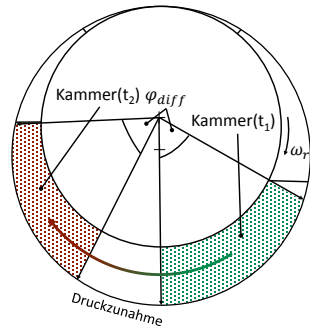


Computergestützte Verdichteroptimierung mit Hilfe von MATLAB – Entwicklung eines Expertentools

Kaczmarek, D.; Lohrengel, A.



In Zusammenarbeit mit dem Industriepartner DEWA Engineering und Anlagenbau GmbH entwickelt das Institut für Maschinenwesen ein Expertentool in MATLAB für eine neuartige Verdichtertechnik. Das Expertentool soll im ersten Schritt ermöglichen, verschiedene Baugrößen des Verdichters simulativ in Bezug auf unterschiedliche Betriebsgrößen abzubilden. Im zweiten Schritt soll die Software so konzipiert werden, dass ideale geometrische Zusammenhänge gefunden und darauf aufbauend eine Baureihenentwicklung entstehen soll. Neben der programmtechnischen Entwicklung entsteht parallel ein Prüfstand, welcher zur Verifizierung der Software dienen wird.

In cooperation with the industrial partner DEWA Engineering und Anlagenbau GmbH, the Institute of Mechanical Engineering is developing an expert tool using MATLAB for a new compressor technology. In the first step, this expert tool should make it possible to simulate various sizes of the compressor. In the second step, the software is to be designed to find ideal geometrical relationships and based on this, a series development is to be developed. In addition to the software a test bench will be created, which will be used to verify the software.

Vorgehensweise

Agile Softwareentwicklung

Das grundlegende Vorgehen bei der Softwareentwicklung für den neuartigen Verdichter, ist an die „agile Softwareentwicklung“ angelehnt. Die „agile Softwareentwicklung“ verfolgt das Prinzip der vollständigen Flexibilität sowie der praxisnahen Entwicklung von Programmmodulen. Erreicht werden die Flexibilität und die praxisnahe Entwicklung durch ständigen Kontakt mit dem Industriepartner und frühzeitigen „Beta“ Bestandteilen der Software zu Testzwecken. Durch den ständigen Kontakt und den Testzyklen wird ein wertvoller Informationsfluss von bedarfsgerechten Programmbestandteilen kommuniziert, um so eine anforderungserfüllende Software programmtechnisch umzusetzen. In dem Zusammenhang gestaltet der Partner die Software maßgeblich mit und kann so

auch bewährte Vorgehensweisen, praxistaugliche I/O-Dateien, technische Abbildungen/Graphen, Hinweisdialoge oder auch weitergehende Zukunftsideen in die Softwareentwicklung integrieren. Das Ergebnis dieser Vorgehensweise ist das problemlösende Tool.

Bedarfsanalyse

Die computergestützte Produktentwicklung beginnt mit einer Bedarfsanalyse, hier wird hinsichtlich des Produktes und der Software eine Anforderungsliste erstellt. In diesem Dokument stehen so die wichtigsten zu erfüllenden Anforderungen sowie weitere Hinweise, welche zur programmtechnischen Modellbildung wichtig sind. Eine besondere Wertstellung besitzt die klare definierte Funktion des neuen Produktes, um so die Funktionalität in der Modellbildung zu berücksichtigen. Dabei sind zusätzlich Informationen bezüglich der Betriebsparameter, geometrischen Abmaße, Systemcharakteristiken (wie zum Beispiel die Effizienz, der kinematische Zusammenhänge, der Leistungsbedarf sowie der Verschleiß) oder aber auch bestimmte Voraussetzungen für die Software (wie zum Beispiel der erforderlichen Schnittstellen, unterstützte Betriebssysteme, Erzeugung von temporären Dateien) und weiteres zu notieren. Das Ergebnis dieses Arbeitspunktes sollte eine Auflistung der wichtigsten Details sowie die zu erreichenden Leistungskriterien des Produktes hervorbringen.

Strukturierung der Inhalte

Nach der Bedarfsanalyse erfolgt die Strukturierung der Software. Bewährt hat sich das Vorgehen über das systematische Analysieren des Gesamtkomplexes. Dabei sind die Teilbereiche des Systems zu sortieren und die Grundlagen zu diesen Themen aufzuarbeiten. Mit Hilfe des Grundverständnisses kann anschließend ein guter Überblick der Subsysteme erstellt und somit die Strukturierung des groben Programmablaufes durchgeführt werden. Eine gute Hilfestellung bietet zum Beispiel ein Flussdiagramm, in dem die Zusammenhänge festgehalten, kommentiert und geordnet werden. Ebenso sollten Gedanken zu möglichen Prüfstandsaufbauten und der Sensorik erfolgen. Dadurch sollen in den jeweiligen Subsystemen die Rechenansätze so angepasst werden, dass die Sensorgrößen ausschlaggebend für das Ergebnis des Subsystems sind. Hierdurch lässt sich im weiteren Verlauf eine schnelle Realisierung der Systemverifizierung sowie -validierung ermöglichen. Darüber hinaus sollte in diesem Arbeitspaket geklärt werden, wie die interne Datenablage realisiert, in welcher Form Ergebnisse ausgegeben und welche Dokumentationsvorgehensweise, bezüglich der Software, verfolgt werden. Das Ergebnis dieses Arbeitspakets ist die Generierung einer klaren Struktur der Software.

Erstellung des Rechenkerns

Sobald die Struktur der Software feststeht, kann mit dem Programmieren des Inhaltes angefangen werden. Als Voraussetzung für die Programmierung gilt es die Grundlagen, zu den verschiedenen technischen Zusammenhängen, zu beherrschen. Zeitlich kinematische Abläufe müssen bekannt sein, Randbedingungen aus der Anforderungsliste, Rückkopplungen und Wechselwirkungen des Gesamtkonstrukts müssen berücksichtigt werden. Bewährt hat sich das Beginnen mit der Programmierung der kinematischen Zusammenhänge. Ausgehend von diesem Standpunkt, werden die Grundlagen für alle weiteren technischen Aspekte des Systems betrachtet. Sofern gewisse Randbedingungen unklar sind, müssen sinnvolle Annahmen oder Idealisierungen getroffen werden. Die Grundaufgabe dieses Arbeitspaketes besteht in der programmtechnischen Aufbereitung des zu simulierenden Gesamtsystems, mit tiefgreifendem Verständnis. Neben der Systemanalyse und Rechenkernprogrammierung, ist es zudem notwendig, Bestandteile der programmtechnischen Organisation auszuarbeiten. Hierzu zählen das Programmieren von Kommunikationsmodulen, für die interne Verwendung der Ergebnisse, eine Fehler- und Hinweishandhabung, die Aufbereitung der Schnittstellendaten, für die interne Verrechnung im System, die Erstellung von nützlichen Abbildungen zum eigenen Verständnis (zum Beispiel programmtechnisch gebildete Punktwolken inklusive Plot), die Graphen von Betriebsgrößen, die Schaffung einer Bedienoberfläche, zudem sind Überlegungen für ergänzende Programmbestandteile durchzuführen und Maßnahmen für die Anbindungen zu treffen. Die Visualisierung der Daten sollte bereits in einem sehr frühen Stadium der Softwareentwicklung erfolgen, um bereits bei der Beta-Versionsausgabe mögliche fehlerhafte Bestandteile in der Software ausfindig zu machen. Mit Abschluss dieses Arbeitspaketes steht der fehlerfreie Rechenkern.

Testphase

Nun folgt die kritische Betrachtung des Gesamtsystems, was auch als die Testphase der Software bezeichnet werden kann. Viele Bestandteile des Programms wurden bereits auf ihre Funktion überprüft, doch im Verbund des gesamten Systems können diverse Fehler auftreten. Beispiele hierfür wären das Überschreiben von Variablen, ein unzulässiger Input für eine Funktion, das Auftreten von ungewollten imaginären Zahlen oder Teilungen durch Null sowie Datentypprobleme. Diese unzulässigen Probleme müssen in den nächsten Schritten abgearbeitet werden. Die Software sollte ab diesem Zeitpunkt fehlerfrei laufen, oder bei unzulässigen Eingaben, Hinweise beziehungsweise Fehler ausgeben. Ein Abstürzen des Programms ist nicht mehr zulässig und muss programmtechnisch abgefangen werden. Weiter gilt es, die erstellte Software mit einem realen Modell zu verifizieren sowie zu validieren. Dazu sollte im besten Fall ein Prüfstand bereits parallel im Arbeitspaket „Erstellung des Rechenkerns“ entwickelt und gefertigt werden. Anhand des Prüfstandes sollten dann fertigungstechnische, praxistaug-

liche sowie konstruktive Maßnahmen überprüft und gegebenenfalls Anpassungen an bestimmten Funktionselementen durchgeführt werden. Sobald der Prüfstand gefertigt und montiert ist sowie die Software fehlerfrei funktioniert, sind Abgleichtests durchzuführen. Das Ergebnis dieser Phase sollte die fertige Software mit einem praxisnahen Parameterabgleich sein. Im Anschluss können an diesen Rechenkern noch weitere programmtechnische Inhalte angefügt werden.

Aktuelles Forschungsprojekt

Zielsetzung

Hintergrund des Forschungsprojektes ist die Entwicklung einer Baureihenentwicklung für einen neuen Verdichtertyp. In der *Abbildung 1* ist der Verdichter exemplarisch abgebildet.

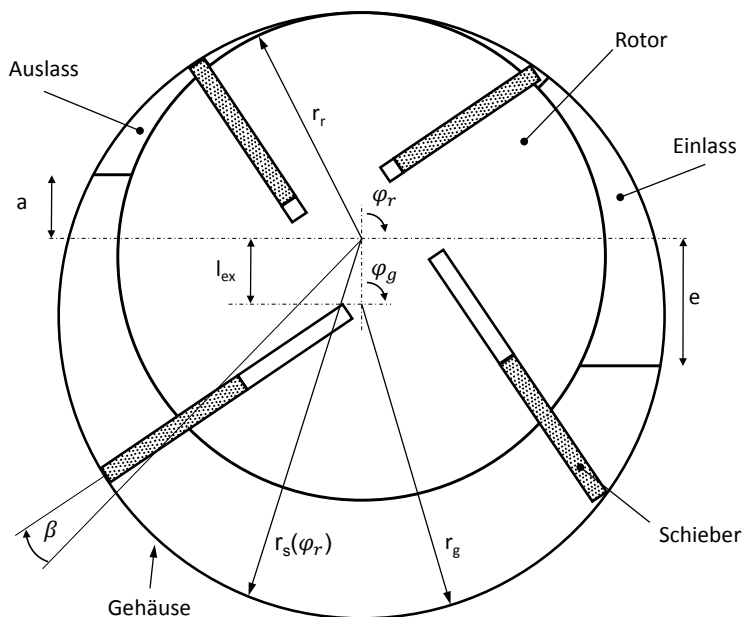


Abbildung 1: Aufbau des Drehschieberverdichters

Das am IMW verfolgte Forschungsprojekt befindet sich im Aufbau des Rechenkerns der Simulationssoftware - „Erstellung des Rechenkerns“. Es ist ein Simulationsmodell entstanden, welches über die grundlegenden konstruktiven Parameter des Drehschieberverdichters einstellbar ist. Dieses Simulationsmodell bietet die Freiheit, verschiedenste Variationsrechnungen in einem großen

Parameterfeld durchzuführen. Dadurch ist es möglich, einen bestimmten Betriebspunkt des entwickelten Modells simulativ abzubilden. Die Ergebnisse der jeweiligen Simulationen ermöglichen eine genaue Analyse des Verdichters, welche eine Auslegung in definierten Betriebszuständen gestattet.

Die Betriebsgrößen des Verdichters werden innerhalb des Programms aufbereitet und passende Abbildungen in Abhängigkeit des Winkels erstellt. Aus diesen Abbildungen können wichtige Kenngrößen oder qualitative Verläufe direkt abgelesen und abgeschätzt werden. Somit stellt die erste Version des Programms eine gute Möglichkeit für Überschlagsrechnungen des Drehschieberverdichters dar. Durch die simulativ ermittelten Ergebnisse kann zum Beispiel die optimale Einlassgeometrie in Abhängigkeit der Schieberanzahl ermittelt werden. Die optimierte Geometrie ermöglicht ein besseres Verdichtungsverhalten zu erzielen, sodass der maximal mögliche Enddruck und Volumenstrom erreicht werden kann. In der *Abbildung 2* ist das Verdichtungsverhältnis in Abhängigkeit der Einlassgeometrie abgebildet.

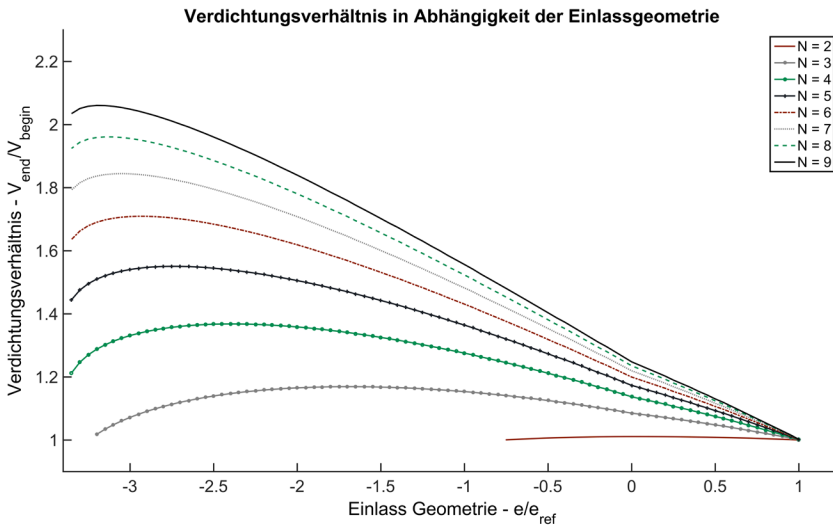


Abbildung 2: Verdichtungsverhältnis in Abhängigkeit der Einlassgeometrie

Die *Abbildung 2* wurde mit einem geometrisch gleichbleibenden Verdichter entwickelt, variiert wurden ausschließlich die Schieberanzahl N sowie die Einlassgeometrie e . Aus der Abbildung ist ersichtlich, dass mit Variation der Schieberanzahl, auch die Extremwerte des maximalen Verdichtungsverhältnisses variieren. Demnach existiert für jede Schieberkonfiguration eine andere optimale geometrische Beziehung von e .

Darüber hinaus wurden weitere Untersuchungen an dem Drehschieberverdichter durchgeführt. Als eine beispielhafte Ausgabe sei hier das Schiebersystem genannt. In der *Abbildung 3* ist die Lastsituation des Schiebers im Rotor gezeigt.

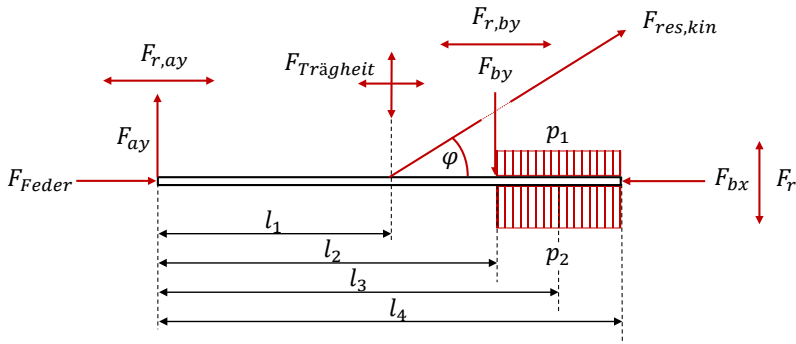


Abbildung 3: Belastungssituation des Schiebers im Gesamtsystem

Die eingetragenen Belastungen sowie Längenverhältnisse sind aus den kinematischen Zusammenhängen, des Zusammenspiels der Eingabeparameter sowie des thermodynamischen Subsystems entstanden. Dieses Teilsystem des Schiebers ermöglicht es, die Belastungssituation des Schiebers numerisch in allen Zeitschritten abzubilden und somit Einflussanalysen durchzuführen. Als beispielhafte Analyse ist in *Abbildung 4* die Variation der Schieberanzahl und deren Einfluss auf die Schiebergleitbuchsenkräfte F_{ay} und F_{by} abgebildet.

Aus der *Abbildung 4* geht hervor, wie eine Erhöhung der Schieberanzahl zum einen die Dynamik der Schieberbelastung verbessert und zum anderen die sich einstellenden Grundbelastungsverläufe beeinflusst. Eine größere Schieberanzahl führt demnach zu einer gleichmäßigeren Belastung. Im Konflikt steht diese Belastungsoptimierung jedoch mit der Effizienz des Drehschieberverdichters, denn die Erhöhung der Schieberanzahl führt zu einer Zunahme der Reibkontakte. Zu erwähnen ist, dass für bestimmte Massendurchsätze ideale Konfigurationen existieren.

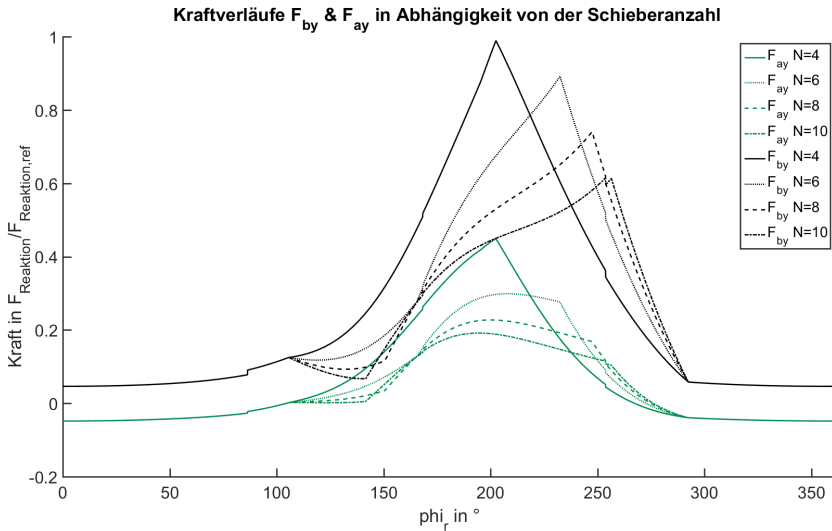


Abbildung 4: Schieberbelastungsgrößen in Abhängigkeit des Rotorwinkels

Dadurch, dass der Informationsgehalt der Ergebnisse sehr umfassend ist, kann nicht aus einfachen Variationen der Eingabeparameter auf die Idealgeometrie geschlossen werden. Somit ist es notwendig, diese Variationsrechnungen im Gesamtsystem zu betrachten und das Wechselwirken der Parametervariationen im System zu berücksichtigen. Im weiteren Verlauf des Forschungsprojektes wird eine Optimierungsaufgabe an das Simulationsmodell angefügt. Hierdurch wird es ermöglicht, durch Vorgabe von gezielten Präferenzen im Bereich der Ergebnisse, eine Extremwertaufgabe zu lösen.

Als beispielhafte Extremwertaufgabe kann die Reibleistung als Optimierungsparameter gewählt werden. Hierzu wird ein Datensatz an den Rechenkern übermittelt, dieser rechnet die zu erwartenden Belastungen, Temperaturen, Drücke, Reibleistungen, Pressungen, Kräfte usw. aus. Diese Ergebnisse werden mit einer Kostenfunktion bewertet, dabei werden den relevanten Parameter Gewichtungsfaktoren zugeteilt und fließen folglich in die Summe der Kostenfunktion ein. Das Ergebnis der Kostenfunktion soll letztlich optimiert werden, um so die Einflüsse weiterer Betriebsgrößen zu berücksichtigen. Dieser Ablauf wird so lange wiederholt, bis die Kostenfunktion einen Extremwert erreicht. Eine Visualisierung der Abläufe ist in *Abbildung 5* gezeigt.

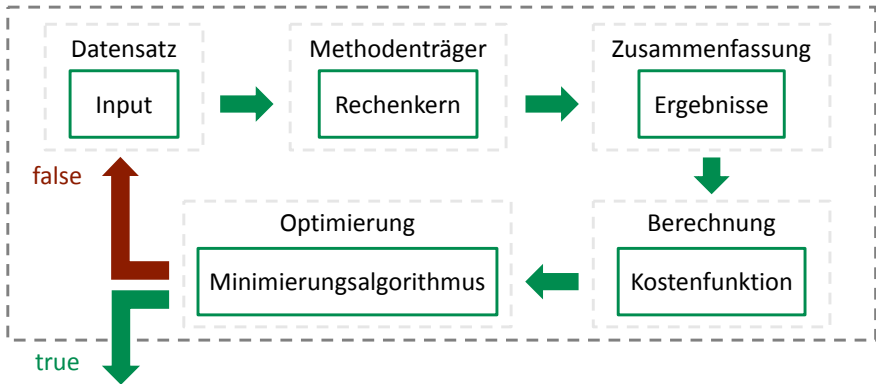


Abbildung 5: Optimierungsablauf

Mit Hilfe des Optimierungsalgorithmus soll im nächsten Schritt eine Baureihenentwicklung entworfen werden. Dazu sollen die idealen Inputparameter für die jeweiligen Optimierungsfälle, unter Vorgabe von Druck und Massestrom, als Funktionen erstellt werden. Diese Funktionen sollen genutzt werden, um mit Hilfe einer Optimierungsgröße wie beispielsweise der Festigkeit und unter Vorgabe der Betriebsgrößen wie Druck sowie des Massestromes einen idealen Verdichter zu ermitteln.

Zusammenfassung

Im Rahmen des Forschungsprojektes sind bisher ein Simulationsprogramm und Empfehlungen für Geometriekombinationen entstanden. Das Simulationsprogramm erlaubt Überschlagsrechnungen zu dem neuartigen Verdichter. Im weiteren Verlauf des Projektes ist eine Baureihenentwicklung des Verdichters mit optimalen Geometrien geplant. Dadurch sollen zum einen Empfehlungen für die Konstruktion des Verdichters entstehen und zum anderen weiterhin Überschlagsrechnungen für spezifische Konstruktionsparameter innerhalb des Simulationsprogramms ermöglicht werden. Nach Abschluss des Forschungsprojektes steht dem Industriepartner eine als stand-alone lauffähige Software zur Verfügung.

Kooperationspartner

