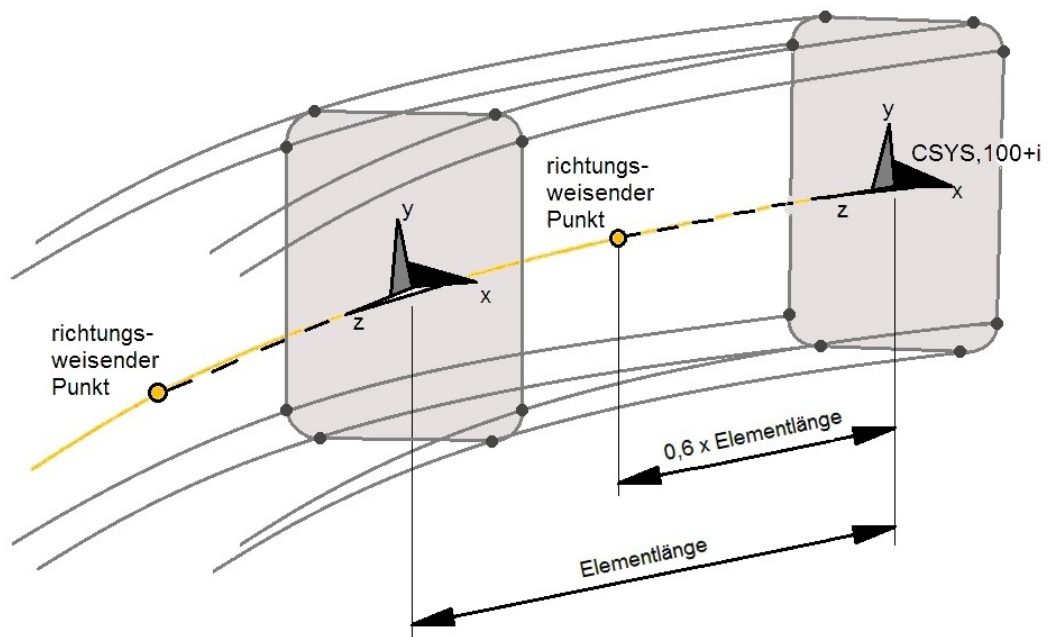
**Abbildung 3.31: Definition von Parameter „ratio\_12“ am Unterspulenteil**

Abbildung 3.31 zeigt die Definition des Parameters „ratio\_12“ welcher die 2 Abschnitte im Spulenteil trennt. Der Parameter beschreibt die Position auf der neutralen Faser mit den Werten zwischen 0 und 1 und zwar so, dass „ratio12 = 0“ die Position des Keypoints am Nutende und „ratio12 = 1“ den Keypoint der Neutralen Faser an der Teilungsebene beschreibt. Alle Positionen dazwischen können also mit einem Wert zwischen 0 und 1 beschrieben werden. Der Parameter hat den Zweck, die Teilung von Position „0“ bis Position „ratio\_12“ in „Teilung\_1“ und folglich den Rest zwischen Position „ratio\_12“ und Position „1“ in „Teilung\_2“ spätere Elemente zu unterteilen. Die Summe aus „Teilung\_1“ und „Teilung\_2“ ist mit „Teilung“ benannt. Abbildung 3.32 nimmt schon vorweg, wie das Spulenteil vernetzt aussehen wird. Es soll zeigen welchen Zweck die eben vorgenommenen Einstellungen haben werden. Der blaue Abschnitt ist mit „Teilung\_1“ vernetzt, der gelbe Abschnitt mit „Teilung\_2“, der Übergang zwischen den Abschnitten liegt im vorliegenden Fall bei 87% der Gesamtlänge des Spulenteils.



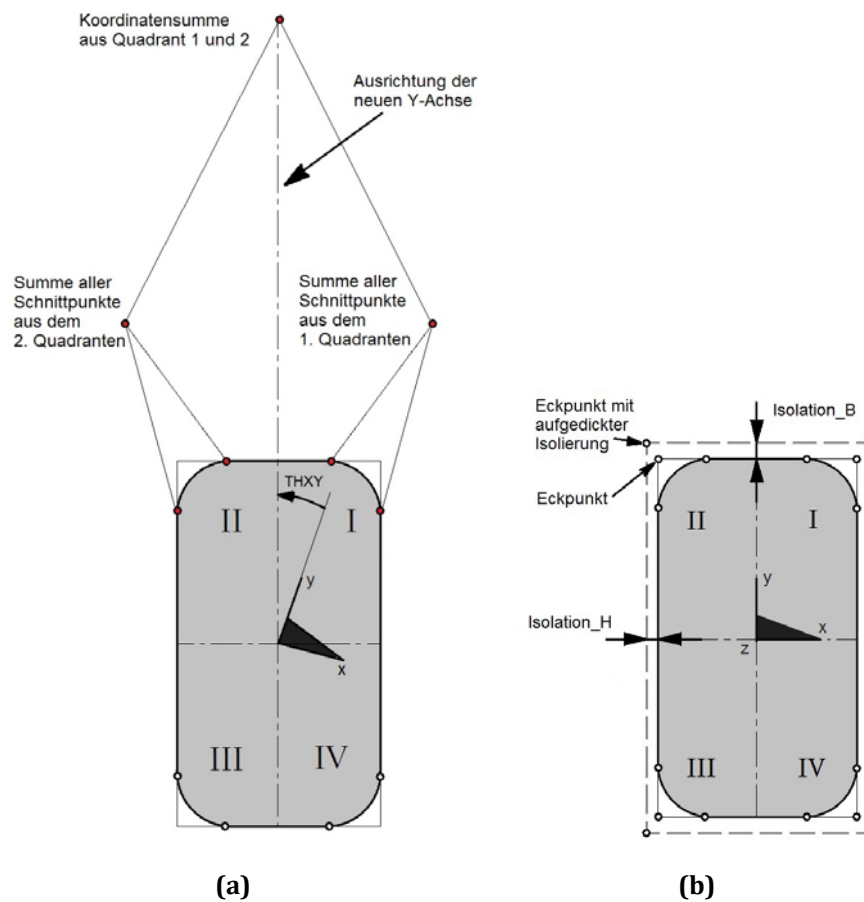
**Abbildung 3.32: Unterspulenteil in 2 Abschnitten vernetzt**

Ist nun die neutrale Faser in „Teilung“ Teile aufgeteilt, so wird an jedem Punkt ein Koordinatensystem erzeugt, dessen Z-Achse auf einen weiteren Punkt auf der neutralen Faser im Abstand von 0,6 Elementlängen zeigt. Der Wert von 0,6 Elementlängen hat sich rein empirisch als zweckmäßig erwiesen. Abbildung 3.33 zeigt wie die Z-Achsen der lokalen Koordinatensysteme „CSYS,100+i“ nach den richtungsweisenden Punkten ausgerichtet sind. Die X- und Y-Achsen sind derzeit noch nicht definiert. Mit den X-Y-Ebenen der lokalen Koordinatensysteme werden nun die Konturlinien aufgetrennt. Es entstehen somit bei jedem Koordinatensystem 8 Keypoints.



**Abbildung 3.33: Ausrichtung der lokalen Z-Achsen und Schnitt der Konturlinien**

Um jetzt die lokalen Koordinatensysteme ausrichten zu können, müssen zwei Kanten der Geometrie ausgewählt werden. Die Abbildung 3.34 zeigt den Querschnitt der Rohgeometrie in grau und den späteren Querschnitt des fertig vernetzten Modells als dünne Kontur. Die Schnittpunktkoordinaten der ersten beiden Quadranten (I und II) werden summiert und daraus der Drehwinkel THXY ermittelt.

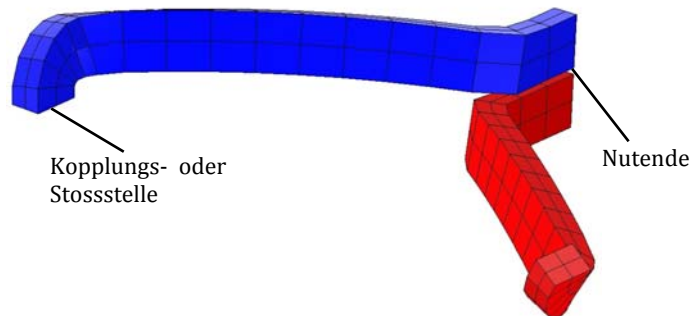


**Abbildung 3.34: Drehen der lokalen Koordinatensysteme:  
a) Bildung der Koordinatensumme, b) Verdrehtes  
Koordinatensystem mit optional aufgedickter Isolierung**

Ist das lokale Koordinatensystem nun auf die Symmetrieachsen des Querschnitts ausgerichtet, dann werden in jedem Quadranten die maximale beziehungsweise minimalen X- sowie Y-Koordinaten herausgesucht und damit der Eckpunkt im jeweiligen Quadranten erzeugt. Abbildung 3.34 veranschaulicht dies. Falls nötig kann noch eine Isolierschichtdicke (Isolation\_B in X-Koordinatenrichtung und Isolation\_H in Y-Koordinatenrichtung) angegeben werden.

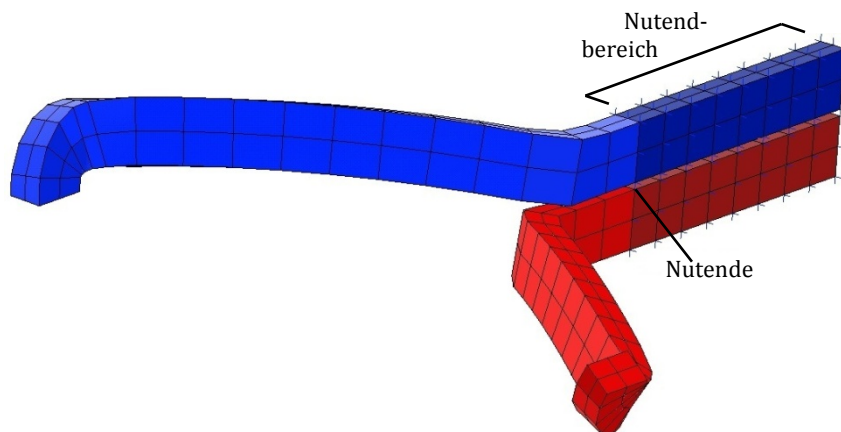
Nachdem die Eckpunkte (Keypoints) erzeugt sind, werden diese um den Winkel „slotangle“ um die CSYS,12 Z-Achse verdreht, an dessen Stelle Knoten (NODE) erzeugt und mittig zwischen den Eckpunkten weitere Zwischenknoten erzeugt, sodass an jedem Eckpunkt der danach erzeugten Elemente ein Knoten mit definierter Nummer (siehe Kapitel 3.3) steht. In dieses Knotengerüst werden daraufhin die Elemente eingehängt. Das Endergebnis sieht dann wie in Abbildung 3.35 aus. Die

Spulengeometrie gleicht somit im Aufbau und der Knotennummerierung, wie gefordert den Stabgeometrien der vorigen Projekte und kann auch wie eine solche in der bestehenden Programmstruktur verarbeitet werden. Einzig das Unterprogramm für die Kupferverbindungsblaschen mit denen Ober- und Unterstab verlötet werden, muss durch das Unterprogramm „Verbindung\_Stosstelle“ ersetzt werden.



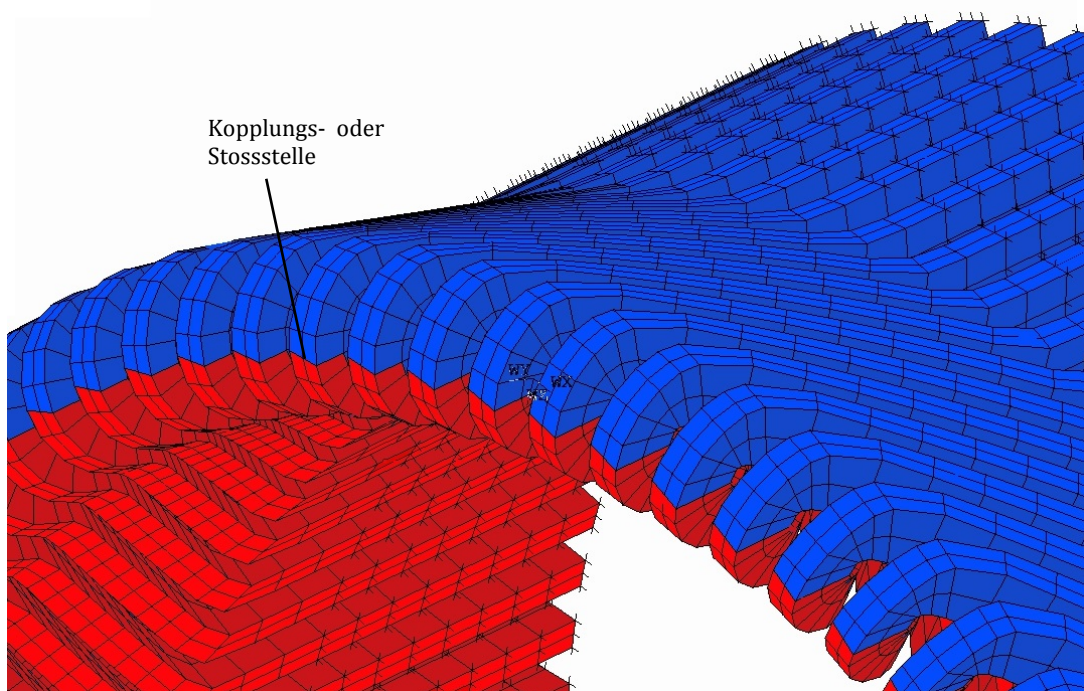
**Abbildung 3.35: Beide Spulenteile fertig vernetzt**

Nun muss noch der Nutendbereich mit der elastischen Bettung, wie in Abbildung 3.36 ersichtlich und in Kapitel 3.4 näher beschrieben, modelliert werden.



**Abbildung 3.36: Vernetzte Spulenteile mit modelliertem Nutendbereich**

Jetzt kann diese Baugruppe aus Abbildung 3.36 rotationssymmetrisch um die CSYS,12 Z-Achse gemustert werden. Das Ergebnis sieht wie in Abbildung 3.37 aus. Die zuvor getrennten Spulenteile müssen nun an der Teilungsstelle wieder zusammengefügt werden. Dazu werden die Knoten an der Teilungsstelle der Oberspule nacheinander ausgewählt werden und jeweils in einem „Toleranzwürfel“ rundherum weitere Knoten gesucht werden. Der Ausgangsknoten an der Oberspule und der gefundene Knoten an der Unterspule werden zu einem Knoten verschmolzen, wobei die Nummer des neuen Knotens die kleinere der beiden verschmolzenen Knoten, also die des Oberspulenknotens ist.



**Abbildung 3.37: Kopplung der gemusterten Spulenteile am Spulenauge**

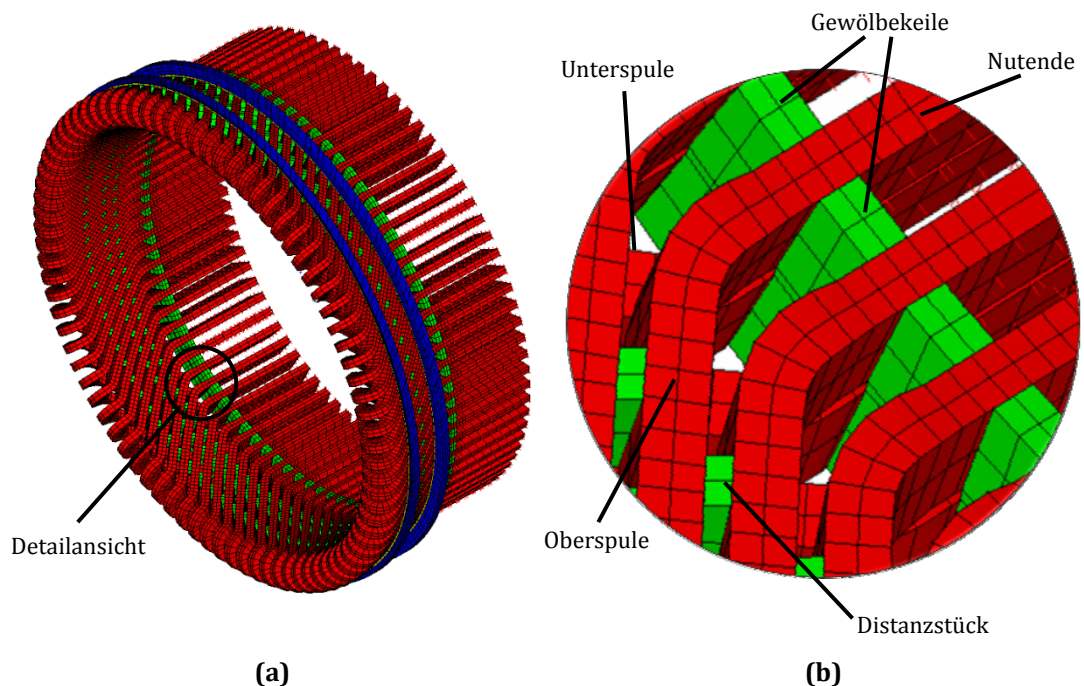
Die weitere Vorgehensweise beinhaltet das Modellieren der Anbauteile wie Stützringe, Distanzstücke und Gewölbekeile. Der verhältnismäßig große Aufwand in der Geometriaufbereitung bis hier, durch das Auftrennen der Spule in zwei Teile und die penible Einhaltung der gleichen Vernetzungsvorschriften wie beim Stabmodell, hat den gewünschten Effekt, dass die beiden Unterprogramme für die Stützringe und Distanzstücke übernommen werden können. Ein Unterprogramm für die Gewölbekeile existierte aber nicht und musste deshalb neu programmiert werden.



### 3.6.2 Gewölbekeile

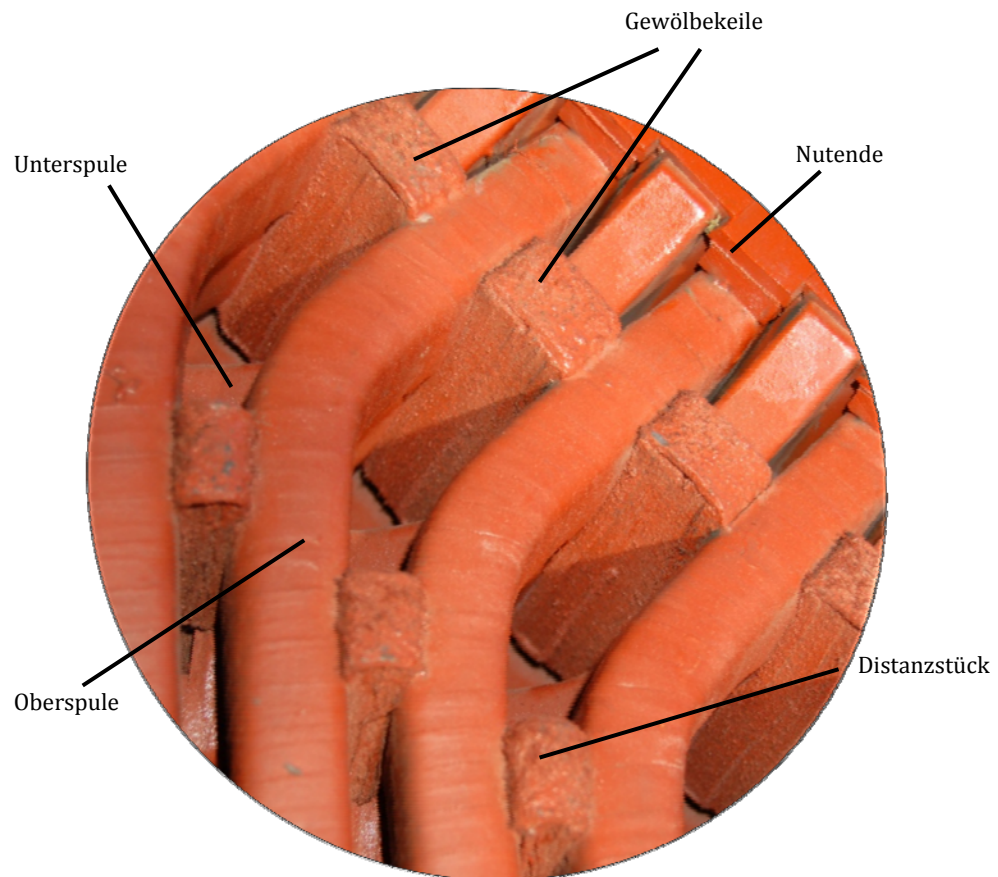
Das Unterprogramm für die Anbauteile „Gewölbekeile“ wurde im Rahmen dieser Arbeit zur Gänze neu programmiert. Das folgende Kapitel beschreibt die Modellierung und Anbindung der Gewölbekeile an den Wickelkopf.

Die Gewölbekeile liegen in kurzem Abstand zum Nutende jeweils zwischen zwei, aus benachbarten Statornuten herauslaufenden, Leitern. Sie reichen, anders als Distanzstücke, nicht nur über eine Leiterschicht (Ober- oder Unterschicht) sondern gehen über die gesamte radiale Ausdehnung des Wickelkopfs. Das Material sind beschichtete, epoxidharzgetränkte Glasfasermatten, welche nach dem Einbau ausgehärtet werden. Die Abbildung 3.38 stellt schematisch die Position der Gewölbekeile im Wickelkopf dar. Darstellung (a) bildet den gesamten Wickelkopf ab, der Inhalt des eingezeichneten Kreises ist daneben als Detailansicht (b) vergrößert dargestellt.



**Abbildung 3.38: Positionsskizze der Gewölbekeile:**  
 a) Gesamtansicht des Wickelkopfs, b) Detailansicht mit Gewölbekeilen

In der Realität sehen die Gewölbekeile wie in Abbildung 3.39 aus. Der Blickwinkel und der Bildausschnitt entsprechen dem von Abbildung 3.38 (b).



**Abbildung 3.39: Detailbild der Gewölbekeile [6]**

Die Knotennummerierung der Gewölbekeile erfolgt nach einem festgelegten Plan. Die verwendete Hunderterstelle in der Knotennummerierung ist mit „9“ vorgesehen. Sie kann aber durch Verändern der Variable „NR“ am Anfang des Programms leicht angepasst werden, sollte dies von Nöten sein.

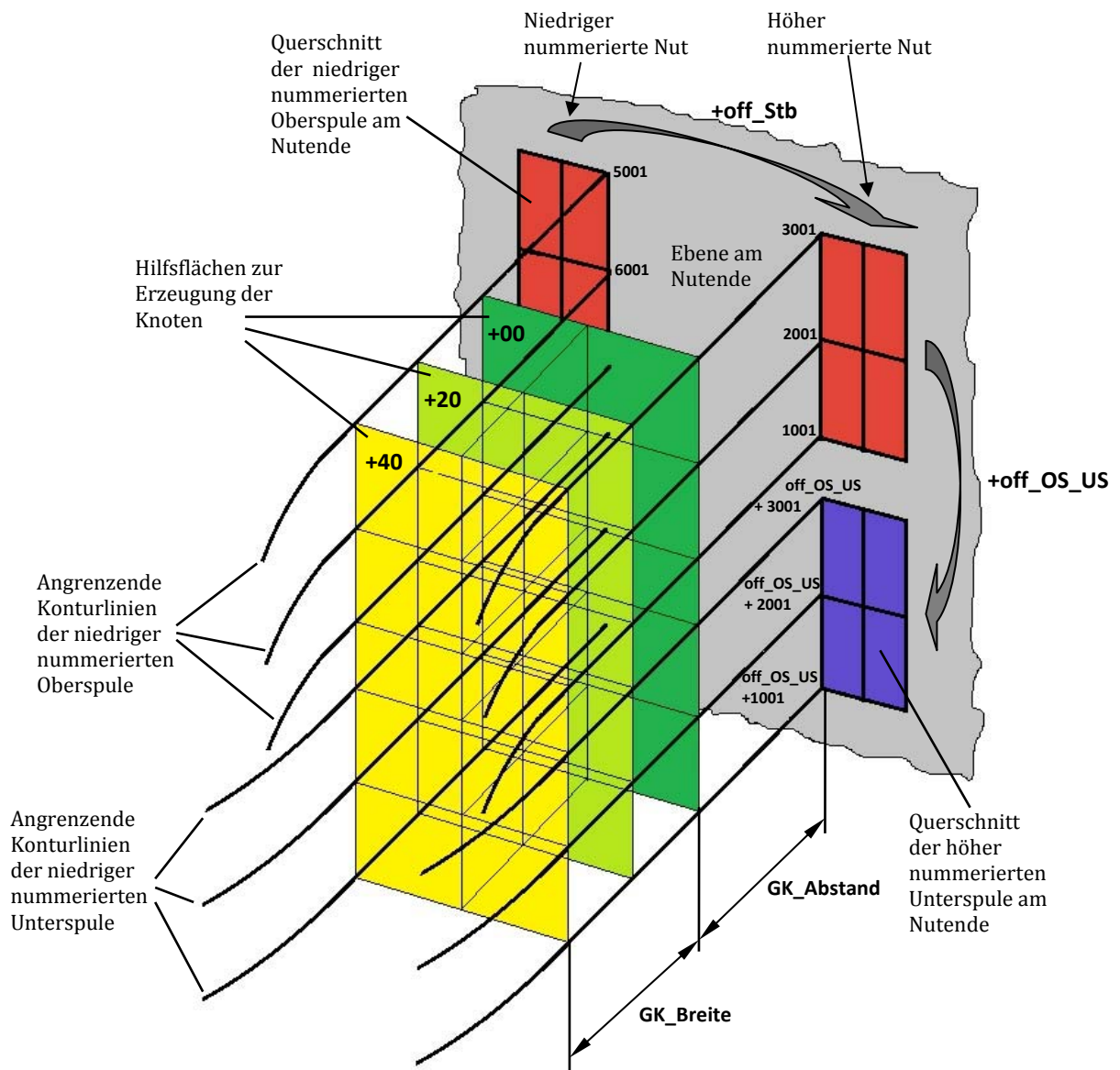


Abbildung 3.40: 3D-wireframe-Skizze der Gewölbekeile

Die Abbildung 3.40 stellt die Hilfsgeometrien rund um die Modellierung der Gewölbekeile dar. Die Darstellung der Spulengeometrien ist auf ihre angrenzenden Konturlinien beschränkt. Die grau eingezeichnete Ebene stellt das Nutende dar und die rot und blau eingefärbten Felder in dieser Ebene, die Spulengeometrie mit ihren 2x2 Elementen im Querschnitt. Die Knoten am Nutende der Konturlinien sind mit ihren Nummern versehen. Zu beachten dabei ist, dass die rechte Nut um den Wert „off\_Stb“ höher nummeriert ist als die linke, genauso wie die Unterspulenknoten um „off\_OS\_US“ höher nummeriert sind als die Oberspulenknoten. Die drei Hilfsflächen in grün, hellgrün und gelb stellen die Lage der Working Planes dar, welche zur Erzeugung der

Gewölbekeilknoten benötigt werden. Die Zahl in der linken oberen Ecke der Hilfsflächen hat Relevanz bei der Nummerierung der Knoten. Rechts unten im Bild sind die Maße „GK\_Abstand“ und „GK-Breite“ eingezeichnet, welche in der „parameter“-Datei eingegeben werden müssen.

Stellt man nun eine der drei Hilfsflächen, wie in Abbildung 3.41 separat dar, so kann man die Schnittpunkte der eingezeichneten Hilfslinien mit Nummern von 1-18 versehen, wobei die Knoten 13-18 zur niedriger nummerierten Nut gezählt werden, die Knoten 1-12 zur höher nummerierten Nut, also einen um „off\_Stb“ höheren Wert haben. Die Nummerierung der Knoten ist in allen drei Hilfsflächen gleich aufgebaut, nur dass die, links oben in der Fläche eingetragene, Zahl zur Knotennummer addiert wird. In diesem Fall ist die am weitesten vom Nutende entfernte Hilfsfläche dargestellt. Zu den eingetragenen Nummern ist daher die Zahl 40 addiert.

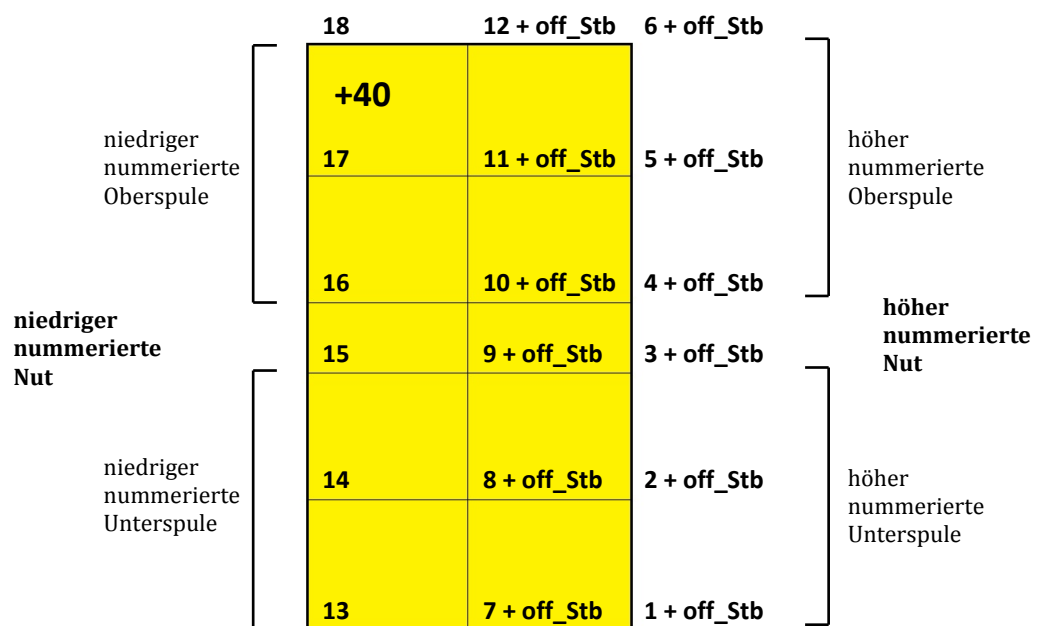
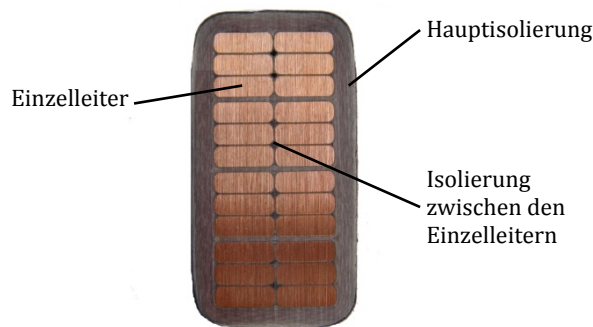


Abbildung 3.41: Hilfsfläche zur Modellierung der Gewölbekeilknoten

### 3.7 Vereinfachungen

Alle nachfolgend beschriebenen Einschränkungen und Ursachen für Abweichungen des Modells vom realen Aufbau ergaben sich aus der Notwendigkeit, die komplexe Struktur des Wickelkopfs entsprechend zu vereinfachen. Diese Vereinfachungen und deren Auswirkungen müssen aber bei der Auslegung der Ergebnisse, sowie bei Veränderungen am Modell berücksichtigt werden um Fehlinterpretationen zu vermeiden.

- Der Aufbau aus Abbildung 3.42 mit den unterschiedlichen Materialeigenschaften von Kupferleiter und Isolierschichten wird im Versuch getestet und daraus auf die Materialeigenschaften eines fiktiven, homogenen Materials mit demselben Verhalten geschlossen, welche dann zur Simulation verwendet werden.



**Abbildung 3.42: Aufbau des Leiterquerschnitts [6]**

- Die Isolierschichtdicken variieren je nach Krümmung und Ort, da die Isolierung erst nach der Formung der Spulen aufgewickelt wird. Der Dickensprung nach dem Nutende soll das veranschaulichen, die Realität sieht aber anders aus. Bei geraden Leiterteilen wie im Blechpaket wird weniger Isolierung benötigt als bei gekrümmten Teilen, da bei Krümmungen die Mindestdicke an der Außenseite der Krümmung eingehalten werden muss und an der Innenseite somit mehr Isolierung anfällt.

- Bei der Fertigung werden die Spulen über Schablonen geformt. Es ist somit nicht sichergestellt, dass die Spule im CAD Parasolid-File exakt der realen Geometrie entspricht.
- Die Kontaktbedingungen in ANSYS® Mechanical APDL können nur für Zug- und Druckbelastung gleichermaßen definiert werden. In der Realität können aber manche Kontakte zwischen den Leitern und den Anbauteilen auf Zug keine wesentlichen Lasten aufnehmen. In der Simulation wirken die Kontakte wie „verschweißt“, in Realität würden sich bei Zugbelastung, noch vor Erreichen der Belastungsgrenzwerte, Spalte bilden.
- Die Problematik aus dem vorigen Punkt führt in Realität zu Nichtlinearitäten im Schwingungsverhalten. Mit ANSYS® Mechanical APDL kann man aber kein nichtlineares Schwingungsverhalten simulieren.
- Die Teilung der Leitergeometrie hat wesentlichen Einfluss auf die Eigenfrequenz des Wickelkopfs. Die Knoten an den Anbauteilen sowie an der Leitergeometrie werden in allen ihren Freiheitsgraden zueinander gesperrt, sodass bei zu großen Abständen zwischen ebendiesen Knoten eine unnatürliche Versteifung durch die Kopplung auftritt. Sinnvollerweise muss die Teilung in den, mit den Anbauteilen verbundenen, Geometrieteilen möglichst fein gewählt werden, damit diese unnatürlichen Versteifungen minimal werden.

### 3.8 Ergebnisse der rechnerischen Modalanalyse

Die Ergebnisse der rechnerischen Modalanalyse sind die Eigenfrequenzen und die dazugehörigen Eigenformen des Wickelkopfs. In den nachfolgenden Abbildungen sind die ersten drei Eigenformen dargestellt. Die roten Bereiche sind jene, die am weitesten von der Ruheposition abweichen. Die Simulation wurde mit folgenden Eingangsparametern, zu finden in der „parameter“-Datei des entsprechenden Projektordners, durchgeführt. In Tabelle 3.2 sind nur die relevantesten Parameter angeführt.

**Tabelle 3.2: Parameterbezeichnung**

Parameterbezeichnung	Wert	Bedeutung
Vorteilung_Eingabefile	<b>200</b>	Vorteilung der Konturlinien
ratio_12	<b>0,87</b>	Position des Übergangs von Teilung_1 zu Teilung_2
Teilung_1	<b>35</b>	Teilung am ersten Stababschnitt
Teilung_2	<b>5</b>	Teilung am zweiten Stababschnitt
e_laenge	<b>0,03</b>	Elementlänge im Nutendbereich
zus_Elemente_OS	<b>8</b>	Anzahl der Elemente im Nutendbereich für den OS
zus_Elemente_US	<b>8</b>	Anzahl der Elemente im Nutendbereich für den US
Federlaenge	<b>0.005</b>	Länge der Bettungsfedern im Nutendbereich
Federsteifigkeit_tan_OS	<b>2,7 E6</b>	Federsteifigkeit in tangentialer Richtung am Oberstab
Federsteifigkeit_rad_OS	<b>2,6 E7</b>	Federsteifigkeit in radialer Richtung am Oberstab
Federsteifigkeit_tan_US	<b>2,7 E6</b>	Federsteifigkeit in tangentialer Richtung am Unterstab
Federsteifigkeit_rad_US	<b>2,6 E7</b>	Federsteifigkeit in radialer Richtung am Unterstab

Die noch relativ großen Abweichungen der Simulationsergebnisse von den Messergebnissen sind auf das Fehlen der richtigen Materialparameter, der einzelnen Bauteile zurückzuführen. Die hier verwendeten Materialparameter sind von einer ähnlichen Maschine übernommen. Es wird in weiterer Folge nötig sein, Studien zu den Materialeigenschaften durchzuführen, die passenden Werte zu ermitteln und die Simulation unter Verwendung der neuen Materialeigenschaften erneut auszuführen.

Trotzdem sind die Simulationsergebnisse, mit den bisherigen Materialdaten, in den folgenden Unterkapiteln beschrieben und mit den Messergebnissen verglichen.

### 3.8.1 Die 1. Eigenfrequenz (Nickmode)

Die Abbildung 3.43 beschreibt die, zur ersten Eigenfrequenz zugehörigen, Eigenform. Es bewegt sich der Ring, der von den Spulenaugen beschrieben wird, gegenüber dem Nutende linear in radialer Richtung, ohne wesentlich von der Kreisform abzuweichen. Die Pfeile in den Darstellungen sollen die Bewegung veranschaulichen. Die rot eingefärbten Stellen sind jene mit den maximalen, die blauen Bereiche die mit den geringsten Verformungen.

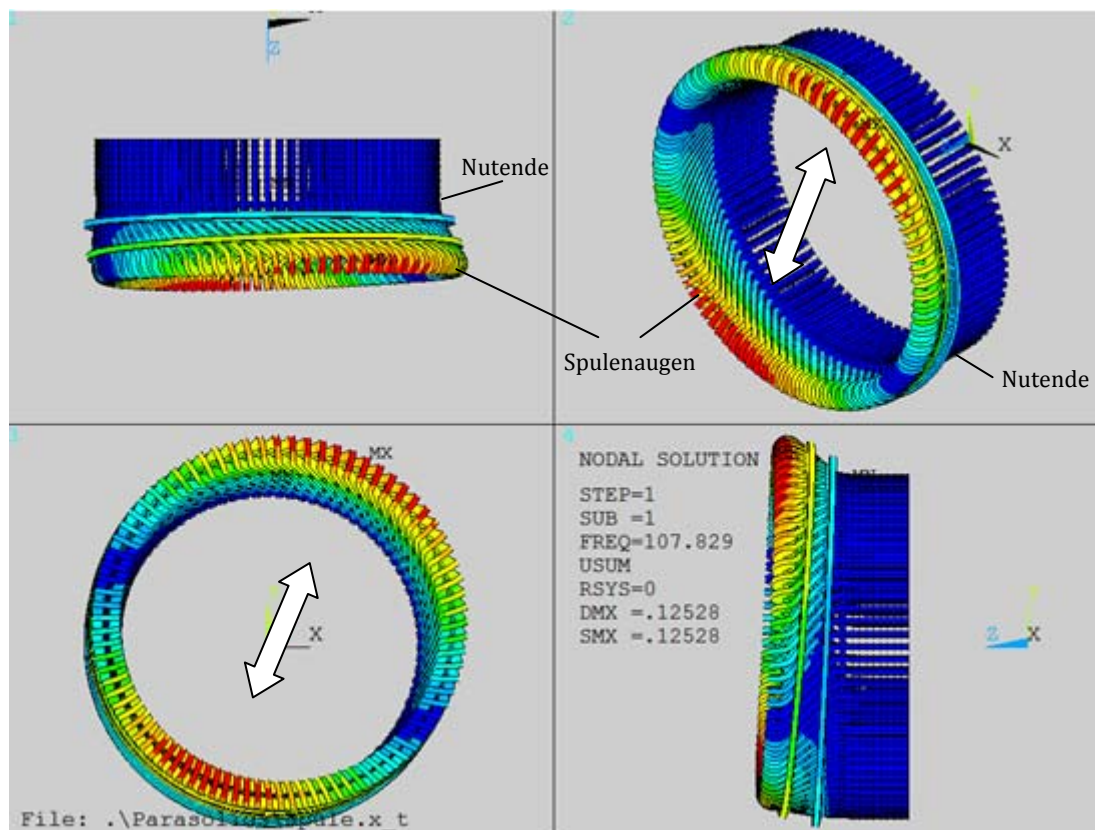


Abbildung 3.43: Eigenform der 1. Eigenfrequenz

Der Vergleich zwischen Simulationsergebnis und Messergebnis, sowie die prozentuelle Abweichung, ist in Tabelle 3.3 zusammengefasst:

Tabelle 3.3: Vergleich 1. Eigenfrequenz

Simulationsergebnis	Messergebnis	Abweichung des Simulationsergebnisses in % vom Messergebnis
<b>107,83 Hz</b>	<b>166 Hz</b>	<b>-35,0 %</b>



### 3.8.2 Die 2. Eigenfrequenz (4-Knoten Radialmode)

Die, der zweiten Eigenfrequenz zuordenbare, Eigenform ist in Abbildung 3.44 dargestellt. Der Ring der Spulenaugen verformt sich dabei elliptisch. Die Bewegung ist durch die Pfeile angedeutet. Jeweils zwei gegenüberliegende rot eingefärbte Bereiche bewegen sich gegengleich radial. Die 4 blau gefärbten Stellen beschreiben die Knoten des Modes.

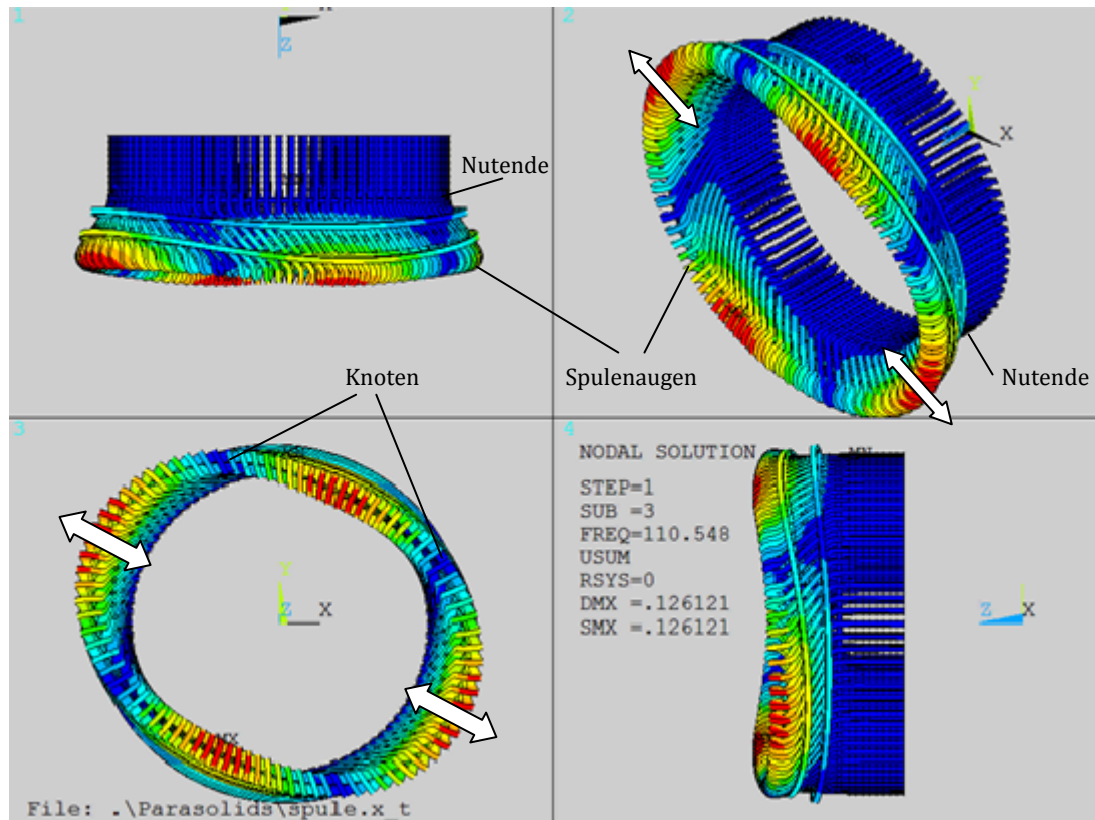


Abbildung 3.44: Eigenform der 2. Eigenfrequenz

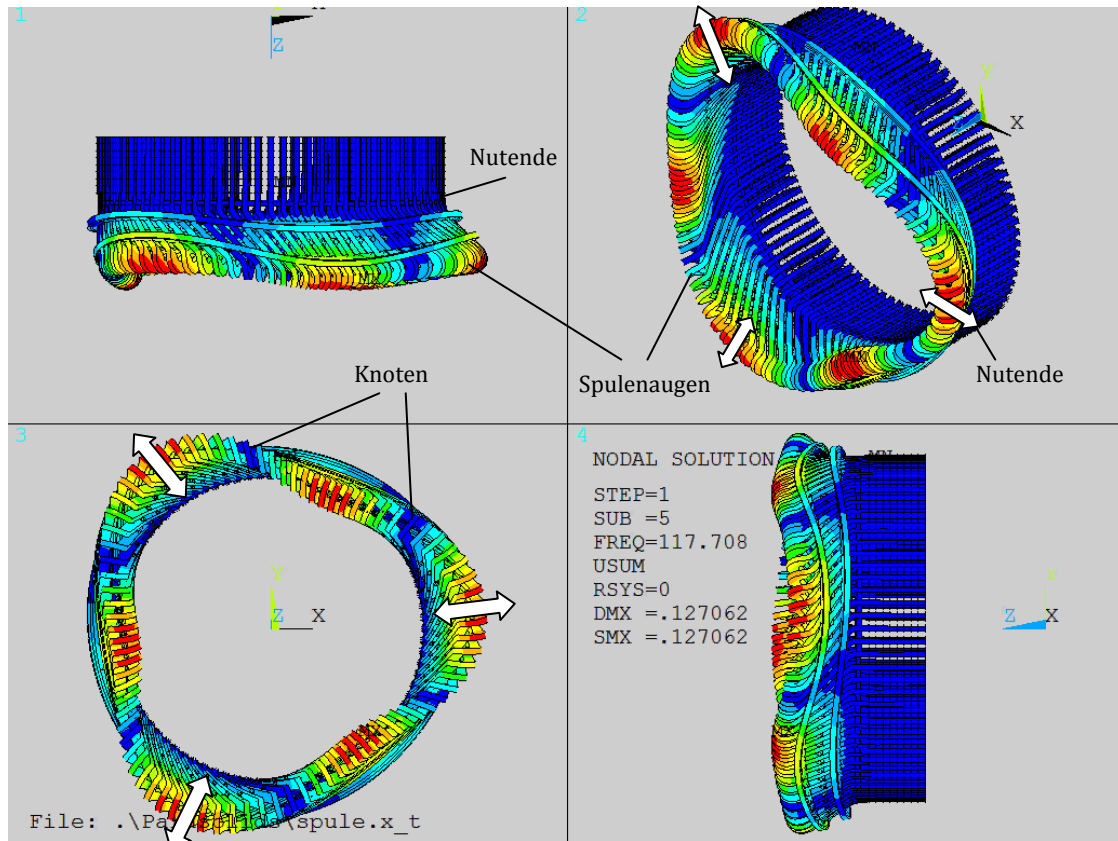
Der Vergleich zwischen Simulationsergebnis und Messergebnis, sowie die prozentuelle Abweichung, ist in Tabelle 3.4 zusammengefasst:

Tabelle 3.4: Vergleich 2. Eigenfrequenz

Simulationsergebnis	Messergebnis	Abweichung des Simulationsergebnisses in % vom Messergebnis
<b>110,55 Hz</b>	<b>177 Hz</b>	<b>-37,5 %</b>

### 3.8.3 Die 3. Eigenfrequenz (6-Knoten Radialmode)

Die Eigenform des 6-Knoten Radialmodes in Abbildung 3.45 ähnelt einem Dreieck. Jeweils 3 Abschnitte führen eine radiale Bewegung aus, während die dazwischenliegenden gegengleich mitschwingen. Zwischen den schwingenden Bereichen sind insgesamt 6 Knoten.



**Abbildung 3.45: Eigenform der 3. Eigenfrequenz**

Der Vergleich zwischen Simulationsergebnis und Messergebnis, sowie die prozentuelle Abweichung, ist in Tabelle 3.5 zusammengefasst:

**Tabelle 3.5: Vergleich 3. Eigenfrequenz**

Simulationsergebnis	Messergebnis	Abweichung des Simulationsergebnisses in % vom Messergebnis
<b>117,71 Hz</b>	<b>189 Hz</b>	<b>-37,7 %</b>

### 3.8.4 Die 4. Eigenfrequenz (8-Knoten Radialmode)

Die Eigenform des 8-Knoten Radialmodes in Abbildung 3.46 ähnelt einem Viereck. Wie beim 6-Knoten Radialmode schwingen die Hälfte der Bereiche zwischen den Knoten gegengleich zu den Bereichen dazwischen.

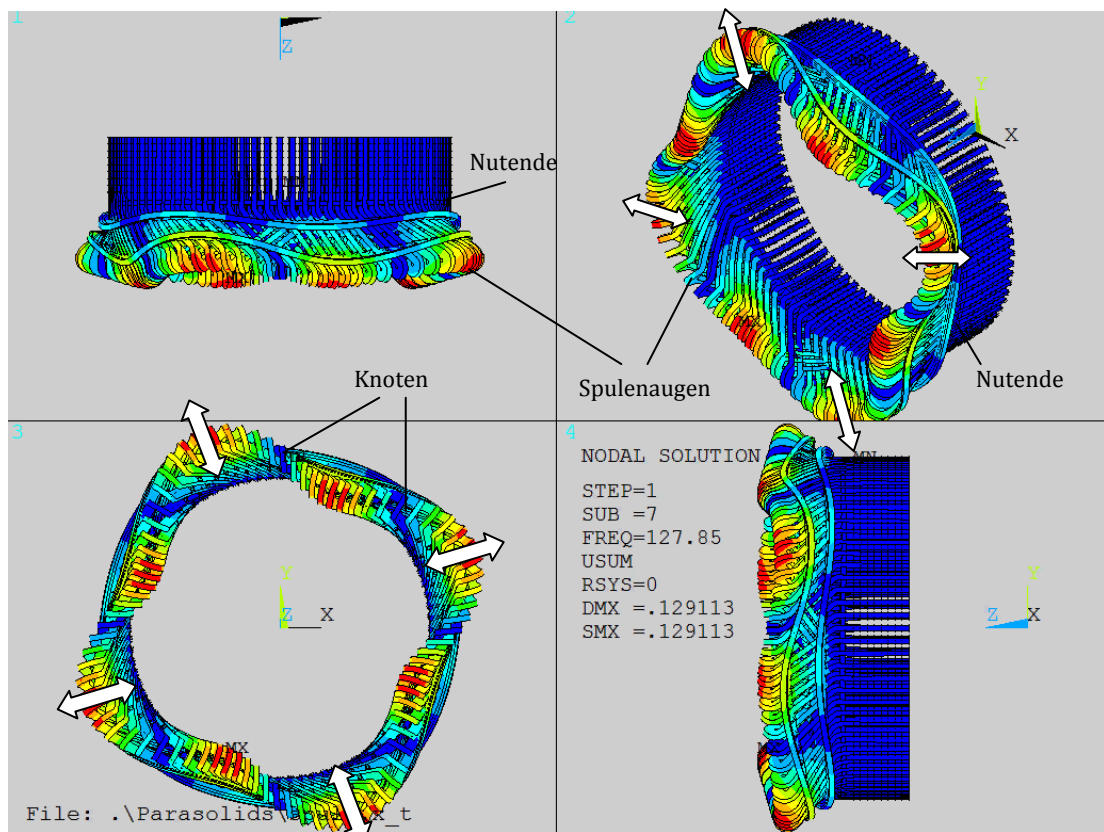


Abbildung 3.46: Eigenform der 4. Eigenfrequenz

Der Vergleich zwischen Simulationsergebnis und Messergebnis, sowie die prozentuelle Abweichung, ist in Tabelle 3.6 zusammengefasst:

Tabelle 3.6: Vergleich 4. Eigenfrequenz

Simulationsergebnis	Messergebnis	Abweichung des Simulationsergebnisses in % vom Messergebnis
<b>127,85 Hz</b>	<b>219 Hz</b>	<b>-41,6 %</b>

### 3.8.5 Die 5. und 7. Eigenfrequenz (nicht vergleichbare Modes)

Die Eigenform der 5. Eigenfrequenz aus Abbildung 3.47 ist gänzlich anders als die Eigenformen der 1.- 4. Eigenfrequenz. Hier weitet und verengt sich der gesamte Ring der Spulenaugen periodisch, ohne dabei von der Kreisform im Wesentlichen abzuweichen. Diese simulierte Eigenform besitzt keine Knoten. Sehr ähnlich sieht die Eigenform der 7. Eigenfrequenz (ohne Abb.) aus. Diese Modes besitzen ihre am stärksten ausgelenkten Bereiche im Umfeld der Stützringe.

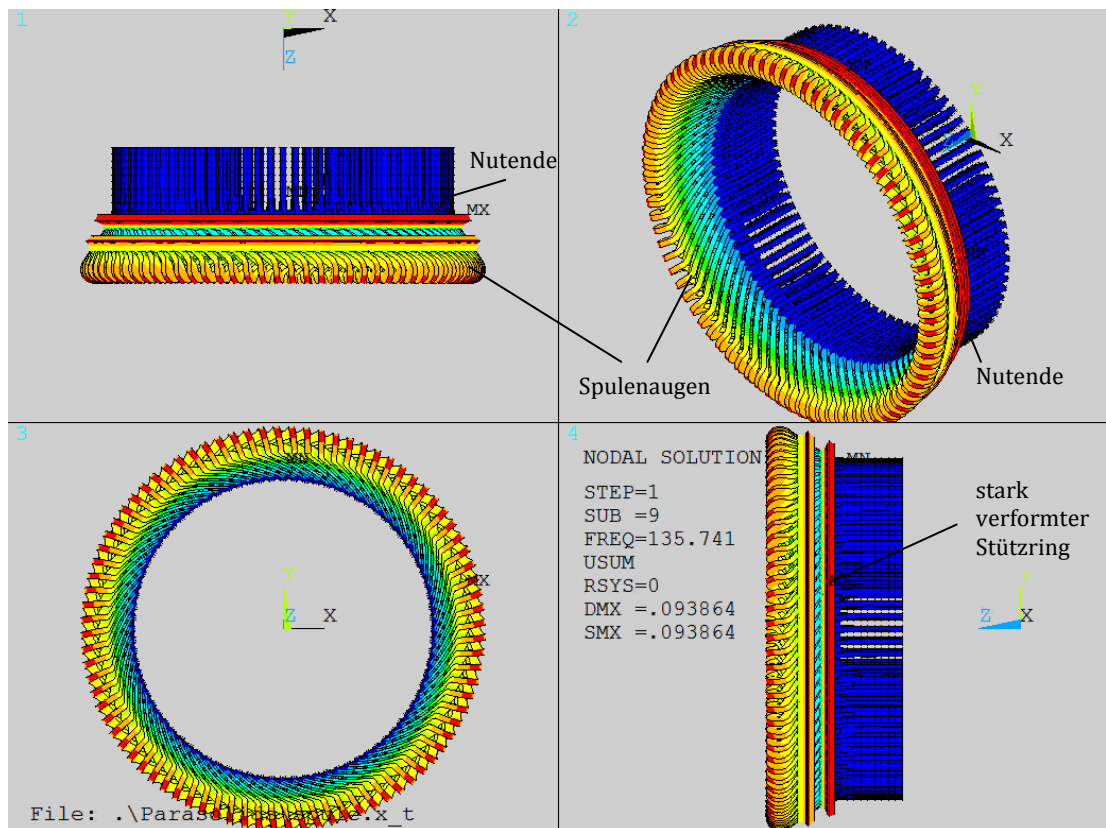


Abbildung 3.47: Eigenform der 5. Eigenfrequenz

Die Tabelle 3.7 zeigt das Simulationsergebnis der 5., Tabelle 3.8 das der 7. simulierten Eigenfrequenz für die es keine vergleichbaren, gemessenen Eigenformen in diesem Frequenzbereich gibt.

**Tabelle 3.7: Simulationsergebnis 5. Eigenfrequenz**

Simulationsergebnis	Messergebnis	Abweichung des Simulationsergebnisses in % vom Messergebnis
<b>135,74 Hz</b>	-	-

**Tabelle 3.8: Simulationsergebnis 7. Eigenfrequenz**

Simulationsergebnis	Messergebnis	Abweichung des Simulationsergebnisses in % vom Messergebnis
<b>144,52 Hz</b>	-	-

In den Visualisierungen der verfügbaren Messergebnisse konnte kein Bewegungsmuster gefunden werden, welches dieser Eigenform ähneln würde. Ein möglicher Grund wäre, dass diese Eigenform, durch die stark von der Realität abweichenden Materialparameter, verursacht wird. Die stark deformierten Stützringe und die maximalen Verformungen (DMX) in diesem Bereich deuten darauf hin, dass es sich um einen solchen Fall handelt. Das Auftreten solcher unrealistischer Eigenformen in der Simulation könnte somit als Hinweis auf nicht korrekte Materialparameter gedeutet werden. Ein Vergleich zwischen Simulation und Messung ist in diesem Falle nicht möglich.

Eine andere Erklärung dieser, nur in der Simulation auftretenden, Eigenformen wäre, dass diese Modes mit den Messaufnehmern nur unzureichend detektiert werden können und somit in den Ergebnissen der experimentellen Modalanalyse nicht aufscheinen.

### 3.8.6 Die 6. Eigenfrequenz (10-Knoten Radialmode)

Die Eigenform des 10-Knoten Radialmodes in Abbildung 3.48 gleicht einem Fünfeck. Wie schon beim 6- und 8-Knoten Radialmode schwingen die Hälfte der Bereiche zwischen den Knoten gegengleich zu den Bereichen dazwischen.

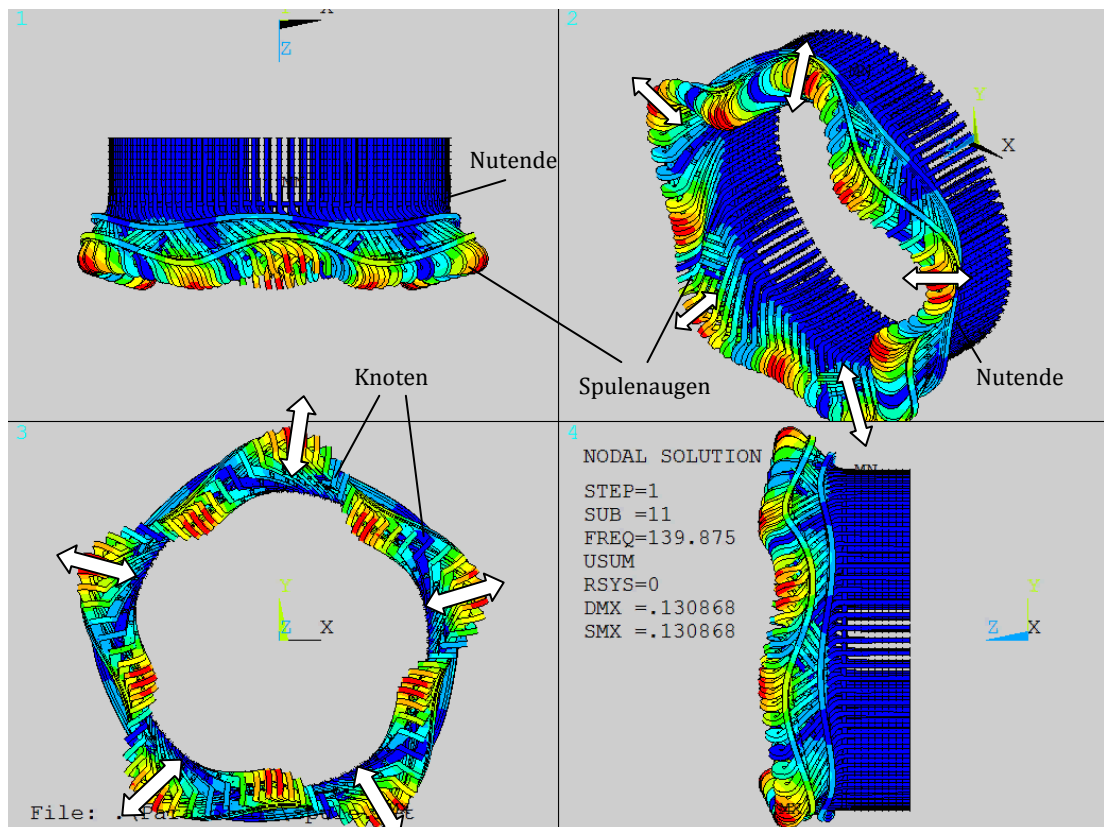


Abbildung 3.48: Eigenform der 6. Eigenfrequenz

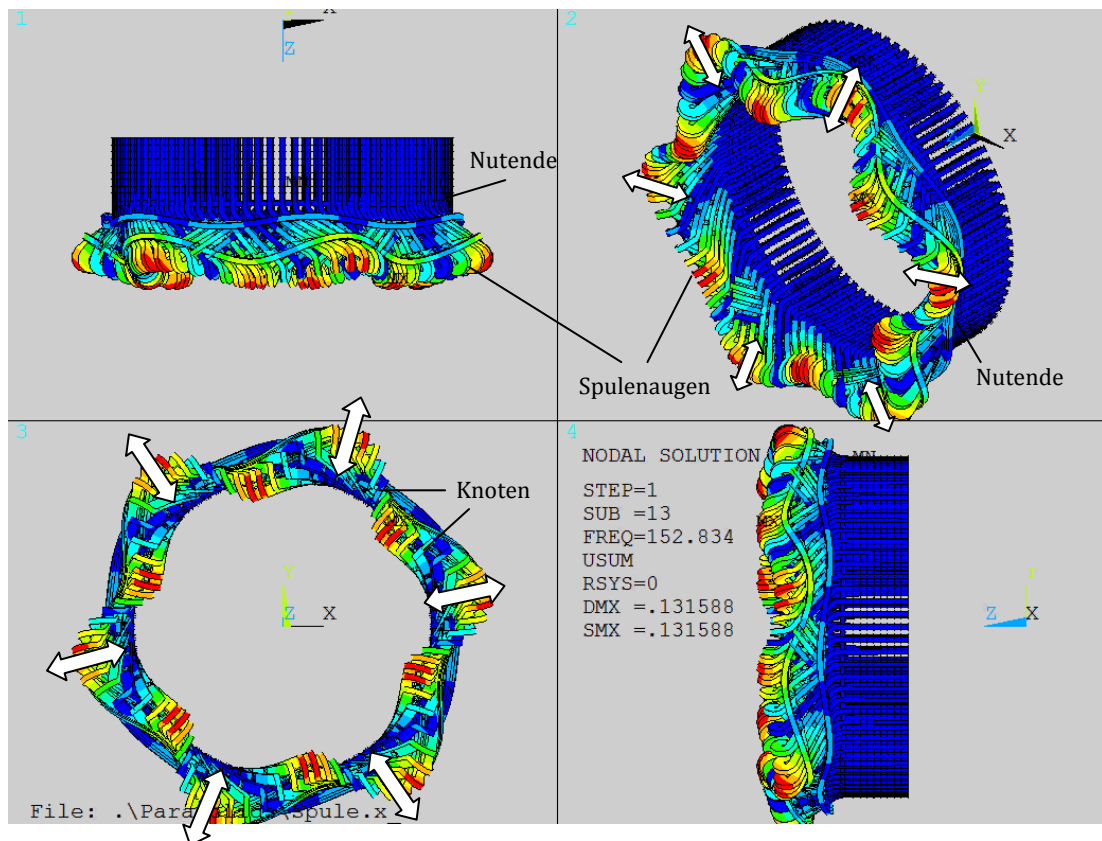
Der Vergleich zwischen Simulationsergebnis und Messergebnis, sowie die prozentuelle Abweichung, ist in Tabelle 3.9 zusammengefasst. ACHTUNG! Es muss der 6. Eigenfrequenz der Simulation, die 5. Eigenfrequenz der Messung gegenübergestellt werden:

Tabelle 3.9: Vergleich 6. Eigenfrequenz

Simulationsergebnis	Messergebnis	Abweichung des Simulationsergebnisses in % vom Messergebnis
<b>139,88 Hz</b>	<b>256 Hz</b>	<b>-45,4 %</b>

### 3.8.7 Die 8. Eigenfrequenz (12-Knoten Radialmode)

Die Eigenform des 12-Knoten Radialmodes in Abbildung 3.49 gleicht einem Sechseck. Wie bei allen anderen Radialmoden mit Knoten schwingen die Hälfte der Bereiche zwischen den Knoten gegengleich zu den Bereichen dazwischen.



**Abbildung 3.49: Eigenform der 8. Eigenfrequenz**

Der Vergleich zwischen Simulationsergebnis und Messergebnis, sowie die prozentuelle Abweichung, ist in Tabelle 3.10 zusammengefasst. ACHTUNG! Es muss der 8. Eigenfrequenz der Simulation, die 6. Eigenfrequenz der Messung gegenübergestellt werden:

**Tabelle 3.10: Vergleich 8. Eigenfrequenz**

Simulationsergebnis	Messergebnis	Abweichung des Simulationsergebnisses in % vom Messergebnis
<b>152,83 Hz</b>	<b>289 Hz</b>	<b>-47,1 %</b>

### 3.8.8 Zusammenfassung der Ergebnisse

Stellt man nun die Ergebnisse der Simulation mit jenen der experimentellen Modalanalyse gegenüber, so ist trotz der starken Abweichungen durch die falschen Materialparameter eine Systematik erkennbar.

Folgende Tabelle 3.11 zeigt alle Ergebnisse, sowohl der Simulation, als auch der Messung noch einmal gegenübergestellt:

**Tabelle 3.11: Gegenüberstellung der Ergebnisse**

Eigenfrequenz Nummer	Eigenfrequenz der Simulation	Eigenfrequenz Mittelwert der Messergebnisse	Abweichung des Simulationsergebnisses in % vom Messergebnis
1	107,83 Hz	166 Hz	-35,5 %
2	110,55 Hz	177 Hz	-37,0 %
3	117,71 Hz	189 Hz	-37,7 %
4	127,85 Hz	219 Hz	-41,6 %
5	135,74 Hz	-	-
6	139,88 Hz	256 Hz	-45,4 %
7	144,52 Hz	-	-
8	152,83 Hz	289 Hz	-47,1 %

Die Abweichungen liegen bei den untersuchten Eigenfrequenzen bei rund 35 bis 47 %, wobei ein stetiger Anstieg der Abweichung mit steigender Frequenz feststellbar ist.

#### 3.8.1 Parameterstudie

Um nun den Einfluss der vermuteten Hauptfehlerquellen, also die Materialeigenschaften der Layerschichten zwischen Stützringen und Spulen sowie der Distanzpakete, genauer abschätzen zu können, wurde eine Parameterstudie durchgeführt. Die beiden Parameter wurden dabei variiert und die möglichen Kombinationen aus den veränderten Parametern der Simulation zugeführt.

Fazit der Untersuchung:

Die Zwischenschichten zwischen Stützringen und Spulen haben einen weniger großen Einfluss auf die Frequenzen als die Steifigkeit der Distanzpakete. Zudem ändern die Modes (Nickmode, elliptischer Mode, etc.) schon ab rund 3-mal steifer simuliertem



Zwischenschichtmaterial ihre Reihenfolge. Vertauscht angeordnete Modes bedingen unregelmäßige Abweichungen der Ergebnisse und sind somit unrealistisch.

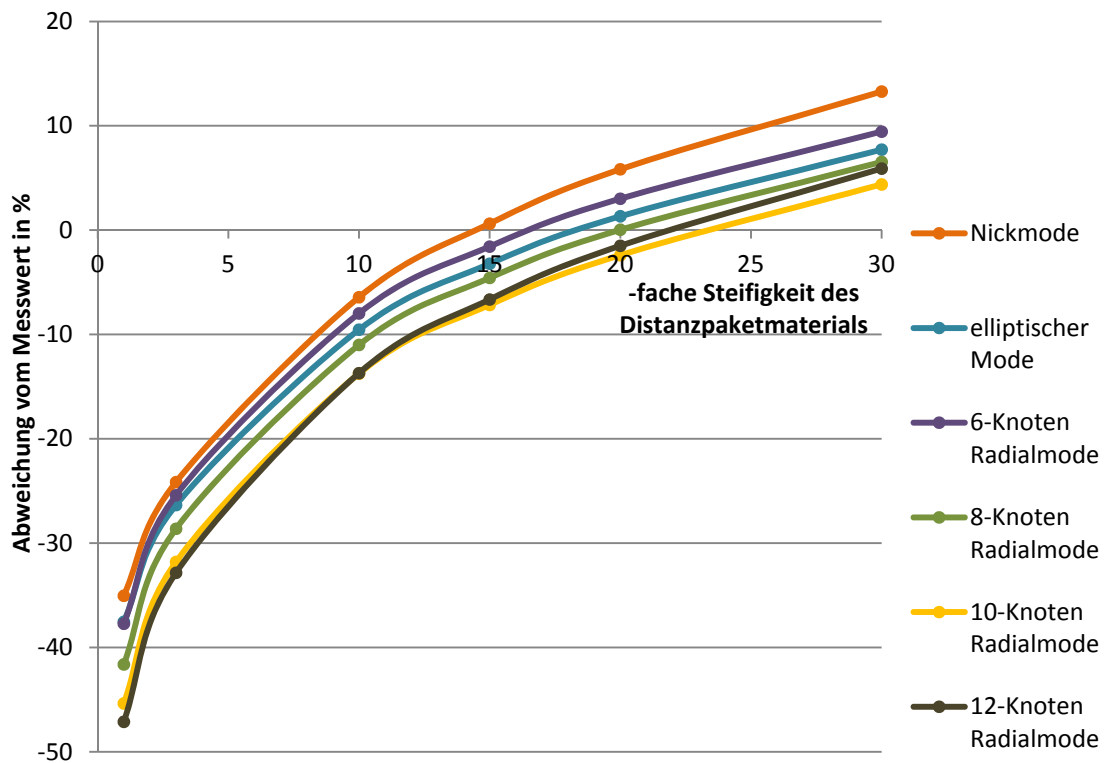


Abbildung 3.50: Auswertung der Parameterstudie

Das Diagramm in Abbildung 3.50 zeigt die Abhängigkeit der Frequenz von den Materialparametern der Distanzpakete. Die Zahlenwerte auf der horizontalen Achse geben an, um das Wievielfache die Elastizitäts- und Schubmoduln vergrößert wurden. Die vertikale Achse zeigt die Abweichung der simulierten Frequenzen von den gemessenen, in % der gemessenen Werte. Man erkennt, dass bei rund 15-20-facher Steifigkeit der Distanzpakete die Abweichungen der ersten 6 Schwingungsmodes minimal werden. Die anderen Materialparameter sind gleich belassen worden.

Der Einfluss der Federsteifigkeiten in der Einspannung (Statorblechpaket) wurde ebenfalls untersucht. Es ergab sich ein nur minimaler, kaum relevanter Einfluss auf das Schwingungsverhalten des Wickelkopfs.

Die detaillierte Auswertung der Parameterstudie ist im Anhang beigefügt.

## 4 Zusammenfassung

Aufgabe dieser Diplomarbeit war es, den Wickelkopf eines spulengewickelten Hydrogenerators mit Hilfe von ANSYS Mechanical APDL aufzubauen, zu vernetzen und eine rechnerische Modalanalyse durchzuführen. Die komplexere Geometrie von Spulenwicklungen erforderte eine spezielle Herangehensweise und stellte eine große Herausforderung hinsichtlich der Geometrieaufbereitung und Vernetzung dar.

Im Zuge der Arbeit wurde ein schon bestehendes Berechnungstool für Stabwicklungen zum Großteil durch neue Programmteile ersetzt, wobei das Augenmerk sowohl in der Einhaltung der zahlreichen Randbedingungen und Einschränkungen als auch bei der dabei größtmöglichen Variabilität sowie bestmöglichen Netzqualität lag.

Nach Abschluss der Modellierung und Vernetzung erfolgte die rechnerische Modalanalyse und der Vergleich mit den gemessenen Daten aus der experimentellen Modalanalyse. Eine zuletzt durchgeführte Parameterstudie soll Fehlerquellen und mögliche Lösungswege aufzeigen.

Wichtige Erkenntnisse dieser Arbeit waren unter anderem, wie weit das System vereinfacht werden muss, damit es mit vertretbarem Aufwand modellierbar und berechenbar wird aber im Gegenzug auch wie präzise der Netzaufbau sein muss, damit die Erfordernisse der verwendeten finiten Elemente sicher erfüllt werden.

Die Berechnungsergebnisse, also die berechneten Eigenfrequenzen weichen erwartungsgemäß noch stark von den Messergebnissen ab, da die genauen Materialparameter der verbauten Bauteile noch nicht bekannt sind. Es wurden die Materialeigenschaften einer ähnlichen Maschine übernommen und mittels Parameterstudie die vermuteten Fehlerquellen untersucht. Das Fazit der Untersuchung war, dass das Gesamtsystem in der Simulation noch zu weich ist und dies am wahrscheinlichsten am Distanzpaketmaterial liegen wird.

Die folgenden Schritte werden nun sein, die Materialeigenschaften soweit als möglich zu bestimmen, das System erneut zu berechnen, unter Berücksichtigung der, unter Kapitel 3.7 beschriebenen, Vereinfachungen die Ergebnisse erneut mit den Messergebnissen, als auch mit anderen vermessenen und berechneten Maschinen zu vergleichen und zu interpretieren.

## 5 Abbildungsverzeichnis

Abbildung 2.1: Aufbau einer spulengewickelten Drehstrommaschine [5] .....	3
Abbildung 2.2: Wickelkopf schematisch mit Blickwinkel der folgenden Abb.....	4
Abbildung 2.3: Wickelkopfaufbau eines spulengewickelten Generators [6].....	4
Abbildung 2.4: Austritt der Leiter aus dem Blechpaket am Nutende [6].....	5
Abbildung 2.5: Ausführungen von Synchronmaschinen [7]: .....	6
Abbildung 2.6: Läuferausführungen: .....	7
Abbildung 2.7: Schematischer Querschnitt durch einen Hydrogenerator [1]:.....	8
Abbildung 2.8: Geschlossene Spulenschleife .....	9
Abbildung 2.9: Die zwei separaten Stäbe einer Stabwicklung .....	9
Abbildung 2.10: Wickelkopf einer Spulenwicklung schematisch .....	10
Abbildung 2.11: Detailfoto der Spulenaugen [6].....	10
Abbildung 2.12: Verbindungslaschen an Stabwicklungen isoliert [6] .....	11
Abbildung 2.13: Verbindungslaschen an Stabwicklungen nicht isoliert [6].....	11
Abbildung 2.14: Aufbau eines Roebelstabes [10] .....	12
Abbildung 2.15: Spulenaufbau [10].....	12
Abbildung 2.16: Leiteranordnung in der Spule [6]:.....	13
Abbildung 2.17: Aufbau der Programmstruktur.....	15
Abbildung 2.18: Das Stabelement.....	16
Abbildung 2.19: Verschiebung und Kraftwirkung.....	17
Abbildung 2.20: Verknüpfung der Elemente und Matrizen.....	19
Abbildung 2.21: Skizze zum Gedankenmodell für die Massenmatrix .....	20
Abbildung 2.22: Determinantenwert über der Frequenz.....	22
Abbildung 2.23: Schema der experimentellen Modalanalyse.....	23
Abbildung 2.24: Black-Box Modell .....	23
Abbildung 2.25: Imaginärteil der Vergrößerungsfunktion schematisch .....	24
Abbildung 2.26: Messung an einem spulengewickelten Wickelkopf [6].....	25
Abbildung 2.27: Schnappschuss einer animierten Eigenform [6] .....	26
Abbildung 3.1: Skizze des SOLID 45 Volumenelements [13].....	28
Abbildung 3.2: Skizze des COMBIN 14 Elements [13] .....	29
Abbildung 3.3: Die Geometrie im Raum mit Koordinatensystem CSYS,11 .....	31
Abbildung 3.4: Mögliche Ausgangssituationen für den Leiterquerschnitt.....	32
Abbildung 3.5: Erlaubte und nicht erlaubte Linien zwischen den Konturlinienzügen ...	33

---

Abbildung 3.6: Alter und neuer Algorithmus .....	34
Abbildung 3.7: Warping Factor bei ebenen Flächen [13] .....	35
Abbildung 3.8: Warping Factor bei einem Würfel bzw. 8-Knoten Element [13] .....	35
Abbildung 3.9: Seitenverhältnis bei Rechtecken und Quadern [13] .....	36
Abbildung 3.10: Winkelabweichung von idealerweise parallelen Kanten [13] .....	36
Abbildung 3.11: Eckenwinkel [13] .....	37
Abbildung 3.12: Position des Nutendes im Bezug zu den Leitern .....	38
Abbildung 3.13: Federanbindung und Knotennummerierung hinter dem Nutende .....	38
Abbildung 3.14: Nummerierung der Konturlinien im Leiterquerschnitt .....	40
Abbildung 3.15: Stabgeometrie mit Blechpaket .....	41
Abbildung 3.16: Spulengeometrie .....	41
Abbildung 3.17: Wegschneiden nicht gebrauchter Geometrieteile .....	42
Abbildung 3.18: Schrittweises Aufschneiden .....	43
Abbildung 3.19: Bisektionsverfahren .....	44
Abbildung 3.20: Auftrennen am Spulenaug (Konzept 1) .....	45
Abbildung 3.21: Auftrennen am Spulenaug (Konzept 2) .....	45
Abbildung 3.22: Oberspule .....	46
Abbildung 3.23: Unterspule .....	46
Abbildung 3.24: Spulenteile wireframe .....	46
Abbildung 3.25: Spulenteile wireframe ohne Querlinien .....	47
Abbildung 3.26: Verdrehen der Spulenteile: .....	47
Abbildung 3.27: Vorteilen der Konturlinien .....	48
Abbildung 3.28: Gruppieren der Vorteilungspunkte .....	49
Abbildung 3.29: Erzeugen einer neutralen Faser .....	49
Abbildung 3.30: Fehler durch ungünstig gewählte Vorteilung: .....	50
Abbildung 3.31: Definition von Parameter „ratio_12“ am Unterspulenteil .....	50
Abbildung 3.32: Unterspulenteil in 2 Abschnitten vernetzt .....	51
Abbildung 3.33: Ausrichtung der lokalen Z-Achsen und Schnitt der Konturlinien .....	52
Abbildung 3.34: Drehen der lokalen Koordinatensysteme: .....	53
Abbildung 3.35: Beide Spulenteile fertig vernetzt .....	54
Abbildung 3.36: Vernetzte Spulenteile mit modelliertem Nutendbereich .....	54
Abbildung 3.37: Kopplung der gemusterten Spulenteile am Spulenaug .....	55
Abbildung 3.38: Positionsskizze der Gewölbekeile: .....	56
Abbildung 3.39: Detailbild der Gewölbekeile [6] .....	57

---

Abbildung 3.40: 3D-wireframe-Skizze der Gewölbekeile .....	58
Abbildung 3.41: Hilfsfläche zur Modellierung der Gewölbekeilknoten.....	59
Abbildung 3.42: Aufbau des Leiterquerschnitts [6].....	60
Abbildung 3.43: Eigenform der 1. Eigenfrequenz.....	63
Abbildung 3.44: Eigenform der 2. Eigenfrequenz.....	64
Abbildung 3.45: Eigenform der 3. Eigenfrequenz.....	65
Abbildung 3.46: Eigenform der 4. Eigenfrequenz.....	66
Abbildung 3.47: Eigenform der 5. Eigenfrequenz.....	67
Abbildung 3.48: Eigenform der 6. Eigenfrequenz.....	69
Abbildung 3.49: Eigenform der 8. Eigenfrequenz.....	70
Abbildung 3.50: Auswertung der Parameterstudie .....	72

## 6 Tabellenverzeichnis

Tabelle 2.1: Ergebnisse der Messung.....	25
Tabelle 3.1: Knotennummerierung.....	39
Tabelle 3.2: Parameterbezeichnung.....	62
Tabelle 3.3: Vergleich 1. Eigenfrequenz.....	63
Tabelle 3.4: Vergleich 2. Eigenfrequenz.....	64
Tabelle 3.5: Vergleich 3. Eigenfrequenz.....	65
Tabelle 3.6: Vergleich 4. Eigenfrequenz.....	66
Tabelle 3.7: Simulationsergebnis 5. Eigenfrequenz.....	68
Tabelle 3.8: Simulationsergebnis 7. Eigenfrequenz.....	68
Tabelle 3.9: Vergleich 6. Eigenfrequenz.....	69
Tabelle 3.10: Vergleich 8. Eigenfrequenz.....	70
Tabelle 3.11: Gegenüberstellung der Ergebnisse.....	71

## 7 Literaturverzeichnis

- [1] J. Gieseke und E. Mosonyi, Wasserkraftanlagen, Planung Bau und Betrieb, 5. Auflage, 2009.
- [2] B. Schlegl, „Untersuchung von Wickelkopfschwingungen in Großgeneratoren,“ Dissertation an der Technischen Universität Graz, 2013.
- [3] C. Scheinecker, „Entwicklung eines parametrisierten Berechnungstools zur Untersuchung von Wickelkopfschwingungen bei Hydrogeneratoren“ Diplomarbeit an der Technischen Universität Graz, 2012.
- [4] F. Schönleitner, „Faserverbundwerkstoffe in Wickelköpfen,“ Diplomarbeit an der Technischen Universität Graz, 2010.
- [5] „Maglev Express,“ [Online]. Available: <http://www.magnetbahnen.de/linearmotor/grundlagen.htm>. [Zugriff am 25 September 2013].
- [6] Private Korrespondenz mit ANDRITZ HYDRO GmbH.
- [7] J. Teigelkötter, Energieeffiziente elektrische Antriebe: Grundlagen, Leistungselektronik, Betriebsverhalten und Regelung von Drehstrommotoren, Aschaffenburg: Springer, 2012.
- [8] „Partzsch Unternehmensgruppe,“ [Online]. Available: <http://de.partzsch.de/schleifringlaeufer>. [Zugriff am 25 September 2013].
- [9] „Wikimedia-Hoover Dam,“ [Online]. Available: [http://upload.wikimedia.org/wikipedia/commons/thumb/6/69/Hoover\\_Dam\\_Generator.jpg/450px-Hoover\\_Dam\\_Generator.jpg](http://upload.wikimedia.org/wikipedia/commons/thumb/6/69/Hoover_Dam_Generator.jpg/450px-Hoover_Dam_Generator.jpg). [Zugriff am 25 September 2013].

- [10] „Krempel Group,“ [Online]. Available: <http://www.krempel-group.com/deutsch/home/anwendungen/elektroisolierstoffe/generatoren.html>. [Zugriff am 02 September 2013].
- [11] M. Leibetseder, „Numerische Berechnung der Einspannsituation von Wicklungsstäben in Hydrogeneratoren,“ Bachelorarbeit an der Technischen Universität Graz, 2011.
- [12] C. C. Celigoj und M. O. Schmidt, Methode der Finiten Elemente, Graz: Institut für Festigkeitslehre an der TU Graz, Juni 1998.
- [13] „ANSYS® release 11.0 Documentation,“ ANSYS, Inc., [Online]. Available: [http://www.kxcad.net/ansys/ANSYS/ansyshelp/Hlp\\_E\\_COMBIN14.html](http://www.kxcad.net/ansys/ANSYS/ansyshelp/Hlp_E_COMBIN14.html). [Zugriff am 30 April 2011].
- [14] G. Müller und C. Groth, FEM für Praktiker, 8. neu bearbeitete Auflage, Bd. 1, Expert Verlag, 2007.
- [15] K. Willner, Kontinuums und Kontaktmechanik; Kapitel: "Die Methode der Finiten Elemente" S.383-416, 8. Auflage, Springer Verlag, 2009.
- [16] A. Binder, „Elektrische Maschinen und Antriebe,“ Springer Verlag, Berlin, Heidelberg, 2012.



## 8 Anhang

### 8.1 Programmbeschreibung „Spulenmodell“

#### 8.1.1 Variablendeklaration

Variablenname	Beschreibung
<b>A_Nutende</b>	Ebene normal auf die Generatorachse mit Abstand <b>Nutende</b> vom Generatormittelpunkt
<b>actdistance</b>	Berechneter Abstand zwischen den zwei aktuell bearbeiteten Vorteilungs-KP
<b>actLINE</b>	Liniennummer der aktuell bearbeiteten Linie
<b>actLINE_KP1</b>	Keypoint 1 der aktuell bearbeiteten Linie
<b>actLINE_KP2</b>	Keypoint 2 der aktuell bearbeiteten Linie
<b>break</b>	Abbruchkrit. (wenn diese Variable gleich 1 wird, dann wird die Schleife abgebr.)
<b>Contourline</b>	Nummer der aktuell bearbeiteten Konturlinie
<b>Contourline_KP1</b>	Keypoint 1 der aktuell bearbeiteten Konturlinie
<b>Contourline_KP2</b>	Keypoint 2 der aktuell bearbeiteten Konturlinie
<b>delta</b>	Abstand zwischen den Bisektionsschritten (wird jeden Schritt halbiert, ggf. negiert)
<b>epsilon</b>	Schwellwert für den Abbruch des Bisektionsverfahrens
<b>i_Stab</b>	<b>1</b> für Oberstab, <b>2</b> für Unterstab, beschreibt Nummer des Programmaufrufs
<b>I_step</b>	Zählervariable für die Anzahl der Schritte bei der Teilungspunktsuche
<b>Isolation_B</b>	Isolationsdicke zusätzlich zur Parasolid-Geometrie in lokalen X-Koordinaten
<b>Isolation_H</b>	Isolationsdicke zusätzlich zur Parasolid-Geometrie in lokalen Y-Koordinaten
<b>KPact</b>	Speichert die aktuell zu verarbeitenden Keypointnummer für die nächsten Befehle
<b>KPcomp</b>	Komponentenname für eine Gruppe von KPs, um diese leichter zu selektieren
<b>KPcontmaxX_1</b>	Betragsmäßig maximale X-Koordinate aller Keypoints im 1. Quadranten
<b>KPcontmaxX_2</b>	Betragsmäßig maximale X-Koordinate aller Keypoints im 2. Quadranten
<b>KPcontmaxX_3</b>	Betragsmäßig maximale X-Koordinate aller Keypoints im 3. Quadranten
<b>KPcontmaxX_4</b>	Betragsmäßig maximale X-Koordinate aller Keypoints im 4. Quadranten
<b>KPcontmaxY_1</b>	Betragsmäßig maximale Y-Koordinate aller Keypoints im 1. Quadranten
<b>KPcontmaxY_2</b>	Betragsmäßig maximale Y-Koordinate aller Keypoints im 2. Quadranten
<b>KPcontmaxY_3</b>	Betragsmäßig maximale Y-Koordinate aller Keypoints im 3. Quadranten
<b>KPcontmaxY_4</b>	Betragsmäßig maximale Y-Koordinate aller Keypoints im 4. Quadranten
<b>KPcontX_1</b>	X-Koordinate des aktuell bearbeiteten Keypoints auf einer Konturlinie im 1. Quadr.
<b>KPcontX_2</b>	X-Koordinate des aktuell bearbeiteten Keypoints auf einer Konturlinie im 2. Quadr.
<b>KPcontX_3</b>	X-Koordinate des aktuell bearbeiteten Keypoints auf einer Konturlinie im 3. Quadr.
<b>KPcontX_4</b>	X-Koordinate des aktuell bearbeiteten Keypoints auf einer Konturlinie im 4. Quadr.
<b>KPcontY_1</b>	Y-Koordinate des aktuell bearbeiteten Keypoints auf einer Konturlinie im 1. Quadr.
<b>KPcontY_2</b>	Y-Koordinate des aktuell bearbeiteten Keypoints auf einer Konturlinie im 2. Quadr.
<b>KPcontY_3</b>	Y-Koordinate des aktuell bearbeiteten Keypoints auf einer Konturlinie im 3. Quadr.
<b>KPcontY_4</b>	Y-Koordinate des aktuell bearbeiteten Keypoints auf einer Konturlinie im 4. Quadr.
<b>KPcoordX</b>	Koordinate in X-Richtung des aktuell bearbeiteten Keypoints
<b>KPcoordY</b>	Koordinate in Y-Richtung des aktuell bearbeiteten Keypoints

<b>KPcount</b>	Anzahl aller def. Keypoints in der Auswahl (oft gleich der Anz. der Konturlinien)
<b>KPdivideX</b>	X-Koordinate des jeweiligen Verteilungskeypoints zum darauffolgenden Summ.
<b>KPdivideY</b>	Y-Koordinate des jeweiligen Verteilungskeypoints zum darauffolgenden Summ.
<b>KPdivideZ</b>	Z-Koordinate des jeweiligen Verteilungskeypoints zum darauffolgenden Summ.
<b>KPmiddleX</b>	X-Koordinate des Stab <b>mittelpunkts</b> am Nutende bzw. der Keypoints der neutr. F.
<b>KPmiddleY</b>	Y-Koordinate des Stab <b>mittelpunkts</b> am Nutende bzw. der Keypoints der neutr. F.
<b>KPmx</b>	Beschreibt die aktuell höchste Keypointnr. (als Startwert für die Benennung neuer KPs)
<b>KPmx_new</b>	Beschreibt die nach einer Bearbeitung nach KPmx aktuell höchste Keypointnummer
<b>KPnutendeX</b>	X-Koordinate des jeweiligen Keypoints am Nutende
<b>KPnutendeY</b>	Y-Koordinate des jeweiligen Keypoints am Nutende
<b>KPstatact</b>	Komponentenname für alle KPs zu diesem Zeitpunkt (aktueller Status der Geom.)
<b>KPstatold</b>	Komponentenname für alle KPs vor der Bearbeit. (alter Status der Geom. vorher)
<b>LINEact</b>	Speichert die aktuell zu verarbeitende Liniennummer für die nächsten Befehle
<b>LINEcomp</b>	Komponentenname für eine Gruppe von Linien, damit diese leichter selekt. werden können.
<b>LINEcount</b>	Anzahl aller definierten Linien im selektierten Set
<b>LINEstatold</b>	Komponentenname für alle LINES zu diesem Zeitpunkt (alter Status der Geom. vorher)
<b>mindistance</b>	Minimaler Abstand zwischen zwei Verteilungs-Keypoints
<b>n_Nuten</b>	<b>Nuten</b> anzahl über den Umfang des Stators
<b>neutralFibre</b>	Liniennummer der neutralen Faser
<b>neutralFibre_KP1</b>	Keypoint 1 der neutralen Faser
<b>neutralFibre_KP2</b>	Keypoint 2 der neutralen Faser
<b>numKP</b>	Anzahl der Schnittkeypoints der Linien mit der aktuellen Working Plane
<b>Nutende</b>	Maß vom Generatormittelpunkt zum Nutende (Z-Abstand von CSYS,11 zum CSYS,13)
<b>off_OS_US</b>	Knoten <b>offset</b> zwischen Ober- und Unterstab = <b>n_Nuten</b> mal <b>off_Stb</b>
<b>off_Stb</b>	fixer Wert <b>10000</b> damit die Nummerierungen bei Ober- und Unterspule nicht kollidieren
<b>Orientierung_OS</b>	Variable aus dem Parameterfile um einzugeben, ob die Oberspule beim Blick entgegen der CSYS,11 Z-Achse links oder rechts liegt.
<b>P1</b>	Zwischenspeicher für KP-Nummern zum Erzeugen der Abschnitte an der neutralen Faser
<b>P2</b>	Zwischenspeicher für KP-Nummern zum Erzeugen der Abschnitte an der neutralen Faser
<b>rangeXmn</b>	Parameter für die <b>minimal</b> vorkommende X-Koordinate aller selektierten Entitäten
<b>rangeXmx</b>	Parameter für die <b>maximal</b> vorkommende X-Koordinate aller selektierten Entitäten
<b>rangeYmn</b>	Parameter für die <b>minimal</b> vorkommende Y-Koordinate aller selektierten Entitäten
<b>rangeYmx</b>	Parameter für die <b>maximal</b> vorkommende Y-Koordinate aller selektierten Entitäten
<b>rangeZmn</b>	Parameter für die <b>minimal</b> vorkommende Z-Koordinate aller selektierten Entitäten
<b>rangeZmx</b>	Parameter für die <b>maximal</b> vorkommende Z-Koordinate aller selektierten Entitäten
<b>ratio_12</b>	Position zwischen Nutende (0) und Teilungsebene am Spulenaug (1) auf der neutr. Faser
<b>rownumber</b>	Startwert bzw. Position für die Suche von nächstgeleg. Keypoints für den nächsten Schritt
<b>slotangle</b>	Nutwinkel des entspr. Stababschnittes, Schiefstellung der Nut bezogen auf die Y-Achse
<b>splitangle</b>	Teilungswinkel plus Sicherheitszuschlag, damit sicher nur am Spulenaug geteilt wird
<b>step</b>	Schrittweite für das Suchen des Spulenauges (muss kleiner als die Wickelkopflänge sein)
<b>sumKP</b>	Zählvariable für die Anzahl der bearbeiteten Keypoints
<b>SumKPcontX_1</b>	Summe aller X-Koordinaten der Keypoints auf den Konturlinien im 1. Quadr.
<b>SumKPcontX_2</b>	Summe aller X-Koordinaten der Keypoints auf den Konturlinien im 2. Quadr.
<b>SumKPcontY_1</b>	Summe aller Y-Koordinaten der Keypoints auf den Konturlinien im 1. Quadranten
<b>SumKPcontY_2</b>	Summe aller Y-Koordinaten der Keypoints auf den Konturlinien im 2. Quadranten

<b>SumKPMiddleX</b>	Summe aller X-Koordinaten der jeweiligen Keypoints, zum Mitteln der Position
<b>SumKPMiddleY</b>	Summe aller Y-Koordinaten der jeweiligen Keypoints, zum Mitteln der Position
<b>sumXpos</b>	Summenvariable für alle KPcoordX
<b>sumYpos</b>	Summenvariable für alle KPcoordY
<b>TAN_actangle</b>	<b>Tangens</b> des <b>aktuell</b> berechneten Winkels zum Vergleichen mit TAN_splitangle
<b>TAN_slotangle</b>	<b>Tangens</b> des Nutwinkels
<b>TAN_splitangle</b>	<b>Tangens</b> des Teilungswinkels
<b>Teilung</b>	Summe aus Teilung_1 und Teilung_2, bzw. die Anzahl der Elemente der Länge nach
<b>Teilung_1</b>	Parameter für die Elementteilung im Abschnitt vor ratio_12
<b>Teilung_2</b>	Parameter für die Elementteilung im Abschnitt nach ratio_12
<b>THXY</b>	Verdrehung des lokalen Koordinatensystems an der neutralen Faser um seine Z-Achse
<b>Vorteilung</b>	Vorteilung bedeutet, dass die Konturlinien unabhängig von der späteren Teilung der Elemente in Stücke geteilt werden um eine neutrale Faser der Geometrie zu ermitteln
<b>Xdir</b>	X-Koordinate des richtungsweisenden Punktes für die Ausrichtung der Z-Achse
<b>Xorig</b>	X-Koordinate des Ursprungs des lokalen Koordinatensystems
<b>Ydir</b>	Y-Koordinate des richtungsweisenden Punktes für die Ausrichtung der Z-Achse
<b>Yorig</b>	Y-Koordinate des Ursprungs des lokalen Koordinatensystems
<b>Zcoord</b>	Z-Koordinate der Eckpunkte im lokalen Koordinatensystem
<b>Zdir</b>	Z-Koordinate des richtungsweisenden Punktes für die Ausrichtung der Z-Achse
<b>Zorig</b>	Z-Koordinate des Ursprungs des lokalen Koordinatensystems
<b>Zpos</b>	Variable für die aktuelle Z-Position der Working Plane im Bisektionsverfahren

## 8.1.2 Arrays

Variablenname	Beschreibung
<b>KPosus(i,j)</b>	2D-Array zum Trennen der KPs in jene der Ober- bzw. Unterspule (i = laufende Nummer für die Keypoints, j = Sortierung in X größer oder kleiner als Null)
<b>KPsplit(i)</b>	1D-Array für die 3 Koordinaten des Teilungspunktes (wird im Bisektionsverfahren erm.)
<b>LINEcont8(i,j,k)</b>	3D-Array zum Vorteilen der Konturlinien (i = laufende Nummer für die Vorteilung, j = laufende Nummer für alle existierenden Konturlinien, k = Funktionsvariable)
<b>LINEmerge(i,j)</b>	2D-Array zum Verschmelzen der Konturlinien (i = laufende Nummer für die Keypoints, j = Funktionsvariable für Liniennummern, Keypointnummern und Koordinaten)
<b>LINEnutende(i,j)</b>	2D-Array für Linien welche am Nutende aufgetrennt werden müssen (i = laufende Nummer für die Linien, j = Funktionsvariable für Keypointnummern und Koordinaten)
<b>LINEosus(i,j)</b>	2D-Array zum Trennen der LINES in jene der Ober- bzw. Unterspule (i = laufende Nummer für die Linien, j = Funktionsvariable für Keypointnummern und Koordinaten)
<b>MESHcoord(i,j)</b>	2D-Array zum Erzeugen von lokalen Hilfskoordinatensystemen (i = laufende Nummer für die Teilung, j = Funktionsvariable für Keypointnummern, Koordinaten und Koordinatensysteme)
<b>MESHpoints(i,j,k)</b>	3D-Array zum Vorteilen der Konturlinien (i = laufende Nummer für die Elementteilung, j = laufende Nummer für alle existierenden Konturlinien inkl. Neutraler Faser, k = Sortierungsvariable für die 4 Querschnittsquadranten)

Folgende Beschreibung und die Nummerierung der Kapitel bezieht sich auf den Programmcode des Unterprogramms „Spulenmodell“.

## 1.10 Vorbereiten der Geometrie

### 1.10.1 Programmbaustein: Auswahl Oberstab oder Unterstab

Die Variable **iStab** wird bei jedem Programmaufruf um 1 erhöht. Wenn **iStab** = 1 ist, dann werden die Parameter **off\_OS\_US** und **off\_Stb** gleich Null gesetzt, beim zweiten Programmaufruf werden **off\_OS\_US** und **off\_Stb** so gesetzt, damit die Knotennummerierungen in den einzelnen Spulenteilen nicht kollidieren.

### 1.10.2 Löschen aller Volumenkörper und Flächen. Verschmelzen in der Nähe liegender Keypoints

**VDELE,all** löscht alle Volumenkörper

**ADELE,all** löscht alle Flächen

**NUMMRG,KP,5e-4,5e-4** legt all jene Keypoints zusammen, welche weniger als  $5 \times 10^{-4}$  m auseinanderliegen.

Mit dem Befehl **NUMSTR,KP** wird die aufsteigende Knotennummerierung mit Startwert 1 eingestellt.

### 1.10.3 Koordinatensysteme für weitere Unterprogramme erzeugen

**CSYS,0** liegt im Zentrum des Generators direkt auf der Drehachse, die Drehachse ist die Z-Achse, die Y-Achse zeigt in positiver Richtung radial von der Drehachse weg. Zwei Lokale Koordinatensysteme **11** und **12** werden am Ursprung des **CSYS,0** erzeugt. Diese Koordinatensysteme werden in den weiterfolgenden Programmteilen benötigt.

### 1.10.4 Koordinatensystem auf der Drehachse in Z um Nutende verschoben Working Plane als Schnittebene

Erzeugen des lokalen, kartesischen Koordinatensystems **13** am Ursprung des **CSYS,0** welches in positiver Z-Richtung (Drehachse) um den Wert **Nutende** verschoben ist und somit am Austritt der Spulen aus den Wicklungsblechen liegt. **WPCSYS,1,13** erzeugt eine Working Plane (Arbeitsebene) in der XY-Ebene des **CSYS,13**.

---

*Anmerkung: Alles auf der positiven Z-Richtung von **CSYS,13** gehört zum „frei schwingenden“ Teil des Wickelkopfes, alle Entitäten mit negativen Z-Koordinaten wird in Folge weggeschnitten.*

#### 1.10.5 Alle Linien am Nutende aufschneiden

**LSEL,all** selektiert alle Linien, der **\*GET**-Befehl speichert die Anzahl der selektierten Linien in die Variable **LINEcount**. **\*SET** erzeugt den benutzerdefinierten Parameter **LINEnutende**, der **\*DIM**-Befehl erzeugt ein 2-dimensionales Array mit dem Namen **LINEnutende** welches für jede vorkommende Linie eine Zeile und 5 Spalten enthält.

Die erste **\*DO**-Schleife zählt mit dem Parameter **i** alle Linien durch und schreibt zeilenweise die kleinste selektierte Liniennummer in die erste, die Nummern der an beiden Enden der Linie liegenden Keypoints in die 2. und 3. sowie deren Z-Koordinate in die 4. und 5. Spalte des Arrays. Die gerade untersuchte Linie wird abgewählt, sodass beim nächsten Durchlauf der **\*DO**-Schleife die global nächstgrößere Liniennummer die kleinste im selektierten Set ist...

Die zweite **\*DO**-Schleife vergleicht die Vorzeichen der Z-Koordinaten der Liniendenen. Sind diese Koordinaten im **CSYS,13** verschiedenen Vorzeichens, so muss die Linie die Working Plane, also die Ebene auf Höhe des Nutendes schneiden. Trifft diese Eigenschaft zu, so wird die entsprechende Linie mit dem **LSEL**-Befehl ausgewählt, mit **LSBW** an der Working Plane aufgetrennt und ein Keypoint am Schnittpunkt erzeugt.

Am Ende werden alle Linien und Keypoints hinter der Working Plane, also alle Entitäten mit negativen **CSYS,13** Z-Koordinaten werden gelöscht. Die Grenzen mit  $-10 \cdot \text{Nutende}$  bis  $-0,001\text{m}$  sind deshalb so gewählt, dass sicher alle Entitäten mit negativen Z-Koordinaten und keine Keypoints in der Working Plane ausgewählt werden.

## 1.20 Auftrennen des Modells

### 1.20.1 Suchen des Teilungspunktes bzw.

#### dessen Koordinaten (KPsplit)

Die **\*IF**-Schleife am Beginn des Programmpunktes bewirkt, dass nur beim ersten Aufruf des Unterprogramms nachfolgende Befehle ausgeführt werden.

**\*SET** und **\*DIM** erzeugen ein Array für die Teilungspunktkoordinaten **KPsplit**.

Status der Geometrie (KPs und LINES) wird in Komponenten **KPstatold** und **LINEstatold** gespeichert, Working Plane in X-Y-Ebene des CSYS,13 gelegt, Schrittweite für stückweises Durchschneiden der Geometrie auf 5 cm festgelegt und weitere Variablen **i\_step** für die Anzahl der Schritte und **numKP** für die Anzahl der gefundenen Schnittpunkte **\*DOWHILE**-Schleife festgelegt.

**\*DOWHILE**-Schleife wird ausgeführt, solange die Variable **numKP** größer NULL ist.

**KSEL** wählt alle Keypoints aus. Die Aktuellen Keypoints werden mit **CM** in **KPstatact** gespeichert.

**WPOFFS** erzeugt eine Working Plane im Abstand **step** von der vorigen Working Plane in Z-Richtung verschoben.

Die Variable **i\_step** wird bei jedem Schleifendurchlauf um eins erhöht. Mit **CMSEL** werden alle vorher vorhandenen Linien ausgewählt, mit **LSBW** mit der Working Plane geschnitten. Auswählen aller Keypoints und abwählen aller vor dem Schnitt vorhandenen Keypoints. Die Anzahl aller übriggebliebenen, neu erzeugten, Keypoints wird in **numKP** gespeichert.

Werden keine Keypoints mehr gefunden, dann ist die Variable **numKP** gleich NULL und die **\*DOWHILE**-Schleife wird abgebrochen.

---

!---- Start des Bisektionsverfahrens -----

Einmalige Definition der Variable **delta** mit dem Wert **-step/2**, einem Schwellwert **epsilon** unter dessen Wert die Schrittweite gesunken sein muss, damit das Bisektionsverfahren abgebrochen wird und **Zpos**, welche die aktuelle Position der Working Plane beschreibt.

**\*DO**-Schleife wird maximal **step/epsilon** mal ausgeführt, aber i.A. durch Abbruchbedingungen schon früher abgebrochen.

Keypoints werden zu **KPstatact** zusammengefasst

Die Working Plane wird um **delta** verschoben.  
*Anmerkung: Die Schrittweite delta ist zu Beginn -step/2 und wird bei den Schleifendurchläufen halbiert und je nachdem ob Schnittpunkte gefunden werden mit dem passenden Vorzeichen versehen. Die Working Plane nähert sich somit immer weiter an den äußersten Punkt der Geometrie.*

Die Variable **Zpos** wird ebenfalls um die Schrittweite **delta** verschoben.

**CMSEL** selektiert alle Linien der Geometrie, **LSBW** schneidet alle diese Linien mit der Working Plane.

Wie vorhin werden alle Keypoints abgewählt, alle vorher existierenden wieder abgewählt und die Anzahl der Restlichen in numKP gespeichert.

Die **\*IF**-Bedingung unterscheidet nun die Fälle ob Schnittpunkte gefunden worden sind oder nicht. Wenn **numKP** größer NULL, dann existieren Schnittpunkte und **delta** wird positiv definiert (Schritt nach vorne), andernfalls wird **delta** negativ gesetzt (Schritt in Richtung Nutende).

Die weitere **\*IF**-Bedingung bricht die **\*DO**-Schleife ab sobald der Betrag von **Delta** kleiner als **epsilon** wird.



Am Ende wird noch  $KPsplit(3) = Zpos + 2 * \epsilon$  gesetzt um sicherzustellen, dass der Punkt **KPsplit** außerhalb der Geometrie liegt.

!---- Ende des Bisektionsverfahrens -----

Definition und Nullsetzen von **sumXpos**, **sumYpos** und **sumKP**.

Inhalt der **\*DO**-Schleife wird 6 mal wiederholt.

Keypoints auswählen und in Komponente **KPstatact** zusammenfassen.

Working Plane um 10-fachen Schwellwert **epsilon** in Richtung Nutende verschieben.

Definierte Komponente **LINEstatold** auswählen, mit der Working Plane schneiden

Alle Keypoints auswählen, die Komponente **KPstatact** abwählen und die Anzahl der restlichen Keypoints in **numKP** speichern.

Weitere **\*DO**-Schleife für jeden gefundenen neuen Keypoint:

Kleinste Keypointnummer aller neuen Keypoints in **KPact**, seine X und Y-Koordinaten in **KPcoordX** und **KPcoordY** speichern.

Aufsummieren der Koordinaten in **sumXpos** und **sumYpos**.

Abwählen des aktuellen Keypoints, damit beim nächsten Durchlauf der Schleife der nun zweitkleinste der kleinste ist.

Hinaufzählen der Variable **sumKP**.

Ende beider **\*DO**-Schleifen.

Überschreiben der Arrayparameter **KPsplit(1)** und **KPsplit(2)** mit den gemittelten Koordinaten der durchlaufenen Punkte.

!---- Löschen aller zusätzlich erzeugten Entitäten -----

Löschen aller neu erzeugten Entitäten außer den am Anfang definierten Komponenten **KPstatold** und **LINEstatold**.

Ende der **\*IF**-Bedingung für ersten Programmaufruf.

### 1.20.2 Zuordnen aller KP zu OS oder US

**KSEL** wählt alle Keypoints aus. Mit den Beiden **\*GET**-Befehlen werden in die Variablen **KPmx** die größte vorkommende Keypointnummer und **KPcount** die Anzahl aller Keypoints geschrieben.

**\*SET** und **\*DIM** erzeugen ein 2-dimensionales Array **KPosus** mit ebenso vielen Zeilen wie es Keypoints gibt und 2 Spalten.

Die nachfolgende **\*DO**-Schleife zählt alle Keypoints durch: Die aktuell kleinste Keypointnummer in der Auswahl wird mit **\*GET** in **KPact** gespeichert, der 2. **\*GET**-Befehl schreibt die X-Koordinate des aktuellen Keypoints **KPact** in **KPcoordX**.

Die **\*IF**-Abfrage sortiert die Keypoints nun in jene, die „rechts“ bzw. „links“ vom Teilungskeypoint **KPsplit** liegen oder anders ausgedrückt, die eine größere bzw. kleinere **CSYS,11** X-Koordinate als dieser besitzen. Die Nummer jedes verarbeiteten Keypoints mit einer größeren X-Koordinate als der Teilungskeypoint wird in die erste Spalte des **KPosus**-Arrays gespeichert, alle mit kleineren X-Koordinaten in die 2. Spalte. Sobald der Keypoint verarbeitet ist oder nicht verarbeitet werden kann, wird er abgewählt, sodass beim nächsten Durchlauf der **\*DO**-Schleife die nächstgrößere Nummer die aktuell kleinste ist.

### 1.20.3 Teilungswinkel ermitteln

Definition des **TAN\_splitangle** als Tangens des Schnittwinkels.

Erneutes Durcharbeiten aller Keypoints mit einer **\*DO**-Schleife:

Die äußere **\*IF**-Abfrage liest nun den im Eingabefile definierten Parameter **Orientierung\_OS** ein.

*Anmerkung: Ist dieser auf „li“, also auf Links, eingestellt, so dreht bei einer Betrachtung von Vorne, also entgegen der positiven **CSYS,11** Z-Achse, jener der Drehachse näher liegende Spulenteil (OS) nach links (Gegenuhrzeigersinn) weg. In der nicht aufgetrennten Spule muss dieser achsnähere Teil aber bei Beibehaltung der Blickposition der rechte sein. Die Auftrennung des Stabes soll*

*möglichst normal zur neutralen Faser der Spule erfolgen, deshalb wird die Teilungsebene so gelegt, dass diese knapp an der achsnächsten Kante des achsferneren Spulenteils (US) vorbeigeht. Dieser liegt bei der ungeteilten Spule noch linker Hand. Bei Ausrichtung „re“ stelle man sich die Geometrie gespiegelt zur YZ-Ebene vor.*

Für den vorliegenden Fall ist die Ausrichtung „li“, ist also nur der erste Teil der \*IF-Abfrage relevant. Wie in der Anmerkung beschrieben, wird nun die Teilungsebene so aufgespannt, dass sie durch den Teilungspunkt **KPsplit** und knapp am achsferneren Spulenteil (US, links) vorbeigeht. Wurde also der aktuell zu bearbeitende Keypoint in *Programmteil 1.20.2* der 2. Spalte des Arrays **KPosus** zugeordnet, liegt dieser also links des Teilungspunktes, so werden mit den \*GET-Befehlen seine Koordinaten in die temporären Variablen **KPcoordX** und **KPcoordY** gespeichert. Der Tangens des aktuellen Winkels **TAN\_actangle** zwischen der **CSYS,11** X-Achse und der Verbindungsstrecke von Teilungskeypoint und dem aktuell zu bearbeitenden Keypoint wird im Anschluss berechnet bevor mit der inneren \*IF-Abfrage der Tangens des Teilungswinkels **TAN\_splitangle** mit dem Tangens des aktuellen Winkels **TAN\_actangle** überschrieben wird, sofern dieser größer ist.

Am Ende des Prozedere ist in der Variable **TAN\_splitangle** der maximale Wert aller berechneten **TAN\_actangle** gespeichert. Der tatsächliche Teilungswinkel **splitangle** wird mit  $ATAN(TAN\_splitangle)$  berechnet, von Rad auf Grad umgerechnet und mit einem Sicherheitszuschlag (1,05) multipliziert, damit auch der exponierteste Keypoint noch sicher auf der richtigen Seite der Teilungsebene zu liegen kommt.

#### 1.20.4 Definieren des Teilungskoordinatensystems und der Teilungsebene

Umschalten auf **CSYS,11**.

Wenn die Variable **iStab** gleich 1 ist, dann wird die Oberspule (OS) benötigt, bei **iStab** gleich 2 die Unterspule (US). Das lokale, kartesische Koordinatensystem **CSYS,14** wird nun mit seinem Ursprung in den Teilungskeypoint **KPsplit** gelegt und abhängig vom benötigten Spulenteil dergestalt gedreht, dass die Teilungsebene von der XY-Ebene des **CSYS,14** aufgespannt wird

und der **nicht** benötigte Spulenteil **positive CSYS,14 Z-Koordinaten** hat.

#### 1.20.5 Definieren der Teilungsebene (Aufteilung un OS und US)

Umschalten auf CSYS,14.

Erzeugen einer Working Plane auf Basis des **CSYS,14**. Eine Working Plane wird immer in der XY-Ebene des zugrundeliegenden Koordinatensystems aufgespannt.

#### 1.20.6 Linienzüge auftrennen

Auswählen aller Linien mit **LSEL**. **\*GET** schreibt in **LINEcount** die Anzahl aller ausgewählten Linien. **\*SET** und **\*DIM** erzeugen ein 2-dimensionales Array **LINEosus** mit ebenso vielen Zeilen wie es Linien gibt und 5 Spalten.

Die erste **\*DO**-Schleife arbeitet alle Linien durch und schreibt in das Array **LINEosus** in die erste Spalte die aktuell kleinste Liniennummer aller ausgewählten Linien, in die 2. und 3. Spalte die Nummern der beiden Keypoints an den Endpunkten der jeweiligen Linie und in die 4. und 5. Spalte die entsprechenden **CSYS,14 Z-Koordinaten** der Keypoints aus den Spalten 2 und 3. Jede abgearbeitete Linie wird abgewählt, sodass beim nächsten Durchlauf der **\*DO**-Schleife die nächstgrößere Nummer die aktuell kleinste ist.

Die zweite **\*DO**-Schleife arbeitet erneut alle Linien durch.

Die beiden **\*IF**-Abfragen überprüfen ob die **CSYS,14 Z-Koordinaten** der Linien unterschiedliches Vorzeichen haben, sprich ob die jeweilige Linie die XY-Ebene des **CSYS,14** Koordinatensystems, bzw. die der XY-Ebene äquivalente Working Plane, schneidet. Ist dies der Fall, so wird mit **LSEL** die Linie ausgewählt und mit **LSBW** die Linie an der Working Plane aufgetrennt.

#### 1.20.7 Löschen aller Entitäten in Abhängigkeit des benötigten Spulenteils (OS/US)

**KSEL** wählt alle Keypoints aus.

**\*GET** schreibt die maximal vorkommende **CSYS,14 Z-Koordinate** aller Keypoints in **rangeZmx**.

---

Auswählen und Löschen aller Linien und Keypoints welche nicht zum gewünschten Spulenteil gehören.

### 1.20.8 Löschen aller Linien welche quer zwischen den Konturlinien liegen

Aktivieren von **CSYS, 13**

Alle Linien auswählen und die Anzahl in **LINEcount** schreiben.

Alle Keypoints mit der Z-Koordinate gleich NULL (Nutende) werden ausgewählt und die Anzahl in **KPcount** gespeichert.

Erzeugen eines 2D-Arrays **LINEmerge** mit so vielen Zeilen wie Konturlinienzüge (i.A. 8 Stück) und 7 Spalten

Nullsetzen der Abbruchvariable **break**.

**\*DO**-Schleife wird für jede Unterbrechung der Konturlinien mindestens 1 x ausgeführt.

!---- Startpunkt des ersten Linienzugs einlesen -----

Auswählen der Keypoints am Nutende

**\*DO**-Schleife für alle Konturlinienzüge

Kleinste Liniennummer wird in die erste Spalte von **LINEmerge** geschrieben.

Abwählen der aktuell kleinsten Liniennummer, damit beim nächsten Durchlauf die aktuell zweitkleinste Liniennummer die kleinste ist.

Die Nummern der ersten Linienstücke stehen nun in den ersten Spalten von **LINEmerge**.

Ende der **\*DO**-Schleife

**\*DO**-Schleife für alle Konturlinienzüge

Startpunkt des entsprechenden Konturlinienzugs aus **LINEmerge** auswählen, dazugehörige Linie auswählen und ihre Nummer in die zweite Spalte von **LINEmerge** schreiben.

---

Ende der **\*DO**-Schleife

!---- Abbruchkriterium überprüfen -----

Aktivieren von **CSYS, 14**

**\*DO**-Schleife für alle Konturlinienzüge

Keypoint 1 der bisher verschmolzenen  
Konturlinie in Spalte 4 schreiben, Keypoint 2 in  
Spalte 5.

**CSYS, 14** Z-Position von Keypoint 1 und 2 in  
Spalten 6 und 7 schreiben.

**\*IF**-Bedingung, wenn einer der beiden  
Keypointkoordinaten betragsmäßig kleiner als e-  
10 m also fast Null ist, dann wird die  
Abbruchvariable **break** NULL gesetzt.

Ende der **\*DO**-Schleife

**CSYS, 13** aktivieren

**\*IF**-Bedingung, wenn **break = 1** , dann breche Schleife ab.

!---- Querlinien löschen -----

Erste bisher verschmolzene Konturlinie auswählen.

Mit der **\*DO**-Schleife alle weiteren bisher  
verschmolzenen Konturlinien auswählen.

Alle Keypoints der ausgewählten Linien auswählen.

Mit der **\*DO**-Schleife alle Startkeypoints abwählen.

Alle Linien, welche ausschließlich selektierte Keypoints  
enthalten (Querlinien) auswählen und löschen.

!---- aufeinanderfolgende Linienzüge löschen -----

**\*DO**-Schleife für alle Konturlinienzüge

Bisher verschmolzenen Konturlinienzug auswählen.  
 Alle Keypoints des Linienzugs auswählen.  
 Alle Linien, welche einen selektierten Keypoint besitzen auswählen -> bisher verschmolzene und nachfolgende Linie.  
 Abwählen des bisher verschmolzenen Linienabschnitts.  
 Übriggebliebenen nachfolgenden Abschnitt in die 3. Spalte von **LINEmerge** schreiben.

Auswählen des bisher verschmolzenen (Spalte 2) und des nachfolgenden (Spalte 3) Konturlinienzugs.  
 Beide Linien kombinieren.

Ende der **\*DO**-Schleife.

Ende der höheren **\*DO**-Schleife für jede Unterbrechung.

Startpunkt der ersten Konturlinie auswählen.

Mit der **\*DO**-Schleife alle restlichen Startkeypoints auswählen.

Dazugehörige Linien auswählen.

Mit **LSEL,inve** Auswahl invertieren.

Alle nun ausgewählten Linien löschen.

#### 1.20.9 Definition eines KS, sodass die Orientierung in der Nut parallel zur y-Achse liegt

Aktivieren von CSYS, 13

Definition und Nullsetzen von **KPmiddleX, Y, SumKPmiddleX** und **Y**.

**\*DO**-Schleife für alle Konturlinienzüge

Speichern der X und Y Koordinaten der Startkeypoints am Nutende in die Variablen **KPnutendeX** und **Y** und Aufsummieren zu den Variablen **SumKPmiddleX** und **Y**.

Ende der **\*DO**-Schleife.

Die Mittelwerte der Koordinaten werden in **KPmiddleX** und **Y** gespeichert.

Jene Winkel um welchen die Spulenteile verschoben werden müssen, damit diese in einer Nut zu liegen kommen, ist der **slotangle**.

Der Tangens des **slotangle** ist mit **TAN\_slotangle** als Verhältnis von mittleren X zu Y Koordinaten definiert.

Der **slotangle** entsprechend der **ATAN** seines Tangens.

#### 1.20.10 Definieren eines KS im Schnittquerschnitt am Nutende

Aktivieren des **CSYS, 11**.

Erzeugen eines lokalen Koordinatensystems **CSYS, 15** mit Ursprung am Mittelpunkt des Stabes am Nutende und um den **slotangle** um die Z-Achse des **CSYS,11** gedreht.

#### 1.20.11 Definieren von Komponenten zur besseren Selektierbarkeit beim Herauslöschten von Hilfsgeometrien

Alle Keypoints und Linien auswählen und in die Komponenten **KPcomp** und **LINEcomp** zusammenfassen.

### 1.30 Vernetzen der Linienzüge

Alles auswählen

Alle Keypoints welche näher als 0,5mm beisammen liegen verschmelzen.

#### 1.30.1 Vorteilen der Konturlinien (im Normalfall 8 Stück)

Die **Vorteilung** wird gleich der im Eingabefile definierten **Vorteilung\_Eingabefile** gesetzt.

Aktivieren von **CSYS, 15**

Alle Keypoints mit der Z-Koordinate gleich NULL (Nutende) werden ausgewählt und die Anzahl in **KPcount** gespeichert.



Erzeugen eines 3D-Arrays **LINEcount8** mit **Vorteilung+1** Zeilen, **KPcount+1** Spalten und **6** Speicherebenen pro Zeile/Spalte-Kombination.

Auswählen aller Keypoints und speichern der größten Keypointnummer in **KPmx**. -> um Kollisionen mit der Nummerierung zu vermeiden.

### 1.30.2 Startkeypoints

Alle Keypoints mit der Z-Koordinate gleich NULL (Nutende) werden ausgewählt.

**\*DO**-Schleife für jeden Konturlinienzug (Startkeypoints einlesen)

Speichern der kleinsten Keypointnummer aus der Auswahl in **LINEcount** erste Zeile für die ersten Punkte und erste Ebene für die Keypointnummer. Die 2., 3. Und 4. Ebene wird mit den Koordinaten in X,Y und Z des jeweiligen Punktes beschrieben. Ebene 5 ist für den nächsten Programmpunkt von Relevanz.

Abwählen des bearbeiteten Punktes, damit beim nächsten Schleifendurchlauf die aktuell zweitkleinste die kleinste Keypointnummer ist.

Ende der **\*DO**-Schleife.

**\*DO**-Schleife für jeden Konturlinienzug (Konturlinien teilen)

!---- Contourline teilen -----

Auswählen des vorher eingelesenen Startkeypoints aus **LINEcount8**.

Dazugehörige Linie auswählen.

Liniennummer in Variable **Contourline** speichern.

Den ersten Keypoint der **Contourline** in **Contourline\_KP1** speichern.

Festlegen der Keypointnummern neu erzeugter Keypoints auf **KPmx+(j-1)\*Vorteilung+1** um Kollisionen mit der Nummerierung zu vermeiden.

**\*IF**-Bedingung, wenn **Contourline\_KP1** gleich dem in **LINEcount8** gespeicherten Startkeypoint ist, dann:

Schreibe den 2. Keypoint der **Contourline** in **Contourline\_KP2**.

Teile die Linie von **Contourline\_KP1** aus in Vorteilung Teilstücke.

Wenn nicht, dann:

Schreibe den 1. Keypoint der Contourline in **Contourline\_KP2**.

Teile die Linie von **Contourline\_KP2** aus in Vorteilung Teilstücke. *(Das negative „Vorzeichen“ vor der Liniennummer bewirkt das Teilen von KP2 aus)*

Ende der **\*IF**-Bedingung.

!---- Endkeypoints -----

Alle Keypoints auswählen.

Den Keypoint in der letzten Zeile (Vorteilung+1) des **LINEcont8**-Arrays (Endkeypoint der Konturlinie in der Teilungsebene) gleich dem **Contourline\_KP2** setzen. Die X, Y und Z Koordinaten des Punktes in die Ebenen 2, 3 und 4 des Arrays schreiben.

**\*DO**-Schleife für jeden Teilungspunkt zwischen Start- und Endkeypoint (Zeile **2** bis Zeile **Vorteilung**):

Setze die Keypointnummer im Array **LINEcont8** gleich der vorher definierten Keypointnummer *(deshalb war die Teilung vom Nutende aus wichtig, andernfalls wären die Nummern der Keypoints nicht zwingend ansteigend vom Nutende aus gesehen)*

Schreiben der X, Y und Z Koordinaten des Punktes in die Ebenen 2, 3 und 4 des Arrays **LINEcont8**.

Ende der **\*DO**-Schleife

Ende der nächsthöheren **\*DO**-Schleife für jeden Konturlinienzug

### 1.30.3 Suchen nächstgelegener Keypoints für möglichst normal zur Stabsachse liegender Querschnitte

Auswählen aller Keypoints und speichern der größten Keypointnummer in **KPmx**. -> um Kollisionen mit der Nummerierung zu vermeiden.

**\*DO**-Schleife für jeden Teilungspunkt ab dem Startpunkt bis zum Endkeypoint (Zeile **2** bis Zeile **Vorteilung+1**):

*Anmerkung: Die Teilungspunkte der ersten Linie (Spalte1) sind die Ausgangspunkte für die Suche möglichst normal zur Stabsachse liegender Querschnitte. Davon ausgehend werden jene Teilungspunkte auf den nächsten Linien (Spalte 2, usw) gesucht, welche zum vorigen Punkt den kleinsten Abstand besitzen. Die Werte in Ebene 5 geben die Position an bei welchem die Suche beim vorigen Querschnitt erfolgreich war. Da sich definitionsgemäß die Querschnitte nicht überschneiden dürfen, wird beim nächsten Durchlauf der Konturlinien an dieser Position zu suchen begonnen.*

Setze die Werte in der 5. Ebene der ersten Spalte gleich der Zeilennummer.

**\*DO**-Schleife für jeden Konturlinienzug bis auf den letzten:

Nullsetzen der Variablen **mindistance** und **break**.

**\*DO**-Schleife für jeden Teilungspunkt vom Start-bis vor den Endkeypoint (Zeile **1** bis Zeile **Vorteilung**):

Setze die Variable **rownumber** auf die vorher gespeicherte „erfolgreiche“ Position.

*Anmerkung: Im Falle der ersten Konturlinie, von welcher ausgegangen wird, ist diese **rownumber** gleich **i**, ansonsten jener Wert welcher in der folgenden **\*ELSEIF**-Bedingung überschrieben wird)*

Setze die Koordinaten X1, Y1 und Z1 gleich den Koordinaten des vorher gefundenen, nächstgelegenen Punktes auf Konturlinie **j** und die Koordinaten X2, Y2 und Z2 gleich den Koordinaten laufenden Punktes auf der Konturlinie **j+1**.

---

**actdistance** berechnet nun das Quadrat des Abstandes zwischen beiden Punkten auf den verschiedenen Konturlinien.

**\*IF-Bedingung:**

Wenn beim ersten Durchlauf der Schleife **mindistance** = 0 ist, dann überschreibe **mindistance** mit **actdistance**.

Oder wenn die aktuell berechnete Distanz **actdistance** zwischen zwei Keypoints kleiner oder gleich der **mindistance** ist, dann überschreibe **mindistance** mit **actdistance** und setze die **rownumber** (**LINEcont8(i,j+1,5)**)

für den nächsten abzuarbeitenden Linienzug gleich der aktuellen Position **l**. Die Abbruchvariable **break** wird 1 gesetzt.

*Anmerkung: Sollte beim nächsten Durchlauf der innersten \*DO-Schleife ein noch kleinerer Abstand gefunden werden, so wird sowohl die **mindistance** als auch die **rownumber** (**LINEcont8(i,j+1,5)**) für den nächsten abzuarbeitenden Linienzug neu überschrieben.*

Oder wenn die aktuell berechnete Distanz **actdistance** zwischen zwei Keypoints größer der **mindistance** ist und schon vorher die vorherige Bedingung erfüllt wurde, sprich **break** schon **1** ist, dann verlasse die innerste **\*DO**-Schleife mit **\*EXIT**.

**\*ENDIF**

Ende der innersten **\*DO**-Schleife.

Ende der **\*DO**-Schleife für alle Linienzüge.

Ende der **\*DO**-Schleife für alle Vorteilungspunkte.

1.30.4 Erzeugen der KPs für die neutrale Faser (9. Linie)  
bis auf den letzten Punkt

Alle Keypoints auswählen und die größte existierende Keypointnummer in **KPmx** speichern.

**\*DO**-Schleife für alle Vorteilungspunkte bis auf den letzten

Nullsetzen der Summenvariablen **SumKPmiddleX, Y** und **Z**.

**\*DO**-Schleife für alle Linienzüge

Setze die Variable **rownumber** gleich **LINEcont8(i,j,5)**

Setze **KPdivideX, Y** und **Z** gleich den in **LINEcont8** auf Ebenen 2, 3 und 4 gespeicherten Koordinaten.

Summiere alle Koordinaten **KPdivideX, Y** und **Z** eines möglichst normal zur Stabachse liegenden Querschnitts zusammen zu **SumKPmiddleX, Y** und **Z**.

**\*ENDDO**

Ermittlung der Mittelwerte **KPmiddleX, Y** und **Z** durch Teilen der Summenwerte **SumKPmiddleX, Y** und **Z** mit **KPcount**, der Anzahl der summierten Punkte.

Erzeugen eines Punktes mit der Nummer **KPmx+i** und den ermittelten Koordinaten **KPmiddleX, Y** und **Z**.

Setze in die **KPcount+1**-te Spalte des Arrays **LINEcont8** die Nummer des soeben erzeugten Keypoints **KPmx+i**.

**\*ENDDO**

1.30.5 Erzeugen des letzten KPs der neutralen Faser (9. Linie)

Selbiger Vorgang wie in *Programmteil 1.30.4* nur, dass nicht mehr die **rownumber** aus **LINEcont8** – Ebene 5 eingelesen wird, sondern gleich die letzten Punkte der Konturlinien aus **LINEcont8(Vorteilung+1,...)** verwendet werden.

1.30.6 Erzeugen einer durchgehenden neutralen Faser (9. Linie) mit Hilfe der KPs

**\*DO**-Schleife für Position **1** bis zum Vorletzten Keypoint (**Vorteilung**)

Setze **P1** gleich dem **i**-ten Keypoint des Linienstücks  
(bei Beginn der Schleife gleich dem Startkeypoint)

Setze **P2** gleich dem **i+1**-ten Keypoint des Linienstücks  
(bei Ende der Schleife gleich dem Endkeypoint)

Verbinde beide Punkte **P1** und **P2** mit einer Linie.

**\*ENDDO**

1.30.7 Löschen aller Hilfsgeometrien -> Es bleiben 8 Konturlinien und eine durchgehende neutrale Faser

Ersten Keypoint der ersten Linie auswählen

**\*DO**-Schleife für Position **2** bis zum **Ende (Vorteilung+1)**

**\*DO**-Schleife für alle Konturlinienzüge

Weitere Keypoints der Konturlinien auswählen  
(Keypoints durch Vorteilung der Linien)

**\*ENDDO**

**\*ENDDO**

Alle Linien welche die ausgewählten Punkte enthalten auswählen.

Abwählen der Komponente LINEcomp, das sind die üblicherweise 8 durchgehenden Konturlinien.

Löschen aller noch ausgewählten Linien.

! -----

Ersten Keypoint der neutralen Faser auswählen

**\*DO**-Schleife für Position **2** bis zum **Ende (Vorteilung+1)**

**\*DO**-Schleife für alle Konturlinienzüge

Weitere Keypoints der neutralen Faser  
auswählen

**\*ENDDO**

**\*ENDDO**

Alle Linien welche die ausgewählten Punkte enthalten  
auswählen. *(Das sind die Linienabschnitte der neutralen Faser)*

**LCOMB**, kombiniert die ausgewählten Linienstücke zu einer  
durchgehenden Linie.

! -----

Alle Keypoints auswählen.

**\*DO**-Schleife für alle Konturlinienzüge und die neutrale Faser.

Abwählen der ersten und letzten Keypoints der nun  
9 Linien.

**\*ENDDO**

Alle ausgewählten Keypoints löschen

### 1.30.8 Erzeugen eines Arrays zur Speicherung aller zur Erzeugung der Koordinatensysteme notwendigen Parameter

Aktivieren von **CSYS, 15**.

Erzeugen eines lokalen, kartesischen Koordinatensystems mit  
der Nummer **101** an Stelle des **CSYS, 15**.

Definieren eines Arrays **MESHcoord** mit sovielen Zeilen wie der  
fertig vernetzte Spulenteil Elemente in der Länge haben soll und  
7 Spalten.

Setze die ersten Werte (*Erste Zeile, Index 1*) die der  
Startkeypoints und die letzten Werte (*Letzte Zeile, Index  
Teilung+1*):

Spalte 1: Keypointnummer  
 Spalte 2: reserviert für richtungsweisenden Keypoint  
 Spalte 3, 4 und 5: Drehungen der Koordinatensyst. um X, Y und Z  
 Spalte 6: Nummer des Hilfskoordinatensyst. (*erstes ist CSYS, 101*)  
 Spalte 7: reserviert für die Nummer des endgültigen KS

### 1.30.9 Aufteilen der neutralen Faser in „Teilung“ Teilstücke

Alle Keypoints auswählen und die größte existierende Keypointnummer in **KPmx** speichern.

Auswählen des ersten Keypoints der neutralen Faser.  
 Dazugehörige Linie (neutrale Faser) auswählen.

Schreibe in die Variable **neutralFibre** die Liniennummer.  
 Schreibe in **neutralFibre\_KP1** den ersten Keypoint der **neutralFibre**.

**\*IF**-Bedingung, wenn **neutralFibre\_KP1** gleich der in **LINEcont8** Zeile **1** gespeicherten Keypointnummer des Startkeypoints ist, dann:

Schreibe in **neutralFibre\_KP2** den **KP2** der **neutralFibre**

**\*DO**-Schleife für alle Teilungsabschnitte bis zum Übergang der Teilungen: (ratio12)

**KL** erzeugt auf der Linie **neutralFibre** an Stelle der späteren Elementgrenzen einen Keypoint mit der Nummer **KPmx+1+2\*i**

Setze den Wert in **MESHcoord(i+1,1)** gleich der oben definierten Keypointnummer.

**KL** erzeugt auf der Linie **neutralFibre**, auf 60% der durchschnittlichen Elementlänge nach dem Teilungskeypoint einen weiteren Keypoint mit der Nummer **KPmx+2+2\*i**.

*Anmerkung: Dieser Keypoint gibt an in welche Richtung die Z-Achse des später gedrehten Koordinatensystems schauen muss, damit die Z-Achse möglichst in Stabrichtung schaut. Die 60% der Stablänge sind ein durch Versuche ermittelter*



*Wert, welcher in den meisten kritischen Fällen in denen Vernetzungsfehler sich häuften die besten Ergebnisse gebracht hat.*

Setze den Wert in **MESHcoord(i+1,2)** gleich der oben definierten Keypointnummer.  
(*richtungsweisender Keypoint*)

**\*ENDDO**

Dieselbe **\*DO**-Schleife wie oben nun auch für den Abschnitt nach dem Übergang der Teilungen

**\*ENDIF**

#### 1.30.10 Erzeugen von Hilfskoordinatensystemen bei welchen die Z-Achse tangential zur neutralen Faser liegt

**\*DO**-Schleife für alle Teilungsabschnitte (2. KP bis Vorletzter)

Aktivieren des **CSYS** am vorigen Teilungspunkt

Schreibe die Koordinaten des aktuellen Teilungspunktes in **Xorig, Yorig** und **Zorig** und die Koordinaten des richtungsweisenden Keypoints in **Xdir, Ydir** und **Zdir**.

Berechnen der Drehungen **THYZ** und **THZX** um die X- und Y- Achse durch Berechnen der Winkelabweichung vom Ursprung des Koordinatensystems (*orig*) zum richtungsweisenden Keypoints (*dir*) aus der Sicht vom vorigen schon gedrehtem Hilfskoordinatensystem.

Setze die die Drehungen **THYZ** und **THZX** in die Arrayspeicherplätze in Spalte 3 und 4 von **MESHcoord**.

Erzeugen eines lokalen, kartesischen Koordinatensystems mit der Nummer **100+i** mit Ursprung am Teilungspunkt (*orig*) und um **THYZ** und **THZX** gedreht.

Setze in **MESHcoord** Spalte 6 die Koordinatensystemnummer **100+i**.

*Anmerkung: Diese Hilfskoordinatensysteme sind zwar in Z-Richtung bereits ausgerichtet, jedoch um die Z-Achse noch verdreht. Das jeweils vorige Hilfskoordinatensystem dient beim Drehen des jeweils Nächsten als Sichtpunkt.*

**\*ENDDO**

#### 1.30.11 Definieren eines KS im Schnittquerschnitt am Nutende

Aktivieren des CSYS, 14 (*Teilungskoordinatensystem*)

Speichern der Koordinaten des letzten Punktes der neutralen Faser in **Xorig**, **Yorig** und **Zorig**.

Erzeugen eines lokalen, kartesischen Koordinatensystems mit der Nummer **101+Teilung** mit Ursprung am letzten Keypoint der neutr. Faser und derselben Ausrichtung wie das **CSYS, 14**.

#### 1.30.12 Löschen aller nicht mehr gebrauchten Linien

Alle Linien auswählen.

Abwählen der Komponente **LINEcomp**, sprich der 8 Konturlinien.

Löschen aller ausgewählten Linien.

#### 1.30.13 Teilen der Konturlinien an den Teilungsebenen der späteren Elemente

Erzeugen eines 3D-Arrays **MESHpoints** mit **Teilung+1** Zeilen für die Teilung des Stabes in der Länge (Index **1** = Start am Nutende, Index **Teilung+1** = Ende an der Teilungsebene), **KPcount+1** (*Konturlinien und neutrale Faser*) Spalten und **5** Ebenen.

**\*DO**-Schleife für alle Konturlinien

*Anmerkung: Die folgenden \*IF-Bedingungen teilen die ersten Keypoints, also die Startkeypoints und die dazugehörigen Konturlinien in 4 Quadranten ein.*

**\*IF**-Bedingung:

---

Wenn die **X**-Koordinate des Startpunktes am Nutende welche in **LINEcont8(1,j,2)** gespeichert ist **UND** die **Y**-Koordinate in **LINEcont8(1,j,3)** größer gleich **NULL** sind, dann:

Setze **MESHpoints(1,j,1)** gleich **LINEcont8(1,j,1)**  
Wähle den Startpunkt und die dazugehörige Konturlinie aus.

Setze die Variable **actline** gleich der Liniennummer der ausgewählten Konturlinie.  
Setze **MESHpoints(1,j,5)** gleich der **actline**  
Setze die Variable **actline\_KP1** gleich dem ersten Keypoint der **actline**.

**\*IF**-Bedingung:

wenn **actline\_KP1** gleich dem Startkeypoint der Konturlinie am Nutende **MESHpoints(1,j,1)** ist, dann:

Setze **actline\_KP2** gleich dem 2. Keypoint der Linie **actline**.

Setze den letzten Eintrag

**MESHpoints(Teilung+1,j,1)**, also den

Wert in der Zeile für den Keypoint an der Teilungsebene gleich **actline\_KP2**.

**\*ELSEIF** wenn **actline\_KP1** ungleich dem Startkeypoint der Konturlinie am Nutende **MESHpoints(1,j,1)** ist, dann:

Setze **actline\_KP2** gleich dem 1. Keypoint der Linie **actline**.

Setze den letzten Eintrag

**MESHpoints(Teilung+1,j,1)**, also den

Wert in der Zeile für den Keypoint an der Teilungsebene gleich **actline\_KP1**.

**\*ENDIF**

**\*ELSEIF .....**

*Anmerkung: Wichtig dabei ist, dass die Keypoints mit positiven X- und Y-Koordinaten in die erste Ebene des Arrays **MESHpoints** gespeichert werden.*

Die folgenden 3 **\*IF**-Bedingungen schreiben die den anderen Quadranten zuzuordnenden Keypoints in die entsprechenden Ebenen des Arrays **MESHpoints**. Am Ende der Bedingungen stehen die Keypoints geordnet nach Quadranten entsprechend in den Ebenen des Arrays.

**\*ENDIF**

**\*ENDDO**

#### 1.30.14 Definieren von Komponenten zur besseren Selektierbarkeit beim Herauslöschchen von Hilfsgeometrien

Alle Keypoints und Linien auswählen und in die Komponenten **KPcomp** und **LINEcomp** zusammenfassen.

#### 1.30.15 Erzeugen von Keypoints an den Elementteilungsebenen

**\*DO**-Schleife für alle Teilungsabschnitte (2. KP bis Vorletzter), Index **i**

Aktivieren von **CSYS, 100+i**, dem lokalen Koordinatensystem auf der neutralen Faser an den Teilungsebenen.  
Erzeugen einer Working Plane in der X-Y-Ebene des Koordinatensystems.

**\*DO**-Schleife für alle Konturlinienzüge, Index **j**

**\*DO**-Schleife für die 4 Quadranten, Index **l**

Alle Keypoints auswählen und die größte Keypointnummer in **KPmx** speichern.

**\*IF**-Bedingung:  
Wenn der Speicherplatz der entsprechenden Konturlinie und Quadranten in der vorigen Zeile ungleich NULL ist, dann:

Setze die Nummer für den nächsten neu erzeugten Keypoint

---

auf **KPmx+1** um Kollisionen bei der Nummerierung zu vermeiden.

Schneide die Linie aus **MESHpoints(i,j,5)** mit der zuletzt erzeugten Working Plane.

Alle Keypoints auswählen  
Die nun nach dem Schnitt größte Keypointnummer in **KPmx\_new** speichern.

Alle Keypoints mit den Nummern zwischen **KPmx** und **KPmx\_new**, also alle beim Schnitt mit der Working Plane neu erzeugten Keypoints auswählen.

Verschmelze alle Keypoints die näher als 5/10mm beisammen liegen und gib dem einen Keypoint die größte Nummer der verschmolzenen Keypoints.

**\*DO**-Schleife für alle neuen KPs  
(Index **n** = KP-Nummer)

Schreibe in **Zcoord** die Z-Koordinate des eben bearbeiteten Keypoints

**\*IF**-Bedingung:  
Wenn **Zcoord** vom Betrag kleiner als 6/10 mm ist, dann:

Setze  
**MESHpoints(i,j,l)**  
gleich dem Index **n**  
bzw. der  
Keypointnummer.

**\*ENDIF**

**\*ENDDO**

**\*ENDIF**

**\*ENDDO**

**\*ENDDO**

**\*ENDDO**

Alle Linien auswählen und löschen.

1.30.16 Berechnen des Neigungswinkels THXY für die Rotation der Hilfskoordinatensysteme um die Z-Achse

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Ende), Index **i**

Nullsetzen der Variablen **SumKPcontX\_1** und **2**, sowie **SumKPcontY\_1** und **2**.

Aktivieren des **CSYS, 100+i**

**\*DO**-Schleife für alle Konturlinienzüge, Index **j**

**\*DO**-Schleife für die ersten 2 Quadranten, Index **l**

**\*IF**-Bedingung:

Wenn Index **l = 1** **UND** **MESHpoints(i,j,l)** ungleich **NULL**, dann:

Schreibe in **KPcontX\_1** die X-Koordinate des entsprechenden Punktes auf der Konturlinie. Dieser ist wegen Index **l=1** im ersten Quadranten. Selbiges mit **KPcontY\_1** ausführen und in den Summenvariablen **SumKPcontX\_1** und **SumKPcontY\_1** aufsummieren.

**\*ELSEIF**

Wenn Index **l=2** **UND** **MESHpoints(i,j,l)** ungleich **NULL**, dann:

Schreibe in **KPcontX\_2** die X-Koordinate des entsprechenden Punktes auf der Konturlinie. Dieser ist wegen Index **I=2** im zweiten Quadranten. Selbiges mit **KPcontY\_2** ausführen und in den Summenvariablen **SumKPcontX\_2** und **SumKPcontY\_2** aufsummieren.

**\*ENDIF**

**\*ENDDO**

**\*ENDDO**

*Anmerkung: Um den Drehwinkel THXY zu berechnen müssen die Summenvariablen SumKPcontX\_1 und 2 sowie SumKPcontY\_1 und 2 nicht durch die Anzahl der Summanden dividiert werden da sich in jedem Quadranten gleich viele Konturlinien befinden und sich somit die Anzahl der Konturlinien im Bruch des Tangens herauskürzen würde.*

Der Drehwinkel um die Z-Achse THXY wird berechnet.

Erzeugen eines lokalen, kartesischen Koordinatensystems mit Nummer **1000+i** wird an Stelle des aktiven Koordinatensystems **CSYS, 100+i** erzeugt welches darauf gelöscht wird.

**\*ENDDO**

### 1.30.17 Erzeugen von KP an den Eckpunkten der späteren Elemente

Alle Keypoints auswählen und die größte Keypointnummer in **KPmx** speichern.

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Ende), Index **i**

Aktivieren des fertig gedrehten **CSYS, 1000+i** (Z-Achse tangential zur Stabachse bzw. der neutralen Faser und X-

und Y-Achse so gedreht, dass die Achsen mittig zwischen den Konturlinien verlaufen)

Definieren und Nullsetzen der Variablen **KPcontmaxX\_1** bis **4**

Und **KPcontmaxY\_1** bis **4**. Die Zahl nach dem Unterstrich steht für den Quadranten in welchem sich die Koordinate befindet.

**\*DO**-Schleife für alle Konturlinienzüge, Index **j**

**\*DO**-Schleife für die 4 Quadranten, Index **l**

**\*IF**-Bedingung:

Wenn die der Index **l = 1** **UND** der Eintrag in **MESHpoints(i,j,l)** ungleich **NULL** ist, also der angewählte Keypoint im **1.** Quadranten ist, dann:

Schreibe seine **X** und **Y**-Koordinaten in **KPcontX\_1** und **KPcontY\_1**.

**\*IF**-Bedingung:

Wenn **KPcontX\_1** größer als **KPcontmaxX\_1** ist, dann:

Setze **KPcontmaxX\_1** gleich **KPcontX\_1**.

**\*ENDIF**

**\*IF**-Bedingung:

Wenn **KPcontY\_1** größer als **KPcontmaxY\_1** ist, dann:

Setze **KPcontmaxY\_1** gleich **KPcontY\_1**.

**\*ENDIF**

**\*ELSEIF**.....

*Anmerkung: In den Variablen **KPcontmaxX\_1** und*



***KPcontmaxY\_1** stehen nun die maximalen Koordinaten aller Keypoints aus diesem Quadranten, also ein Eckpunkt, anstatt mehrerer Keypoints pro Quadrant ist gefunden. Die nachfolgenden 3 **\*ELSEIF**-Bedingungen beschreiben selbigen Vorgang auch für die Quadrant **2,3** und **4**.*

**\*ENDIF**

**\*ENDDO**

**\*ENDDO**

Dazu zählen der Isolationsschichtdicke **Isolation\_H** zu **KPcontmaxX\_1** und speichern des Wertes in **KPcoordX\_1**.

Selbiges mit **Isolation\_B** und Koordinatenrichtung **Y** ausführen.

Erzeugen eines Keypoints an der Ecke der Geometrie inkl. Isolierung mit der Nummer **KPmx+1000+off\_OS\_US+i**. Die Zahl **1000** in der Nummerierung ist schon im Ausblick auf die Knotennummerierung gewählt.

Selbiges Prozedere für:

Quadrant 2 -> **7000**er Nummern

Quadrant 3 -> **5000**er Nummern

Quadrant 4 -> **3000**er Nummern

Erzeugen eines Keypoints mit der Nummer **KPmx+9000+off\_OS\_US+i** am Ursprung (lokale Koordinaten alle NULL). Um diesen Punkt wird später in jedem Quadranten ein Element angeordnet.

**\*ENDDO**

### 1.30.18 Rotieren der Nodes, damit der Stab in Nutrichtung ausgerichtet ist

Aktivieren von **CSYS, 12**

---

Alle Keypoints auswählen und die größte vorkommende Keypointnummer in **KPmx\_new** speichern.

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Ende), Index **i**

Erzeugen von Keypoints an den vier Eckpunkten des Querschnitts und auf der neutralen Faser, welche um den Winkel **slotangle** gedreht sind und um **KPmx\_new-KPmx** höhere Nummern haben.

**\*ENDDO**

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Ende), Index **i**

Erzeugen von Nodes an Stelle der soeben gedrehten Keypoints, die Nummerierung der Nodes ist gleich der Keypointnummern nur um **KPmx\_new** kleiner.

**\*ENDDO**

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Ende), Index **i**

Erzeugen von Nodes zwischen den Ecken des Querschnitts mit **FILL**. Mittig zwischen den 1000er und 3000er Nodes werden z.B. Nodes mit 2000er Nummern erzeugt, usw.

**\*ENDDO**

Alle Keypoints und Linien auswählen und Löschen.  
Die Geometrie besteht nun nur mehr aus Nodes.

### 1.30.19 Erzeugen von Elementen

Auswählen von Material 1.  
Auswählen von Type 1.

Starten der Elementnummerierung bei **1+off\_OS\_US**

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Vorletzten),  
Index **i**

Erzeugen eines Elementes zwischen den Nodes im  
1. Quadranten.

**\*ENDDO**

Starten der Elementnummerierung bei **101+off\_OS\_US**

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Vorletzten),  
Index **i**

Erzeugen eines Elementes zwischen den Nodes im  
2. Quadranten.

**\*ENDDO**

Starten der Elementnummerierung bei **201+off\_OS\_US**

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Vorletzten),  
Index **i**

Erzeugen eines Elementes zwischen den Nodes im  
3. Quadranten.

**\*ENDDO**

Starten der Elementnummerierung bei **301+off\_OS\_US**

**\*DO**-Schleife für alle Teilungsabschnitte (Start bis Vorletzten),  
Index **i**

Erzeugen eines Elementes zwischen den Nodes im  
4. Quadranten.

**\*ENDDO**

1.30.20 Erzeugen durchgehenden Linien und Knoten  
am aufgedickten Stab an den Eckpunkte  
(übernommen aus Diss. DI Bern Schlegl)

**\*IF**-Bedingung:

Wenn **off\_OS\_US** ungleich **NULL** ist  
(also beim 2. Programmaufruf), dann:

Alles Auswählen.

---

Alle Linien löschen.  
Alle Keypoints löschen.

BOPTN stellt ein, dass keine Warnung ausgegeben wird,  
falls eine Boolesche Operation keine Auswirkung hat.

**\*DO**-Schleife für alle **9** Linien (Index **k**)

**\*DO**-Schleife für alle Teilungsabschnitte  
(Index **i** = Nodenummer, Start bis Ende)

Erzeugen von Keypoints mit den selben  
Nummer und an derselben Stelle wie die  
Nodes am Oberstab bzw. Oberspule.

**\*ENDDO**

**\*DO**-Schleife für alle Teilungsabschnitte  
(Index **i** = Nodenummer, 2. Node bis Ende)

**NUMSTR** setzt fest, dass die  
Liniennummern mit **i-1** beginnen.  
Erzeugen einer Linie zwischen den oben  
erzeugten Keypoints am Oberstab bzw.  
Oberspule.

**\*ENDDO**

**\*DO**-Schleife für alle Teilungsabschnitte  
(Index **i** = Nodenummer, Start bis Ende)

Erzeugen von Keypoints mit den selben  
Nummer und an derselben Stelle wie die  
Nodes am Unterstab bzw. Unterspule.

**\*ENDDO**

**\*DO**-Schleife für alle Teilungsabschnitte  
(Index **i** = Nodenummer, 2. Node bis Ende)

**NUMSTR** setzt fest, dass die  
Liniennummern mit **i-1** beginnen.  
Erzeugen einer Linie zwischen den oben  
erzeugten Keypoints am Unterstab bzw.  
Unterspule.

**\*ENDDO**

**NUMSTR** setzt fest, dass die Liniennummern mit **1000\*k** beginnen.

Alle Linien mit Nummern zwischen **1000\*k** und **1000\*k+Teilung** auswählen.

Die ausgewählten Linienstücke zu einer durchgehenden Linie zusammenfassen.  
(Oberstab)

Selbiges Vorgehen am Unterstab.

**\*ENDDO**

Aktivieren von **CSYS, 11**

Erzeugen eines Keypoints mit der Nummer **1** am Ursprung von **CSYS, 11** und einen mit Nummer **2** 100 Einheiten in Z-Richtung verschoben.

Keypoints **3, 4, 5** und **6** spannen eine Fläche am Nutende normal zur Z-Achse auf.

**BOPTN, KEEP, YES** bewirkt, dass die eingegebenen Entitäten nicht gelöscht werden.

Erzeugen eines Keypoints mit Nummer **7** an der Generator Drehachse um Nutende in Z-Richtung verschoben.

Erzeugen eines kartesischen Koordinatensystems **CSYS, 17** welches durch die Keypoints **7, 9001+off\_OS\_US** und **2** aufgespannt wird.

*Anmerkung: Keypoint 7 ist der Ursprung, 9001+off\_OS\_US gibt die Richtung der X-Achse an und Keypoint 2 definiert die Lage der X-Y-Ebene.*

Erzeugen von **CSYS, 18**, einem kartesischen Koordinatensystem, welches um  $-90^\circ$  um die Z-Achse gedreht ist und von **CSYS, 19** einem zylindrischen Koordinatensystem auf Basis **CSYS, 18**.

Erzeugen von Arrays **Kp\_innen\_1** und **2** sowie  
**Kp\_aussen\_1** und **2** und **Kp\_El**.

Definition und Nullsetzen der Variablen **i\_Stuetzring\_a** ,  
**i\_Stuetzring\_i**, **i\_Distanzring**, **i\_Zwischenpackerl** und  
**i\_Zwischenpackerl\_i**.

**\*ENDIF**

**LPLOT** und **ELOT** stellen die Geometrie (Linien und Elemente) dar.

## 8.2 Programmbeschreibung „Gewölbekeile“

### 8.2.1 Variablendeklaration

Variablenname	Beschreibung
<b>NR</b>	Definition der Hunderterstelle der Knotennummerierung durch eine Zahl z.B.: 900
<b>Zpos</b>	Z-Position der Working Plane, an welcher die Gewölbekeilknoten erzeugt werden
<b>KPmx</b>	Größte vorkommende Keypointnummer in der Auswahl
<b>KPmn</b>	Kleinste vorkommende Keypointnummer in der Auswahl
<b>Zcoord</b>	Z-Koordinate eines bestimmten Keypoints
<b>NR_act</b>	Knotenbasisnummer für die aktuell bearbeitete Nut
<b>NR_mid</b>	Knotenbasisnummer für die mittleren Knoten zwischen den Nuten
<b>NR_next</b>	Knotenbasisnummer für die Knoten am Spulenteil der darauffolgenden Nut
<b>X1</b>	X-Koordinaten der Gewölbekeilknoten
<b>Y1</b>	Y-Koordinaten der Gewölbekeilknoten
<b>Z1</b>	Z-Koordinaten der Gewölbekeilknoten
<b>X2</b>	X-Koordinaten der Spulenteilknoten
<b>Y2</b>	Y-Koordinaten der Spulenteilknoten
<b>Z2</b>	Z-Koordinaten der Spulenteilknoten
<b>mindistance</b>	Minimaler Abstand zwischen zwei eventuell zu koppelnden Knoten
<b>break</b>	Abbruchvariable (1...Abbruch)
<b>actdistance</b>	Aktueller Abstand zwischen den bearbeiteten Spulenteilknoten und Gewölbekeilknoten
<b>GK_node</b>	Knotennummer des aktuell zu koppelnden Knoten am Gewölbekeil
<b>SP_node</b>	Knotennummer des aktuell zu koppelnden Knoten am Spulenteil

### 8.2.2 Arrays

Variablenname	Beschreibung
<b>Connect(i,j,k)</b>	3D-Array zum Suchen der nächstgelegenen Knoten um diese mit <b>Correlation(i,j,k)</b> zu koppeln. (i = Teilung der Spulenteile, j = Knotenposition nach Nummerierungsschema der Gewölbekeile, k = 1...Nummer, 2...X-Koordinate, 3...Y-Koordinate, 4...Z-Koordinate)
<b>Correlation(i,j,k)</b>	3D-Array zum Koppeln der nächstgelegenen Knoten zwischen Gewölbekeilen und Spulenteilen. (i = Knotenposition nach Nummerierungsschema der Gewölbekeile, j = Position der Hilfsflächen bzw. Working Planes an den Gewölbekeilen, k = 1...Gewölbekeilknotennummer, 2...dazugehörige nächstgelegene Spulenteilknotennummer)

Folgende Beschreibung und die Nummerierung der Kapitel bezieht sich auf den Programmcode des Unterprogramms „Gewoelbekeile“.

### 2.10 Programmbaustein Gewoelbekeile

Definition der Hunderterstelle der Nummerierung mit der Variable **NR**.

### 2.20 Erzeugen von Knoten an den Leitern zum Einhaengen der Elemente fuer die Gewoelbekeile

Auswaehlen von **CSYS,11** mit Z-Achse auf der positiven Seite der Drehachse und Y-Achse in positiv radialer Richtung.

Alles Auswählen

Material Nummer **MAT,8** und Elementtype **TYPE,1** festlegen

**\*DO**-Schleife für die Hilfsflächen (Working Planes) an denen später die Knoten der Gewölbekeile erzeugt werden. (**i**...0,1,2)

Working Plane in die X-Y-Ebene des CSYS,11 legen

Abhängig von der Schleifenvariable **i** die Position in Z-Richtung **Zpos** der aktuellen Working Plane berechnen.

Mit **WPOFFS** die Position der Working Plane in Z-Richtung um **Zpos** verschieben.

**\*DO**-Schleife für die Unterscheidung zwischen Ober- und Unterspule (**j**...0,1 -> 0....Oberspule, 1...Unterspule)

! -----Die 3 Linien durcharbeiten welche pro niedriger nummerierten Spulenteil (NUT) an die Gewölbekeile anschließen.-----

**\*DO**-Schleife (**k**....1,2,3)

Alle Linien auswählen.

Alle Linien in die Komponente **LINEstatold** zusammenfassen.

Alle Keypoints auswählen.

Alle Keypoints in die Komponente **KPstatold** zusammenfassen.



Mit **\*GET** die größte vorkommende Keypointnummer in **KPmx** speichern.

Mit **NUMSTR** die Nummerierung der neuen Keypoints bei **KPmx** starten.

**LSBW** schneidet die Linien mit der aktuellen Working Plane und erzeugt Keypoints an den Schnittstellen.

*Anmerkung: Die Erzeugung und Nummerierung der Linien wurde in den letzten Kapiteln des Programms Spulenmodell durchgeführt. Durch den Befehl LSBW werden nicht nur Keypoints an den Schnittstellen zwischen Linien und Working Plane erzeugt, sondern auch an diversen anderen Punkten an dieser Linie, weshalb das so ist konnte auch der Support bei CADFEM nicht beantworten. Jedenfalls ist es danach notwendig, den gesuchten Punkt unter allen neu erzeugten Punkten herauszusuchen, was in den folgenden Zeilen gemacht wird.*

Alle Keypoints auswählen und davon die vorhin definierte Komponente **KPstatold** abwählen. Es bleiben die neu erzeugten Keypoints übrig.

Mit **\*GET** werden in **KPmn**, die kleinste und in **KPmx**, die größte Nummer der neuen KP abgespeichert.

**\*DO**-Schleife für alle neuen KP mit Nummern zwischen **KPmn** und **KPmx** ( $n \dots KPmn - KPmx$ )

Mit **\*GET** die Z-Position des aktuellen Keypoints in **Zcoord** speichern.

**\*IF**-Bedingung, wenn der Betrag der Differenz zwischen **Zpos** und **Zcoord** absolut kleiner als der Schwellwert von 5/100mm ist, also tatsächlich an der Working Plane liegt, dann:

Erzeuge mit **NKPT** einen Knoten mit der Nummer **NR+(i\*20)-**

---

**(j\*3)+k+3** an Stelle dieses  
Keypoints.

**\*ENDIF**

**\*ENDDO**

Alle Linien auswählen.  
Davon die Komponente **LINEstatold** abwählen  
Die Übriggebliebenen Linien löschen.

Alle Keypoints auswählen.  
Davon die Komponente **KPstatold** abwählen.  
Die Übriggebliebenen Keypoints löschen.

**\*ENDDO**

! -----Die 3 Linien durcharbeiten welche pro höher  
nummerierten Spulenteil (NUT) an die Gewölbekeile  
anschießen.-----

**\*DO**-Schleife (**k**....5,6,7)

Alle Linien auswählen.  
Alle Linien in die Komponente **LINEstatold**  
zusammenfassen.

Alle Keypoints auswählen.  
Alle Keypoints in die Komponente **KPstatold**  
zusammenfassen.

Mit **\*GET** die größte vorkommende  
Keypointnummer in **KPmx** speichern.

Mit **NUMSTR** die Nummerierung der neuen  
Keypoints bei **KPmx** starten.

**LSBW** schneidet die Linien mit der aktuellen  
Working Plane und erzeugt Keypoints an den  
Schnittstellen.

*Anmerkung: Die Erzeugung und Nummerierung  
der Linien wurde in den letzten Kapiteln des  
Programms Spulenmodell durchgeführt. Durch den  
Befehl LSBW werden nicht nur Keypoints an den  
Schnittstellen zwischen Linien und Working Plane*

*erzeugt, sondern auch an diversen anderen Punkten an dieser Linie, weshalb das so ist konnte auch der Support bei CADFEM nicht beantworten. Jedenfalls ist es danach notwendig, den gesuchten Punkt unter allen neu erzeugten Punkten herauszusuchen, was in den folgenden Zeilen gemacht wird.*

Alle Keypoints auswählen und davon die vorhin definierte Komponente **KPstatold** abwählen. Es bleiben die neu erzeugten Keypoints übrig.

Mit **\*GET** werden in **KPmn**, die kleinste und in **KPmx**, die größte Nummer der neuen KP abgespeichert.

**\*DO**-Schleife für alle neuen KP mit Nummern zwischen **KPmn** und **KPmx** (**n... KPmn - KPmx**)

Mit **\*GET** die Z-Position des aktuellen Keypoints in **Zcoord** speichern.

**\*IF**-Bedingung, wenn der Betrag der Differenz zwischen **Zpos** und **Zcoord** absolut kleiner als der Schwellwert von 5/100mm ist, also tatsächlich an der Working Plane liegt, dann:

Erzeuge mit **NKPT** einen Knoten mit der Nummer **NR+(i\*20)-(j\*3)+k+23** an Stelle dieses Keypoints.

**\*ENDIF**

**\*ENDDO**

Alle Linien auswählen.  
Davon die Komponente **LINEstatold** abwählen  
Die Übriggebliebenen Linien löschen.

Alle Keypoints auswählen.  
Davon die Komponente **KPstatold** abwählen.  
Die Übriggebliebenen Keypoints löschen.

**\*ENDDO**

**\*ENDDO**

**\*ENDDO**

### 2.30 Rotationssymmetrisches Mustern der Anbindungsknoten

Auswählen von **CSYS,12** (zylindrisches KS anstelle von **CSYS,11**)

Alle Knoten, welche in den letzten drei Stellen mit **NR** bis **NR+99** nummeriert sind, auswählen.

Generiere mit **NGEN** für jede Statornut die eben erzeugten Knoten und nummeriere sie wie die Spulenteile mit **off\_Stb\*n\_Nuten**.

### 2.40 Erzeugen von mittleren Knoten zwischen den bisherigen Knoten

**\*DO**-Schleife für die Hilfsflächen bzw. Working Planes (**i...0,1,2**)

**\*DO**-Schleife für alle Nuten (**m...0-n\_Nuten**)

*Anmerkung: Die Variable m beschreibt die aktuelle Nut und deren Spulenteile. Da die Gewölbekeile aber jeweils zwischen zwei Nuten liegen muss auch die darauffolgende Nut betrachtet werden. Zur einfacheren Handhabung wird für die nächste Nut die Variable n definiert. Um das Prozedere beim Sprung von der letzten auf die erste Nut zu erleichtern, wird folgende \*IF-Bedingung eingeschalten:*

**\*IF**-Bedingung, wenn **m = n\_Nuten-1** ist, also der Sprung zwischen letzter und erster Nut erreicht ist, dann

Setze **n = 0**.

**\*ELSE**

ist **n = m+1**

**\*ENDIF**

! -----Erzeugen von Knoten zwischen den bisherigen neuen Knoten.-----

**\*DO**-Schleife für die 6 Knoten pro Hilfsflächen bzw. Working Plane an denen die Gewölbekeile an die Spulen gekoppelt werden. (**p...1,2,3,4,5,6**)

**FILL**, setzt genau zwischen die ersten beiden eingegebenen Knoten einen dritten, mittleren Knoten.

*Anmerkung: Zwischen den Knoten der niedriger nummerierten und der höher nummerierten Nut werden noch Zwischenknoten gesetzt. Diese Knoten werden der höher nummerierten Nut (Nummer **n**) zugeordnet und haben eine um den Wert 6 höhere Nummer als der benachbarte Knoten der höher nummerierten Nut.*

**\*ENDDO**

**\*ENDDO**

**\*ENDDO**

## 2.50 Einhängen der Elemente zwischen den Knoten

**\*DO**-Schleife für die Hilfsflächen bzw. Working Planes (**i**...0,1,2)

**\*DO**-Schleife für alle Nuten (**m**...0-n\_Nuten)

**\*IF**-Bedingung, wenn **m = n\_Nuten-1** ist, also der Sprung zwischen letzter und erster Nut erreicht ist, dann

Setze **n = 0**.

**\*ELSE**

ist **n = m+1**

**\*ENDIF**

! -----Einhängen von Elementen zwischen den Knoten mit höherer Nutnummer **m** und den mittleren Knoten.-----

**\*DO**-Schleife für die 6 Knoten pro Hilfsflächen bzw. Working Plane an denen die Gewölbekeile an die Spulen gekoppelt werden. (**p**...1,2,3,4,5)

*Anmerkung: **p** muss nur bis 5 laufen, da vom Vorletzten Knoten aus die letzten Elemente eingehängt werden.*

**NR\_act** und **NR\_mid** sind Knotenbasisnummern die die Schreibweise bei der Elementerzeugung übersichtlicher macht.

*Anmerkung: **NR\_act** steht für den achsferneren Knoten der niedriger nummerierten Nut sowie der*

niedriger nummerierten Hilfsfläche bzw. Working Plane. **NR\_mid** steht für den mittleren Knoten neben **NR\_act**. Um nun die Knotennummer eines Knotens der nächsten achsnäheren Knotenreihe anzusprechen muss lediglich **1** dazugezählt werden, für den Knoten an der nächsten Hilfsfläche bzw. Working Plane der Wert **20**.

**E**, hängt das Element zwischen die Knoten.

Anmerkung: **E** benötigt in definierter Reihenfolge die Knotennummern der 8 Eckpunkte des zwischenliegenden Elements. Die Elementkoordinatenrichtung muss in X-Richtung tangential zum Generator ausgerichtet sein. Die Richtlinien zur Eingabe sind in Kapitel 3.1.2 angeführt.

**\*ENDDO**

! -----Einhängen von Elementen zwischen den mittleren Knoten und den Knoten mit kleinerer Nutnummer m.-----

**\*DO**-Schleife für die 6 Knoten pro Hilfsflächen bzw. Working Plane an denen die Gewölbekeile an die Spulen gekoppelt werden. (**p**....1,2,3,4,5)

Anmerkung: Diese Schleife erzeugt die Elemente an der Nut mit kleinerer Nutnummer. Zwischen den mittleren Knoten und den Knoten an den Spulenteilen ist ein Nummerierungsoffset von **off\_Stb**.

**\*ENDDO**

**\*ENDDO**

**\*ENDDO**

## 2.60 Knoten der Gewölbekeile mit den nächstgelegenen Knoten an der Spule koppeln

Anmerkung: Die Kopplung der Knoten bringt unnatürliche Versteifungen ins System. Die einzige Möglichkeit dem entgegenzuwirken ist eine sehr feine Teilung der Leitergeometrie im Bereich der Gewölbekeile.

**CSYS,11** auswählen

Erzeugen eines 3D-Arrays „**Connect**(Teilung,18,4)“ und eines 3D-Arrays „**Correlation**(18,3,2)“.

Anmerkung: Obwohl nur 12 Kopplungspunkte pro Hilfsfläche bzw. Working Plane vorkommen, werden in den Arrays 18 Speicherplätze

*freigehalten. Die Speicherplatznummer kann somit direkt mit der Knotennummer in Verbindung gebracht werden. Folglich sind nur die Speicherplätze 1-6 und 13-18 belegt.*

!----- Vollschieben der Arrays mit den schon bekannten Knotennummern und Werten -----

**\*DO**-Schleife für alle Knoten an den Spulenteilen (**i**...1 - Teilung)

Setzen aller theoretisch möglichen Anschlussknotennummern der Spulenteile in das Array **Connect**.

**\*DO**-Schleife für alle Knoten am höher nummerierten Spulenteil (**j**...1,2,3,4,5,6)

Setzen der X-, Y- und Z-Koordinate in die dahinterliegenden Ebenen des Arrays.

**\*DO**-Schleife für die Gewölbekeilknoten (**k**...1,2,3)

Setzen der 3 Gewölbekeilknotennummern, welche an eine Spulenteilkonturlinie angrenzen, in das Array **Correlation**

**\*ENDDO**

**\*ENDDO**

**\*DO**-Schleife für alle Knoten am niedriger nummerierten Spulenteil (**j**...13,14,15,16,17,18)

Setzen der X-, Y- und Z-Koordinate in die dahinterliegenden Ebenen des Arrays.

**\*DO**-Schleife für die Gewölbekeilknoten (**k**...1,2,3)

Setzen der 3 Gewölbekeilknotennummern, welche an eine Spulenteilkonturlinie angrenzen, in das Array **Correlation**

**\*ENDDO**

**\*ENDDO**

**\*ENDDO**

!----- Suchen von Knoten auf der Geometrie, welche minimalen Abstand zu jenen der Gewölbekeile haben -----

**\*DO**-Schleife für die 6 Knoten an der höher nummerierten Nut. (**j**...1,2,3,4,5,6)

**\*DO**-Schleife für die 3 Hilfsflächen bzw. Working Planes  
(**k**...1,2,3)

Schreibe mit **\*GET** die Knotenkoordinaten der  
Gewölbekeilknoten in **X1**, **Y1** und **Z1**

Nullsetzen der Variablen **mindistance** und **break**.

**\*DO**-Schleife für alle Teilungsknoten an den Spulenteilen  
(**i**...1 – Teilung)

Schreibe mit **\*SET** die Knotenkoordinaten der  
Spulenteilknoten vom Array „**Connect**“ in **X2**, **Y2**  
und **Z2**.

Berechnen der aktuellen Distanz **actdistance**  
zwischen Gewölbekeilknoten und  
Spulenteilknoten.

**\*IF**-Bedingung:

Wenn beim ersten Durchlauf der Schleife  
**mindistance** = 0 ist,  
dann überschreibe **mindistance** mit  
**actdistance**.

Oder wenn die aktuell berechnete Distanz  
**actdistance** zwischen zwei Keypoints  
kleiner oder gleich der **mindistance** ist,  
dann überschreibe **mindistance** mit  
**actdistance** und setze **Correlation(j,k,2)**  
gleich der Knotennummer aus  
**Connect(i,j,1)**. Die Abbruchvariable  
**break** wird 1 gesetzt.

*Anmerkung: Sollte beim nächsten  
Durchlauf der innersten \*DO-Schleife ein  
noch kleinerer Abstand gefunden werden,  
so wird sowohl die **mindistance** als auch  
der Speicherplatz in **Correlation(j,k,2)** neu  
überschrieben.*

Oder wenn die aktuell berechnete Distanz  
**actdistance** zwischen zwei Keypoints  
größer der **mindistance** ist und schon  
vorher die vorherige Bedingung erfüllt  
wurde, sprich **break** schon **1** ist, dann



---

verlasse die innerste **\*DO**-Schleife mit  
**\*EXIT**.

**\*ENDIF**

**\*ENDDO**

**\*ENDDO**

**\*ENDDO**

**\*DO**-Schleife für die 6 Knoten an der niedriger nummerierten Nut.  
(j....13,14,15,16,17,18)

**\*DO**-Schleife für die 3 Hilfsflächen bzw. Working Planes  
(k....1,2,3)

Schreibe mit **\*GET** die Knotenkoordinaten der  
Gewölbekeilknoten in **X1, Y1** und **Z1**

Nullsetzen der Variablen **mindistance** und **break**.

**\*DO**-Schleife für alle Teilungsknoten an den Spulenteilen  
(i....1 – Teilung)

Schreibe mit **\*SET** die Knotenkoordinaten der  
Spulenteilknoten vom Array „**Connect**“ in **X2, Y2**  
und **Z2**.

Berechnen der aktuellen Distanz **actdistance**  
zwischen Gewölbekeilknoten und  
Spulenteilknoten.

**\*IF**-Bedingung:

Wenn beim ersten Durchlauf der Schleife  
**mindistance** = 0 ist,  
dann überschreibe **mindistance** mit  
**actdistance**.

Oder wenn die aktuell berechnete Distanz  
**actdistance** zwischen zwei Keypoints  
kleiner oder gleich der **mindistance** ist,  
dann überschreibe **mindistance** mit  
**actdistance** und setze **Correlation(j,k,2)**  
gleich der Knotennummer aus  
**Connect(i,j,1)**. Die Abbruchvariable  
**break** wird 1 gesetzt.

*Anmerkung:* Sollte beim nächsten Durchlauf der innersten **\*DO**-Schleife ein noch kleinerer Abstand gefunden werden, so wird sowohl die **mindistance** als auch der Speicherplatz in **Correlation(j,k,2)** neu überschrieben.

Oder wenn die aktuell berechnete Distanz **actdistance** zwischen zwei Keypoints größer der **mindistance** ist und schon vorher die vorherige Bedingung erfüllt wurde, sprich **break** schon **1** ist, dann verlasse die innerste **\*DO**-Schleife mit **\*EXIT**.

**\*ENDIF**

**\*ENDDO**

**\*ENDDO**

**\*ENDDO**

!----- Kopplung der Knoten-----

Mit **ALLSEL** alles Auswählen  
Alle Elemente plotten

**\*DO**-Schleife für alle Nuten (**n**....0 – **n**\_Nuten)

**\*DO**-Schleife für die 6 Knoten an der höher nummerierten Nut.  
(**j**....1,2,3,4,5,6)

**\*DO**-Schleife für die 3 Hilfsflächen bzw. Working Planes  
(**k**....1,2,3)

Schreibe mit **\*SET** die Knotennummer des zu koppelnden Gewölbekeilknoten von „**Correlation(j,k,1)** und dessen rotationssymmetrische Äquivalente nacheinander in **GK\_node** und die dazugehörigen Spulenteilknoten von „**Correlation(j,k,2)** und dessen rotationssymmetrische Äquivalente nacheinander in **SP\_node**.

Koppeln der jeweils zwei zusammengehörigen Knoten **GK\_node** und **SP\_node** in allen 3 Raumrichtungen und um alle 3 Drehfreiheitsgrade mit dem Befehl **CP**

**\*ENDDO**

**\*ENDDO**

**\*DO**-Schleife für die 6 Knoten an der niedriger nummerierten Nut. (**j**....13,14,15,16,17,18)

**\*DO**-Schleife für die 3 Hilfsflächen bzw. Working Planes (**k**....1,2,3)

Schreibe mit **\*SET** die Knotennummer des zu koppelnden Gewölbekeilknoten von „**Correlation**(j,k,1) und dessen rotationssymmetrische Äquivalente nacheinander in **GK\_node** und die dazugehörigen Spulenteilknoten von „**Correlation**(j,k,2) und dessen rotationssymmetrische Äquivalente nacheinander in **SP\_node**.

Koppeln der jeweils zwei zusammengehörigen Knoten **GK\_node** und **SP\_node** in allen 3 Raumrichtungen und um alle 3 Drehfreiheitsgrade mit dem Befehl **CP**

**\*ENDDO**

**\*ENDDO**

**\*ENDDO**

Alles auswählen  
Alle Elemente plotten.



Vergleich der Federstiften und der Zwischenschichten (layerschichten)

	Zwischenschichten weit bei 94,5				Zwischenschichten 10x steifer				Zwischenschichten 100x steifer				Zwischenschichten 1000x steifer			
	SET	TIME/FREQ.	MODE	ABW in [%]	SET	TIME/FREQ.	MODE	ABW in [%]	SET	TIME/FREQ.	MODE	ABW in [%]	SET	TIME/FREQ.	MODE	ABW in [%]
Federstift/- weihen mit bei 94,5	1	107,83	1	-35,04	1	113,30	1	-30,13	1	127,27	1	-18,50	1	188,13	1	-5,00
	2	107,83	2	-37,24	2	113,30	2	-33,70	2	127,27	2	-20,17	2	188,13	2	-6,17
	3	110,35	3	-37,24	3	113,30	3	-33,70	3	127,27	3	-20,17	3	188,13	3	-6,17
	4	110,35	4	-37,24	4	113,30	4	-33,70	4	127,27	4	-20,17	4	188,13	4	-6,17
	5	117,71	5	-37,24	5	113,30	5	-33,70	5	127,27	5	-20,17	5	188,13	5	-6,17
	6	117,71	6	-37,24	6	113,30	6	-33,70	6	127,27	6	-20,17	6	188,13	6	-6,17
	7	117,71	7	-37,24	7	113,30	7	-33,70	7	127,27	7	-20,17	7	188,13	7	-6,17
	8	117,71	8	-37,24	8	113,30	8	-33,70	8	127,27	8	-20,17	8	188,13	8	-6,17
	9	117,71	9	-37,24	9	113,30	9	-33,70	9	127,27	9	-20,17	9	188,13	9	-6,17
	10	117,71	10	-37,24	10	113,30	10	-33,70	10	127,27	10	-20,17	10	188,13	10	-6,17
Federstift/- weihen 20x steifer	1	113,91	1	-31,38	1	130,99	1	-35,99	1	160,30	1	-9,32	1	184,38	1	4,17
	2	113,91	2	-34,15	2	130,99	2	-39,04	2	160,30	2	-12,11	2	184,38	2	-1,87
	3	116,56	3	-34,15	3	130,99	3	-39,04	3	160,30	3	-12,11	3	184,38	3	-1,87
	4	116,56	4	-34,15	4	130,99	4	-39,04	4	160,30	4	-12,11	4	184,38	4	-1,87
	5	121,96	5	-34,15	5	130,99	5	-39,04	5	160,30	5	-12,11	5	184,38	5	-1,87
	6	121,96	6	-34,15	6	130,99	6	-39,04	6	160,30	6	-12,11	6	184,38	6	-1,87
	7	131,92	7	-38,76	7	149,09	7	-41,92	7	179,99	7	-17,91	7	207,95	7	23,27
	8	131,92	8	-38,76	8	149,09	8	-41,92	8	179,99	8	-17,91	8	207,95	8	23,27
	9	140,24	9	-44,23	9	158,47	9	-45,90	9	198,89	9	-22,31	9	219,90	9	-14,26
	10	140,24	10	-44,23	10	158,47	10	-45,90	10	198,89	10	-22,31	10	219,90	10	-14,26
Federstift/- weihen 100x steifer	1	142,77	1	-46,39	1	163,89	1	-37,44	1	204,73	1	-13,88	1	243,12	1	-13,88
	2	142,77	2	-46,39	2	163,89	2	-37,44	2	204,73	2	-13,88	2	243,12	2	-13,88
	3	142,77	3	-46,39	3	163,89	3	-37,44	3	204,73	3	-13,88	3	243,12	3	-13,88
	4	142,77	4	-46,39	4	163,89	4	-37,44	4	204,73	4	-13,88	4	243,12	4	-13,88
	5	142,77	5	-46,39	5	163,89	5	-37,44	5	204,73	5	-13,88	5	243,12	5	-13,88
	6	142,77	6	-46,39	6	163,89	6	-37,44	6	204,73	6	-13,88	6	243,12	6	-13,88
	7	158,02	7	-46,39	7	179,99	7	-41,92	7	229,99	7	-17,91	7	257,95	7	-13,88
	8	158,02	8	-46,39	8	179,99	8	-41,92	8	229,99	8	-17,91	8	257,95	8	-13,88
	9	158,02	9	-46,39	9	179,99	9	-41,92	9	229,99	9	-17,91	9	257,95	9	-13,88
	10	158,02	10	-46,39	10	179,99	10	-41,92	10	229,99	10	-17,91	10	257,95	10	-13,88
Federstift/- weihen 2000x steifer	1	111,98	1	-30,13	1	133,65	1	-34,49	1	163,96	1	-6,24	1	190,94	1	1,03
	2	111,98	2	-32,97	2	133,65	2	-37,46	2	163,96	2	-9,83	2	190,94	2	8,34
	3	118,65	3	-32,97	3	133,65	3	-37,46	3	163,96	3	-9,83	3	190,94	3	8,34
	4	118,65	4	-32,97	4	133,65	4	-37,46	4	163,96	4	-9,83	4	190,94	4	8,34
	5	124,87	5	-33,93	5	140,01	5	-33,93	5	177,79	5	7,10	5	203,02	5	-7,30
	6	124,87	6	-33,93	6	140,01	6	-33,93	6	177,79	6	7,10	6	203,02	6	-7,30
	7	133,90	7	-39,04	7	151,08	7	-31,01	7	193,97	7	-16,27	7	219,45	7	32,20
	8	133,90	8	-39,04	8	151,08	8	-31,01	8	193,97	8	-16,27	8	219,45	8	32,20
	9	143,97	9	-43,76	9	165,35	9	-35,41	9	206,17	9	-11,42	9	244,88	9	-13,23
	10	143,97	10	-43,76	10	165,35	10	-35,41	10	206,17	10	-11,42	10	244,88	10	-13,23
Federstift/- weihen 5000x steifer	1	116,60	1	-43,76	1	146,18	1	-37,00	1	211,91	1	-13,35	1	244,88	1	-13,23
	2	116,60	2	-43,76	2	146,18	2	-37,00	2	211,91	2	-13,35	2	244,88	2	-13,23
	3	119,32	3	-43,76	3	146,18	3	-37,00	3	211,91	3	-13,35	3	244,88	3	-13,23
	4	119,32	4	-43,76	4	146,18	4	-37,00	4	211,91	4	-13,35	4	244,88	4	-13,23
	5	121,48	5	-43,76	5	146,18	5	-37,00	5	211,91	5	-13,35	5	244,88	5	-13,23
	6	121,48	6	-43,76	6	146,18	6	-37,00	6	211,91	6	-13,35	6	244,88	6	-13,23
	7	134,01	7	-43,76	7	159,79	7	-37,00	7	211,91	7	-13,35	7	244,88	7	-13,23
	8	134,01	8	-43,76	8	159,79	8	-37,00	8	211,91	8	-13,35	8	244,88	8	-13,23
	9	142,27	9	-43,76	9	168,78	9	-37,00	9	211,91	9	-13,35	9	244,88	9	-13,23
	10	142,27	10	-43,76	10	168,78	10	-37,00	10	211,91	10	-13,35	10	244,88	10	-13,23