



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Projektarbeit

Studiendepartment Fahrzeugtechnik und Flugzeugbau

**Methodisches Entwerfen von Verkehrsflugzeugen
mit PrADO**

Sebastian-Verian Herda

16. Juni 2008

Kurzreferat

Diese Projektarbeit beschäftigt sich mit den Möglichkeiten, Schwächen und Leistungsgrenzen von *PrADO* (Preliminary Aircraft Design and Optimization).

PrADO ist ein interdisziplinär agierendes Vorentwurfsprogramm. Es wurde vom Institut für Flugzeugbau und Leichtbau der TU Braunschweig aufgelegt und wird von diesem laufend weiterentwickelt bzw. angepasst.

Das in der Aufgabenstellung vorgegebene Passagierflugzeug vom Typ Boeing B777F wurde während der Projektarbeit gegen eine ATR72-200F ausgetauscht (siehe Motivation).

Aus diesem Grund bauen alle von mir während des Arbeitens mit *PrADO* gewonnenen und hier zusammengetragenen Erkenntnisse auf der ebenfalls die ATR72-200F behandelnden Arbeit „Methodisches Entwerfen von Verkehrsflugzeugen mit *PrADO*“ von Thorsten Kiesel auf und ergänzen diese.

Die Projektarbeit soll dem Leser in Form eines Praxisleitfadens einen Überblick über die Funktionalitäten und die Anwendung von *PrADO* geben.





DEPARTMENT FAHRZEUGTECHNIK UND FLUGZEUGBAU

Methodisches Entwerfen von Verkehrsflugzeugen mit PrADO

Aufgabenstellung zum *Projekt 2* gemäß Prüfungsordnung

Hintergrund

PrADO (Preliminary Aircraft Design and Optimisation Program) ist ein Programm des Instituts für Flugzeugbau und Leichtbau der Technischen Universität Braunschweig für den iterativen, multidisziplinären Entwurfsprozess von Flugzeugen. Es besteht aus einer großen Anzahl von Unterprogrammen, welche die Beiträge der wichtigsten am Entwurf beteiligten Fachgebiete widerspiegeln. Diese Teilprogramme werden je nach Entwurfproblem oder zu untersuchender Konfiguration zu einem Gesamtsystem zusammengestellt und ggf. durch fehlende Analyseprogramme ergänzt. Das Programm wird an der HAW eingesetzt im Forschungsprojekt *Grüner Frachter* (<http://GF.ProfScholz.de>).

Aufgabe

Mit dem Programm PrADO soll ein Modell eines Passagierflugzeugs vom Typ Boeing B777F erstellt werden. Es gilt, die Grundlagen des Programms zu erklären sowie die Arbeitsweise des Programms zu beschreiben und zu bewerten.

Dabei soll/sollen:

- die Modellbildung in PrADO beschrieben werden,
- die Ermittlung und Eingabe der Geometriedaten bestehender Flugzeuge systematisiert und vereinfacht werden,
- verschiedene Entwurfsvarianten untersucht werden,
- die Möglichkeiten von PrADO in Bezug auf Parametervariation und Optimierung getestet und beschrieben werden.

Die Ergebnisse sollen in einem Bericht dokumentiert werden. Bei der Erstellung des Berichtes sind die entsprechenden DIN-Normen zu beachten.

Inhalt

Kurzreferat.....	3
Erklärung	5
Liste der Abkürzungen.....	8
Verzeichnis der Begriffe und Definitionen.....	9
1 Einleitung	10
1.1 Motivation	10
1.2 Ziel der Arbeit.....	11
1.3 Aufbau der Arbeit.....	12
2 Grundlagen des Programms PrADO	13
2.1 Einsatzbereich	13
2.2 Grenzen	14
2.3 Der Kern von PrADO	15
2.3.1 Mathematische und physikalische Modellbildung	16
2.3.2 Integrierte Programme	19
2.4 Interessante Möglichkeiten	20
2.5 Informationen zu der benutzten PrADO-Version.....	22
3 Allgemeines.....	23
3.1 Ordern und Dateien von PrADO	23
3.2 Der Projektordner.....	24
3.2.1 Ursprung der Dateien im Projektordner	24
3.2.2 Die Funktion der einzelnen Dateien	26
3.3 GUI Oberflächen	28
3.3.1 Vorhandene Benutzeroberflächen	29
3.3.2 Starten von Java-Oberflächen	31
3.3.3 Anpassen der GUI-Oberfläche.....	32
3.3.4 Programme ohne GUI ausführen.....	33
3.4 Ergänzungen und Hinweise	34
3.4.1 Allgemeines	34
3.4.2 Fehler während der Installation.....	35
3.4.3 Anpassen des modularen Aufbaus	35
3.4.4 Der Sinn der Technologiefaktoren	35
3.4.5 „BI“ und „AI“	35
4 Die eigentliche Aufgabe von PrADO	36
4.1 E/P/O allgemein	36
4.2 Die Iteration	36
4.3 Die Einzelanalyse	36
4.4 Die Parametervariation	38
4.5 Die Optimierung	39
5 Die Ergebnisse von PrADO	42
5.1 Ergebnismöglichkeiten.....	42
5.1.1 Ergebnisse aus der E/P/O	42
5.1.2 Daten zu den einzelnen Disziplinen	43
5.1.3 Informationen zur Lärmausbreitung	46
5.2 Kontrolle der gewonnenen Informationen	47

5.2.1	Überprüfung der eingegebenen Daten.....	47
5.2.2	Geometriekontrolle von Nachentwürfen.....	48
5.2.3	Fehler während der E/P/O	54
5.3	Bewertung der Genauigkeit	56
6	Das Arbeiten mit PrADO.....	57
6.1	Der Einstieg	57
6.2	Die Handhabung der Vorgabedatei.....	60
6.2.1	Allgemeines:	60
6.2.2	Sam und LLIE	61
6.2.3	Auswahl der gewünschten Vorgabedatei.....	61
6.3	Fehlerquellen	62
6.3.1	In der Vorgabedatei	62
6.3.2	In Tecplot	63
6.3.3	Fehlermeldungen während des Programmlaufs	64
6.4	Hilfreiche Tools für das Arbeiten mit PrADO.....	65
6.4.1	Ermittlung der relativen Geometriedaten	65
6.4.2	Vergleichen von Texten	69
6.4.3	Ermitteln von Flächen	70
6.4.4	Verbinden von Dateien	71
6.4.5	Eine Alternative zum Editor.....	72
7	Anwendung an der ATR 72-200F.....	73
7.1	Allgemeines zur ATR	73
7.2	Was war der Stand?	74
7.3	Vorgenommene Veränderungen.....	75
7.3.1	Das SLW.....	75
7.3.2	Die Flugzeugsysteme	76
7.3.3	Die Rumpfdati und die Cockpitfenster.....	77
7.3.4	Die Profildatei.....	79
7.3.5	Die Triebwerksdatei	80
7.4	Die Ergebnisse des Nachentwurfes	80
7.4.1	Ergebnisse der Machbarkeitsstudie	80
7.4.2	Abweichungen zur realen ATR	81
8	Entwicklungspotential von PrADO.....	82
8.1	Noch zu beseitigende Missstände	82
8.1.1	In den FORTRAN-Programmtexten	82
8.1.2	In der Vorgabedatei	83
8.1.3	In der GUI	83
8.1.4	Im Programmlauf	84
8.2	Verbesserungsideen	84
8.2.1	Allgemein	84
8.2.2	In der GUI, dem Programmlauf und der Vorgabedatei.....	86
8.3	Mögliche zukünftige Ergänzung	89
9	Zusammenfassung	90
10	Literaturverzeichnis.....	91
Anhang A: Kurzprotokoll der ATR72-200F		93

Liste der Abkürzungen

2D	zweidimensional
3D	dreidimensional
ATR	Avions de Transport Régional
CFD	Computational Fluid Dynamics
CPU	Central Processing Unit
DB	Datenbank
DGLR	Deutsche Gesellschaft für Luft- und Raumfahrt
DMS	Daten Management System
DOC	Direct Operational Cost
E/ P/ O	Einzelanalyse, Parametervariation und Optimierung
EAS	Equivalent Air Speed
FAR	Federal Aviation Regulations
FE	Fenite Elemente
FEM	Finite Element Methode
FTM	Flächenträgheitsmoment
GAMMA	Bahnwinkel
GUI	Graphic User Interface
IFL	Institut für Flugzeugbau und Leichtbau
LLIE	Liftingline
LTH	Luftfahrt-Technisches Handbuch
MD	Modul
MTOW	Maximum Take-Off Weight
NACA	National Advisory Committee for Aeronautics
PrADO	Preliminary Aircraft Design and Optimisation Program
PTL	Propeller-Turbinen-Luftstrahltriebwerk
PW	Pratt & Whitney
SAM	Structural Sizing Modul
SEP	Specific-Excess-Power
SFC	Specific Fuel Consumption
SIGMA	(Schubwinkel)
TL	Turbinen-Luftstrahltriebwerk
ZTL	Zweistrom-Turbinen-Luftstrahltriebwerk

Verzeichnis der Begriffe und Definitionen

Anforderung

„Eine Anforderung ist die zumeist zahlenmäßig bestimmte Zuordnung zu einem Flugparameter die der Entwurf erfüllen muss.“

(Walde 2000)

Entwurfsparameter

„Der Entwurfsparameter ist ein Zahlenwert, der durch den Entwurf ermittelt wurde. Er kann danach aber wieder zu einer Eingangsgröße für weitere Berechnungen werden.“

(Walde 2000)

Flugenveloppe

„Ist die Hüllkurve der möglichen Leistung eines Flugkörpers in einem Höhen-Geschwindigkeitsdiagramm oder Lastvielfachem-Geschwindigkeitsdiagramm. Bisweilen werden diese Diagramme auch in einem Bild kombiniert. Flugenveloppen kennzeichnen den möglichen Betriebsbereich eines Flugkörpers.“

(www.Wikipedia.org 2008)

Iteration

„In der numerischen Mathematik bezeichnet er eine Methode, sich der Lösung eines Rechenproblems schrittweise, aber zielgerichtet anzunähern. Sie besteht in der wiederholten Anwendung desselben Rechenverfahrens.“

(www.Wikipedia.org 2007)

Konvergenz

„Mathematik: Eigenschaft von Folgen, einem Grenzwert zuzustreben (zu konvergieren); spezielle Folgen sind die Reihen. Existiert kein solcher Grenzwert, so ist die Folge divergent.“

(www.lexikon.Meyers.de 2007)

Penaud-Diagramm

Das Penaud-Diagramm stellt die Fahrtpolare dar. Es enthält die Darstellung des Verhältnisses zwischen dem Widerstandsbeiwert und dem Auftriebsbeiwert und dem Verhältnis der Schubkraft zur Schwerkraft über der Geschwindigkeit.

Softwarearchitektur

„Eine Softwarearchitektur beschreibt die grundlegenden Komponenten und deren Zusammenspiel innerhalb eines Softwaresystems.“

(www.Wikipedia.org 2008)

1 Einleitung

1.1 Motivation

Aktuelle Prognosen gehen davon aus, dass bis 2025 doppelt so viele Frachtflugzeuge (also ca. 4.000 Stück) notwendig sein werden, um das dann vorherrschende Transportvolumen zu bewältigen.

Da viele Flughäfen die maximale Anzahl an möglichen Starts und Landungen allerdings bereits heute erreicht haben, kann dieser Anforderung nicht durch die oben beschriebene Erhöhung der Anzahl von Frachtflugzeugen entsprochen werden. Stattdessen müsste die Ladekapazität der vorhandenen Flugzeuge erhöht bzw. diese durch leistungsstärkere ausgetauscht werden.

Viele ältere Flugzeuge, die zum Frachter umgebaut wurden oder werden könnten, können aufgrund ihrer veralteten Technik und ihrer konventionellen Bauweise keine nennenswerten Kapazitätssteigerungen ermöglichen.

Aber auch bei den Neuentwicklungen hat man im Bereich der konventionellen Flugzeuge mit dem neuen A380 bezüglich der Größe die Grenze des maximal Möglichen erreicht.

Aus diesem Grund wurde das Projekt *Grüner Frachter*¹ ins Leben gerufen. Die Ziele des Projektes sind:

- Senkung des Kraftstoffverbrauches
- Verwendung neuer Kraftstoffe
- Minimierung des Lärms, um
 - Nachtflüge zu ermöglichen und damit die Flughafenkapazität zu erhöhen
 - die Belastung des Flughafenumfeldes zu reduzieren
- Reduktion der Emissionen (CO₂, NO_x, usw.)
- Senkung der Operationskosten
- keine oder weniger „Environmental Control Systems“
- ggf. Entwicklung eines autonom, also ohne Piloten, fliegenden Flugzeuges

Um diese Ziele zu erreichen sind nicht nur neue Technologien notwendig, sondern auch ein innovatives und unkonventionelles Flugzeugdesign.

Bei der Bewertung neuer Flugzeugkonzepte kommt das im Rahmen dieser Arbeit benutzte interdisziplinäre Optimierungsprogramm PrADO zum Einsatz.

Um mit Hilfe des Programms zu ermitteln, ob sich die neuen Denkansätze gegenüber bestehenden Konzepten positiv oder nachteilig auswirken, werden Vergleiche mit konventionellen Flugzeugkonfigurationen durchgeführt.

¹ Informationen unter: <http://www.fzt.haw-hamburg.de/pers/Scholz/GF.html>

Bei der Wahl eines solchen Referenzflugzeuges fiel die Entscheidung im Rahmen des Grünen Frachters auf das ATR72-200F, das auch T. Kiesel bei seiner Ausarbeitung heranzog. Aus diesem Grund wurde auch in dieser Arbeit der programmierte Algorithmus an der Frachtversion der ATR72-200 validiert.

1.2 Ziel der Arbeit

Ziel dieser Projektarbeit ist es, dem Leser einen schnellen Einstieg in PrADO zu ermöglichen. Da hierbei aber gleichzeitig ein tiefes Verständnis für das Programm und die einzelnen Module, Abläufe und Berechnungen vermittelt werden soll, wird die Kenntnis über Grundzüge des Programms vorausgesetzt. Der Leser kann das entsprechende Grundwissen bei Bedarf mit Hilfe der bereits erwähnten Arbeit von Thorsten Kiesel erwerben.

Die vorliegende Projektarbeit soll den Leser darüber informieren, welche Funktionen zur Verfügung stehen. Außerdem soll er darüber aufgeklärt werden, wo die gewünschten Funktionen angesteuert und wie sie bedient werden können.

Damit der Leser in die Lage versetzt wird, ein korrektes und aussagefähiges Ergebnis zu erzielen, wird auf besonders zu beachtende Details und Stolperfallen konkret hingewiesen. Ergänzend werden wertvolle Tipps zur Fehlervermeidung gegeben.

Die Projektarbeit wurde mit zahlreichen Beispielen und Bildern angereichert, um dem Anwender eine möglichst praxisnahe Hilfestellung zu bieten.

Da der Verfasser bei der Einarbeitung in und Anwendung von PrADO auf einige Unstimmigkeiten im Programm gestoßen ist, wurden in die Projektarbeit auch Verbesserungsvorschläge aufgenommen, die einen Beitrag zur Weiterentwicklung des Programms und Steigerung der Bedienerfreundlichkeit leisten sollen.

1.3 Aufbau der Arbeit

Im Hauptteil dieses Berichtes werden die mit PrADO gemachten Erfahrungen anhand von Beispielen und Beschreibungen geschildert:

Abschnitt 2 erläutert den Zweck und die Struktur von PrADO und informiert über in PrADO eingebundene Programme

Abschnitt 3 zählt allgemeine Inhalte über den Aufbau auf

Abschnitt 4 erklärt die Möglichkeiten der Einzelanalyse, der Parametervariation und der Konfigurationsoptimierung

Abschnitt 5 widmet sich dem praktischen und effizienten Arbeiten mit PrADO

Abschnitt 6 zeigt die erzielbaren Ergebnisse und deren Visualisierung auf

Abschnitt 7 enthält Informationen zur Anwendung des Programmes an einer ATR72-200F

Abschnitt 8 beschreibt mögliche zukünftige Verbesserungen des Programmes

Anhang A Ergebnisse der projizierten ATR72-200F aus der Machbarkeitsstudie in Form eines Kurzprotokolls

2 Grundlagen des Programms PrADO

2.1 Einsatzbereich

Nach den Anfängen der Flugzeugentwurfsoptimierung mit empirischen Ansätzen, entstanden in den letzten beiden Jahrzehnten neben der Optimierung von voneinander getrennten Disziplinen mit CFD, FEA/FEM-Systemen mehr und mehr disziplinübergreifende Optimierungsprogramme. Diese disziplinübergreifende Programme benutzen mathematisch-physikalischen Modellen, um die realen Zusammenhänge möglichst anforderungsgerecht nachzubilden.

Hierbei bauen viele allgemeine Tools auf den Programmen: *Excel*, *Matlab*, und *Mathematica* auf. Diese Softwareumgebungen sind zwar einfach zu erlernen und zu benutzen, aber ihre Performance reicht für große ingenieurstechnische Probleme nicht aus. Desweiteren lassen sich Programme wie CFD und FE nicht oder nur sehr schwer integrieren.

Das multidisziplinäre Optimierungsprogramm, um das es in dieser Arbeit geht, heißt *PrADO* und wurde am Institut für Flugzeugbau und Leichtbau entwickelt. Dieses Programm bildet den gesamten vernetzt iterativen Entwurfsprozess der beteiligten Fachdisziplinen nach.

Es werden folgende Disziplinen mit ihren zunächst diskreten und später kontinuierlichen Entwurfsparametern mit einbezogen:

- Flugzeugbau
- Leichtbau
- Aerodynamik
- Flugmechanik
- Antrieb

Desweiteren werden die Umweltbedingungen in Form eines Atmosphärenmodells berücksichtigt.

Da die Beste und somit wirtschaftlichste Lösung eines Flugzeugkonzeptes ein Kompromiss aus all diesen voneinander abhängigen Bereichen darstellt, lassen sich diese multidisziplinären Berechnungen nur iterativ lösen.

Diese sehr aufwendigen Berechnungen, mit allen benötigten Iterationschritten, lassen sich mit PrADO automatisch durchführen. Somit lässt sich der Zeitaufwand für derartige Rechnungen drastisch minimieren.

Aufgrund dieser globalen Sichtweise von *PrADO* ist es möglich:

- aus einem Satz von Vorgaben für eine definierte Transportaufgabe die Machbarkeit zu überprüfen. (siehe Kapitel 4.3)
- die Auswirkungen von einzelnen Parametern auf die gesamte Flugzeugkonfiguration berechnen zu lassen. (siehe Kapitel 4.4)
- die Flugzeugkonfiguration auf ein bestimmbares Ziel (z. B. die Minimierung der Betriebskosten) hin zu optimieren. (siehe Kapitel 4.5).

Konkret bedeutet dies, dass das Programm auf Basis der Vorgaben, der Anforderungen und unter Berücksichtigung der Randbedingungen überprüft, ob der Entwurf technisch realisierbar ist und ob er sich gegebenenfalls sogar noch verbessern lässt.

PrADO findet seinen Anwendungsbereich, mit seinem interdisziplinären Ansatz und den damit verbundenen sehr komplexen Zusammenhängen, in der Vorentwurfsphase.

2.2 Grenzen

Es ist mit dem Programm nicht möglich Detailkonstruktionen in fortgeschrittenen Entwurfsphasen durchzuführen. Außerdem stößt das Programm in der Phase des sehr frühen Entwurfs an eine Grenze, da viele hierfür unnötige Details definiert werden müssen, die dann noch zusätzlich die Rechnerzeit verlängern.

Hier würde eine Funktion Abhilfe schaffen, die viele Details durch grobe aber noch vertretbare Schätzwerte ersetzt und somit alle, für die Lauffähigkeit der nachfolgenden Module benötigten, Informationen zur Verfügung stellt.

Durch eine solche Möglichkeit, die die gesamte Rechenzeit erheblich verkürzen würde, wäre ein kreativeres Entwickeln und Ausprobieren und somit auch Verstehen und Mitdenken sehr gefördert.

Auch ist bei neuen *unkonventionellen* Flugzeugen ein schnelles Ausprobieren nicht möglich. Beispielsweise ist in der unveränderten PrADO-Version TA2 noch nicht realisierbar, bei einem geplanten autonom fliegenden Flugzeug, das ein beachtliches Potential zum Einsparen von Betriebskosten bietet, die Anzahl der Piloten ohne intensiven Modifikationsaufwand auf Null zu setzen.

Für die benötigten Modifikationen, die je nach Problemstellung tief in die Module eingreifen, ist neben viel Zeit auch einiges an Erfahrung mit dem Programm notwendig. Würde man die Anzahl der Piloten ohne weitere Änderungen auf Null setzen, dann würde z.B. die Einzelanalyse bei der Geometrieerzeugung abbrechen (siehe Bild 2.1).

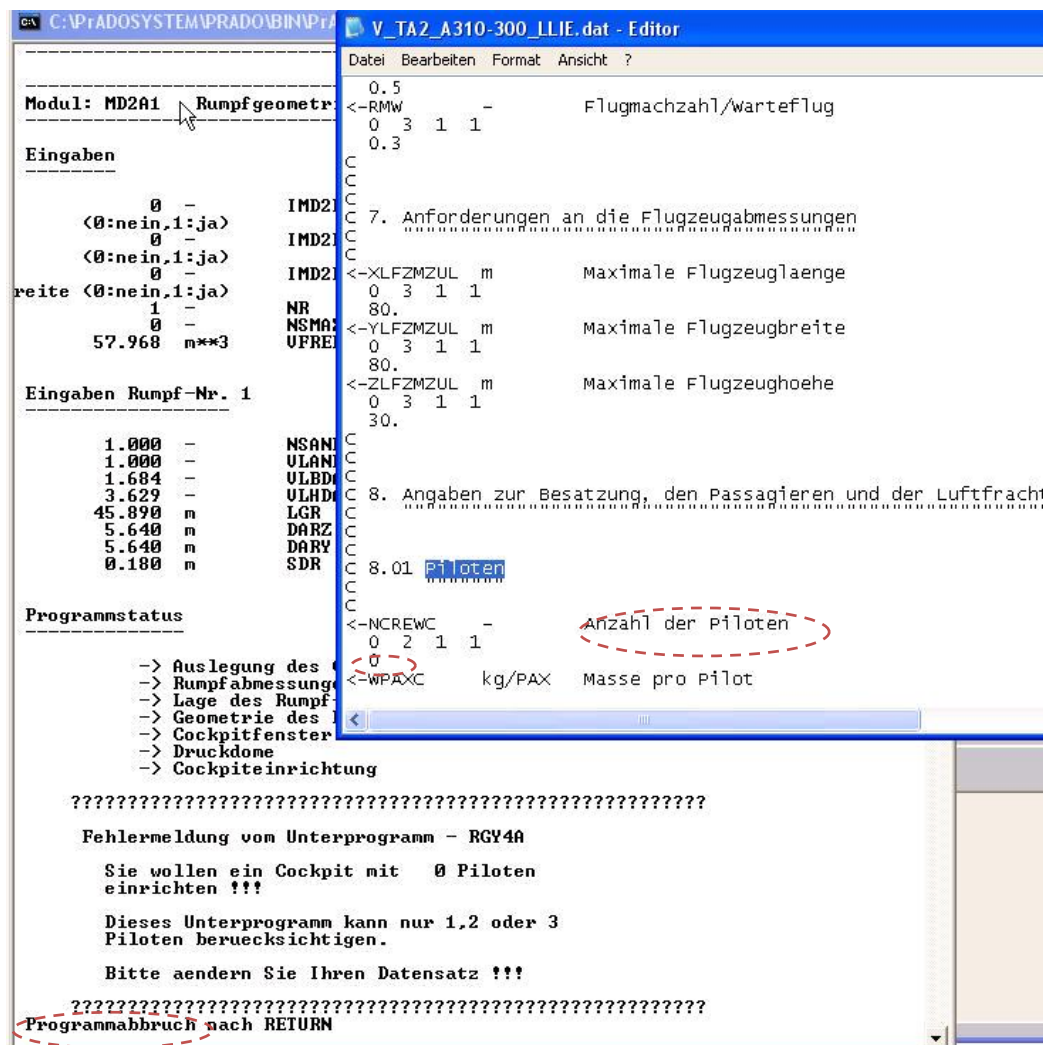


Bild 2.1 Abbruch bei der Geometrieerzeugung

Ein feineres Abstecken der Grenzen wird im Laufe dieser Arbeit vorgenommen.

2.3 Der Kern von PrADO

Der Programmkern besteht aus Modulen, die mit den vorgegebenen Randbedingungen die Teilaufgaben des Flugzeugentwurfes (u. a. Flugzeuggeometrie, Aerodynamik, Flugmechanik) abarbeiten.

Sie benutzen neben mathematischen und physikalischen Modellen in Form von Fortranprogrammen weitere integrierte, selbstständig arbeitende Programme.

Nähere Informationen zum genaueren Aufbau des Programmes sind im Kapitel 3.1 beschrieben.

2.3.1 Mathematische und physikalische Modellbildung

Bei den Zusammenhängen, die von den PrADO-Unterprogrammen benutzt werden, handelt es sich, neben statistisch gewonnenen Relationen und physikalischen Zusammenhängen (besonders aus dem Bereich der Statik, Dynamik und Festigkeitslehre), um Gleichungen aus geometrischen Gegebenheiten.

Nachfolgend werden alle ausfindig gemachten Methoden, Formeln bzw. daraus resultierende Verfahren mit Namen und falls bekannt mit Entstehungsjahr aufgelistet.² Die Verfahren wurden meistens mit dem Namen des jeweiligen Entwicklers benannt.

Fahrwerk:

- Liebermann / LTH

Triebwerk:

- Heinze (2000)
- Heinze/Leitner (1999)
- Heinze/Dechow (1991)
- Mattingly/Heinze (2006)

Massenverteilung:

- Baur/Wissel (1985)
- Burt-Philips
- DeHavilland (1980)
- Driggs
- General Dynamics
- Hansen (2000)
- Heinze (2001)
- Heinze (2000)
- Heinze (1999)
- Heinze/Leitner (2000)
- Howe (2001)
- Koeppen/FST Hamburg
- Liebermann (2001)
- LTH-Gewichte (2001)
- North-American Aviation
- Österheld
- Raymer (1990)
- Smith

² Diese Informationen sind größtenteils aus den Kommentarzeilen der Unterprogramme entnommen worden.

- Torenbeek (1990)
- Wittig

DOC-Kostenmodelle:

- Behrens (2003)
- IFL (1999) und (2000)
- Reim/Jahr 2006

Flugzeugsysteme:

Literaturquellen, aus denen die Verfahren entwickelt wurden:

- Calinski, Daniela; Carl, Udo B.; Koeppen, Carsten:
"Prognose des Leistungsbedarfs und der Masse elektrischer Bordnetze im Flugzeugentwurf (DGLR-JT2003-092)." In: DGLR (Hrsg.): DGLR Jahrbuch 2003
- Koeppen, Carsten; Carl, Udo B.:
"Erfassung und Bewertung von Systemen im Flugzeugentwurf (DGLR-JT2002-190)."
- Koeppen, Carsten; Carl, Udo B.:
"Kopplungen und Sensitivitäten des Einflusses von Systemen auf den Flugzeugentwurf (DGLR-JT2003-182)."
- Schulz, Alexander; Köppen, Carsten; Carl, Udo B.:
"Entwicklung eines Prognosemodells für Masse und Leistungsbedarf von Fahrwerken im Flugzeugvorentwurf (DGLR-JT2004-122)."
- Calinski, Daniela:
"Modellbildung, -verifikation und -optimierung von Systemkopplungen für das elektrische Bordnetzsystem bei Verkehrsflugzeugen." Projektarbeit des Arbeitsbereichs Flugzeug-Systemtechnik der Technischen Universität Hamburg-Harburg, 2003
- Krol, Sebastian:
"Model conception, -verification and -optimisation for a commercial aircraft environmental control system." Studienarbeit des Arbeitsbereich Flugzeug-Systemtechnik der Technischen Universität Hamburg-Harburg, 2004.
- Neumann, Uwe:
"Modellbildung, -verifikation und -optimierung von Systemkopplungen des hydraulischen Energieversorgungsystems für die primäre Flugsteuerung bei Verkehrsflugzeugen." Studienarbeit des Arbeitsbereich Flugzeug-Systemtechnik der Technischen Universität Hamburg-Harburg, 2002.
- Schulz, Alexander:
"Modellbildung, -verifikation und -optimierung von Systemkopplungen des Fahrwerks bei Verkehrsflugzeugen." Studienarbeit des Arbeitsbereich Flugzeug-Systemtechnik

der Technischen Universität Hamburg-Harburg, 2004.

Aerodynamik:

Literaturquellen, aus denen die Verfahren entwickelt wurden:

- E.Torenbeek: "Synthesis of Subsonic Airplane Design"
Delft University Press, 1981

Gesetze:

Gesetze nach FAR25 für:

- Maximales und minimales Lastvielfaches durch Böeneinwirkung
- Maximales Abfanglastvielfaches bei ausgefahrenen Klappen
- Charakteristischen Lastvielfache bei vorgegebene Geschwindigkeiten
- Bemessungsgeschwindigkeiten EAS bei Reiseflug, Sturzflug, Manövern, ein- und ausgefahrenen Klappen und bei den Böeneinfluss
- Maximales und Minimales Abfanglastvielfaches
- Lastvielfaches in der Hochauftriebsparabel
- Mindeststeigwinkel und -geschwindigkeiten bei Start und Landung
- Kenngrößen des V-n-Diagramms.
- Gangzahl und die Aufteilung der Sitze in einer Reihe

Umwelteinflüsse:

- Atmosphärenmodell nach DIN-ISO 2533

Unterstützende mathematische Methoden:

- Determinantenregel zum lösen von Gleichungssystem der 2. und 3. Ordnung
- Gleichungslöser
- Integration nach der Trapezformel
- Interpolationsverfahren
- Regression

Formeln zur Berechnung von:

- Fläche, die durch Punkte im Raum vorgegebenen werden
- Streckenlänge zwischen vorgegebenen Punkten
- Punkten auf einer Geraden im Raum
- Volumen und deren Schwerpunktlage
- Mantelflächen deren Schwerpunktlage
- Flächen deren Schwerpunktlage eines räumlich aufgespannten Dreieckes
- Rotationen eines Punkt um eine Achse

- Transformationen eines Punktes
- Vektordaten für Punkte einer auf einer gespannten Ebene
- Betrag und den Richtung eines Vektors
- Kreuzprodukten von Vektoren
- Schnittpunkten zwischen zwei Geraden, die senkrecht aufeinander stehen
- Punkten auf einer Geraden im Raum
- Winkeln zwischen Vektoren
- Schnittpunkte zweier Geraden im Raum.
- Vektorprodukte zweier Vektoren
- Skalarprodukte zweier Vektoren
- Flächen die von zwei Vektoren aufgespannt wird
- Matrixinversionen
- Determinantenberechnungen
- Rangberechnungen von Determinanten

Die mathematischen Modelle, die hinter der Optimierung stecken, werden im Kapitel 4 beschrieben.

Desweiteren sind in den Fortranprogrammen viele Verfahren enthalten, die vom IFL entwickelt wurden, aber keinen eigenen Namen besitzen. Hierzu zählen z.B.: Verfahren zur Lösung unkonventioneller Entwurfsaufgaben die, um die Besonderheiten zu erfassen, ihre Ergebnisse aus physikalisch begründbaren Modellvorstellungen ableiten.

2.3.2 Integrierte Programme

Neben den in *PrADO* enthaltenen Unterprogrammen sind für die volle Funktionalität von *PrADO*, außer den nachfolgend beschriebenen Programmen *Liftingline* und *Polint*, normalerweise auch noch die Programme *SAM*, *ZAERO* und *NASTRAN* integriert. Über diese Programme wird hier aber keine Aussage getroffen, da sie in der benutzten *PrADO*-Version nicht integriert waren (siehe Kapitel 2.5).

Liftinline:

Beschreibung

Lifting Line ist ein kostenlos erhältliches Programm, das über ein potentialtheoretisches Verfahren die aerodynamischen Beiwerte einer Konfiguration aus einer oder mehrerer beliebig angeordneter Tragflächen ermitteln kann.

Grenzen

Die Genauigkeit des Programmalgorithmus ist laut **Horstmann 2007** mit dem des Tragflächenverfahren vergleichbar und weist folgende Beschränkungen auf:

- Das verwendete Profil muss dünn sein.
- Die Strömung muss als reibungslos idealisiert werden.
- Es können nur inkompressible Strömung betrachtet werden.

POLINT:

Beschreibung

Mit dem Polareninterpolationsmodul *POLINT* können die Profilbeiwerte bei den Berechnungen von *Liftingline* berücksichtigt werden. (**Wunderlich 2007**)

Zusätzlich erweitert *POLINT* den Anwendungsbereich durch die Kompressibilitätskorrektur nach Göthert auf eine kompressible reine Unterschallströmung.

Dadurch ist es möglich, die aerodynamischen Beiwerte einer Konfiguration bei reibungsbehafteter Strömung abzuschätzen. *POLINT* arbeitet folgendermaßen:

- *POLINT* nimmt eine reine zweidimensionale Strömung in den Profilschnitten parallel zur Längsachse der Konfiguration an.
- Die Profilbeiwerte werden für die lokalen Mach- und Reynoldszahlen bilinear interpoliert.
- Die Beiwerte werden entsprechend der Anordnung der Profile auf den Tragflächen linear interpoliert.

Die Ergebnisse von *POLINT* bestehen aus lokalen und globalen Beiwerten der Konfiguration unter Berücksichtigung der Profilbeiwerte.

Grenzen

Neben allen Beschränkungen die für *Liftingline* gelten, bestehen noch die folgenden:

- Die Genauigkeit der Ergebnisse ist von der Vorgabe geeigneter Profilpolaren abhängig.
- Dreidimensionale Strömungseffekte werden bei der Interpolation der Profilpolaren nicht berücksichtigt.
- Der Einfluss der Flügelpfeilung auf den Wellenwiderstand bei transsonischer Strömung wird nicht berücksichtigt.

2.4 Interessante Möglichkeiten

An dieser Stelle sollen ein paar interessante Möglichkeiten aufgezeigt werden, was sich mit dem Programm untersuchen lässt. Das Programm *PrADO* ist gut geeignet für die Untersuchung von Detailfragen zu einer bestimmten Konfiguration. Einige mögliche Fragen wären:

- Welcher Triebwerkstyp (TL oder ZTL) wäre am wirtschaftlichsten?³
- Soll ausschließlich Fracht oder Passagiere transportiert werden?
- Wie groß sind die Vor- oder Nachteile auf die Struktur, die Kosten und das Gewicht bei einem Verzicht auf eine Druckbelüftung?
- Wie verhält sich der Lärmfußabdruck als Funktion des Steigwinkels?

³ Das nachbilden eines PTL ist in der aktuellen *PrADO*-Version noch nicht möglich.

- Ist ein dünnes Profil (um geringe Übergeschwindigkeiten zu bekommen) oder ein dickes Profil mit einem höheren FTM (Flächenträgheitsmoment) nötig, um eine ausreichend steife Struktur zu erlangen? Diese Fragestellung ist möglich, da sowohl die Aerodynamik als auch die Struktur in PrADO implementiert sind und miteinander interagieren.
- Wäre ein gemischter Kraftstoffeinsatz für verschiedene Flugphasen sinnvoll? Hinweis: Diese lässt sich in der aktuellen PrADO-Version nur manuell für jede Flugphase, durch verändern des Datensatzes, durchführen.
- Kann mit Wasserstoff, bei der notwendigen Lagerung in Zusatztanks und den damit verbundenen aerodynamischen Nachteilen, ein wirtschaftlicher oder ökologischer Nutzen im Vergleich zum Kerosin erreicht werden?

Desweiteren können mit dem Programm auch unkonventionelle Konfigurationen bewertet werden beispielsweise kann *PrADO* mit mehreren Rümpfen, Leitwerken oder Tragflächen umgehen. Diese Fähigkeit konnte im Laufe dieses Projektes allerdings nicht getestet werden. Einige Beispiele sind, neben der konventionellen Variante, im Bild 2.2 abgebildet.

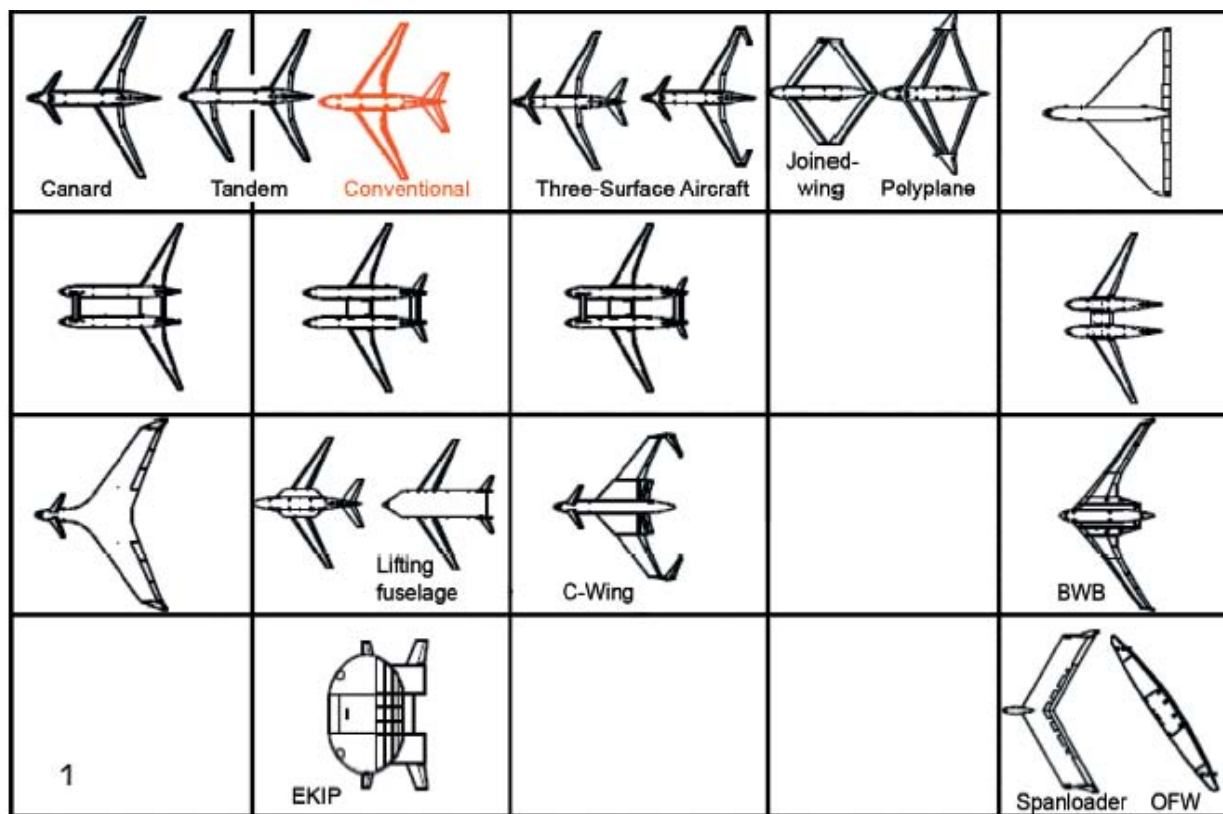


Bild 2.2

Unkonventionelle Konfigurationen (Trahmer 2004)

2.5 Informationen zu der benutzten PrADO-Version

Bei der für diese Arbeit verwendete *PrADO*-Version handelt es sich um die Version TA2 vom April 2007 mit einem Update vom 01.12.06.

Die *PrADO*-Version, die das IFL der HAW zur Verfügung gestellt hat und mit der während dieses Projektes gearbeitet wurde, beinhaltet nicht den wichtigen Programmteil mit der Struktur- und Aerodynamiksoftware SAM. Somit war es nicht möglich, eine Aussage über die Funktionsweise oder die Genauigkeit von SAM zu treffen. Ebenfalls fehlte das mit dem Strukturanalyse-Programm *Nastran* kommunizierende Programm IFL2Nast.

Auch war es nicht möglich, genaue aeroelastische Stabilitätsanalysen durchzuführen, da weder das dafür notwendige Programm *Zaero* von Zonatech⁴ noch das anbindende Programm IFL2Zaero vorhanden waren. Dieses Programm wurde, in einigen Dokumenten über *PrADO*, für die Flatteranalyse verwendet.

⁴ Weitere Informationen zu Zonatech und Zaero siehe : www.zonatech.com

3 Allgemeines

3.1 Ordern und Dateien von PRADO

In diesem Abschnitt wird die Programmstruktur kurz erläutert. Dadurch soll der Kontext, zwischen den, in dieser Arbeit behandelten, Dateien und dem gesamten Programm hergestellt werden.

Genauere Informationen über den Aufbau findet man in der Arbeit „Structure and function of the aircraft design program PRADO“ von Lionel Salavin. Dieser Arbeit ist das folgende Bild, zum Erlangen einer ersten Übersicht, entnommen. In diesem Bild ist der Inhalt des gesamten PRADO-Programmordners dargestellt.

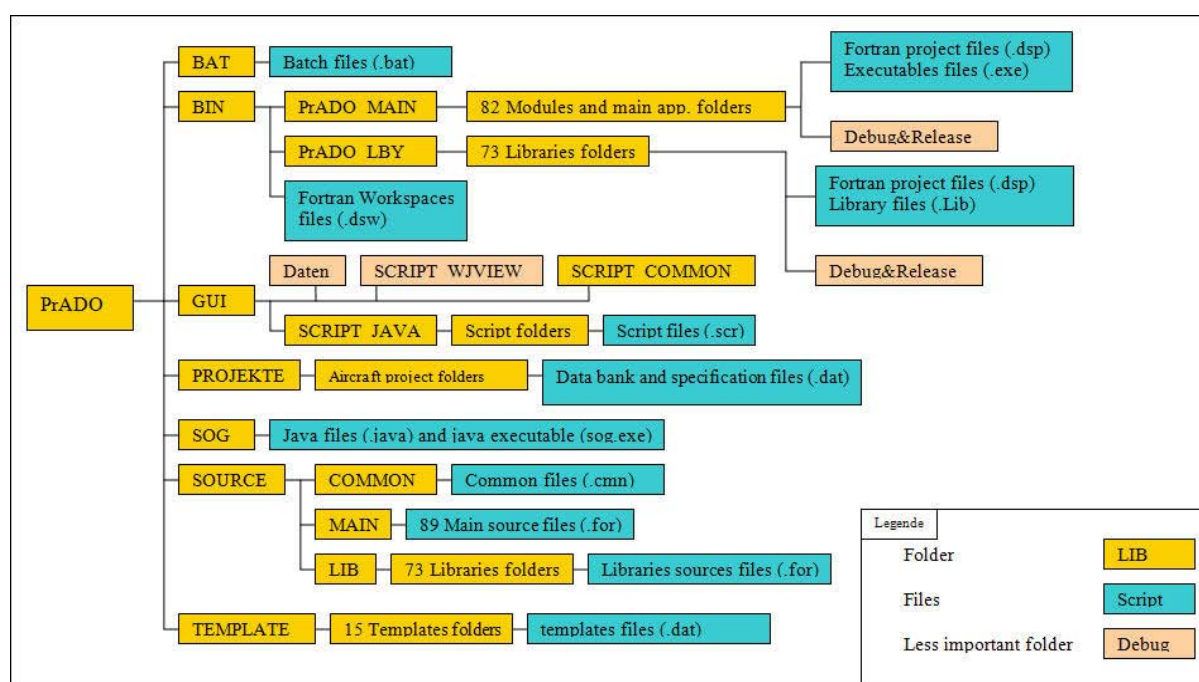


Bild 3.1 Der Programmbaum (nach Salavin 2008)

Die in Bild3.1 enthaltenen *PrADO*-Hauptordner haben folgenden hauptsächlichen Inhalt:

BAT:

Hier finden sich alle, von den Java-Oberflächen aufgerufenen, Stapeldateien wieder.

GUI:

In diesem Ordner befinden sich die Dateien der Java-Oberflächen.

PROJEKTE:

Hier finden sich die Projektordner wieder, auf die im Kapitel 3.2 eingegangen wird.

SOG:

In diesem Ordner sind Java-Dateien abgelegt, die die Java-Oberflächen im Ordner „GUI“ aufrufen.

SOURCE:

Hier befinden sich die FORTRAN-Programmtexte der einzelnen Unterprogramme.

Dokumentation:

Dieser Ordner enthält Doktor- und Studienarbeiten im pdf-Format.

BIN:

Beim Kompilieren des Quellcodes werden in diesem Ordner die ausführbaren „exe“-Dateien hinterlegt.

3.2 Der Projektordner

3.2.1 Ursprung der Dateien im Projektordner

Hier soll geklärt werden, was im Projektordner enthalten ist und wann es erstellt wird. Um die Übersichtlichkeit zu erhöhen, folgen die Beschreibungen der einzelnen Dateien erst anschließend im Kapitel 3.2.2.

Nach dem Anlegen eines Projekt-Ordners muss man hier die Vorgabedatei hinterlegen. Die Vorgabedateien werden folgendermaßen bezeichnet (diese Bezeichnungen sind nicht zwingend erforderlich):

- V_Programmversion_Flugzeugname.dat
- V_Programmversion_Flugzeugname_LLIE.dat
- V_Programmversion_Flugzeugname_SAM.dat

Durch das aufrufen der „Projektdefinition“ im *MainWindow* (siehe Kapitel 3.3), der Eingabe der Projekt-Ordner-Adresse und dem anschließenden starten der Hilfe „**Vorgabedatei einlesen**“ werden folgende *Ordner* erzeugt:

- aerodynamic
- plots
- xml

Desweiteren werden folgende, in der benutzten PrADO-Version funktionslose *Ordner* erzeugt:

- nastran
- zaero
- ssm

Nach dem beendeten Einlesen der Vorgabedatei sind folgende zusätzlichen *Dateien* im Projektordner enthalten:

- PRA-db1.dat
- PRA-db2.dat
- ...

Nach dem Aufrufen der Hilfe „**Kurzprotokoll**“:

- KP.dat (Hinweis: nur mit einem „P“)
- PRADO-Protokoll.dat

Nach dem Durchführen der Hilfe „**Flugzeuggeometrie**“:

- PRADO-Protokoll.dat

Durch das Betätigen der **Einzelanalyse** entstehen die *Dateien*:

- GEIT.dat
- PRADO-Protokoll.dat
- KP.dat (Hinweis: nur mit einem „P“)

Nach dem Durchlaufen der **Parametervariation**:

- GEIT.dat
- PRADO-Protokoll.dat

- PRA-BASIS-db1.dat
- PRA-BASIS-db2.dat
- ...

- GPVAR_A.dat
- GPVAR_B.dat
- ...

- KPP.dat (Hinweis: mit „PP“)

Nach dem Durchlaufen der **Optimierung**:

- GEIT.dat
- PRADO-Protokoll.dat

- PRA-BASIS-db1.dat
- PRA-BASIS-db2.dat
- ...

- GOVAR_A.dat
- GOVAR_B.dat
- ...

- KPO.dat

Zusätzlich entsteht seit dem PrADO-Update vom 01.12.06 nach einem E/P/O die folgende *Datei*:

- TWKENF_T1F1.dat

Hinweis:

Alle hier aufgeführten Dateien in dem Projektordner, müssen für ihre Funktionalität die Endung „.dat“ besitzen.

3.2.2 Die Funktion der einzelnen Dateien

An dieser Stelle soll eine Übersicht über die, in den Dateien des Projektordner enthaltenen, Informationen gegeben werden.

GEIT.dat :

In dieser Datei werden die Informationen über die Iterationsschritte der Einzelanalyse gespeichert. Die GEIT-Datei wird solange nach jedem Iterationsschritt mit den neu hinzugekommenen Werten überschrieben, bis das Iterationskriterium erfüllt ist. In dem, im folgenden Bild 3.2 zu sehenden, Ausschnitt einer „GEIT.dat“-Datei war die erlaubte Differenz zwischen den Schritten 0,01. In diesem Fall wurde die maximale Differenz für alle Variablen schon nach dem vierten Iterationsschritt erreicht, wodurch die Einzelanalyse erfolgreich beendet wurde.

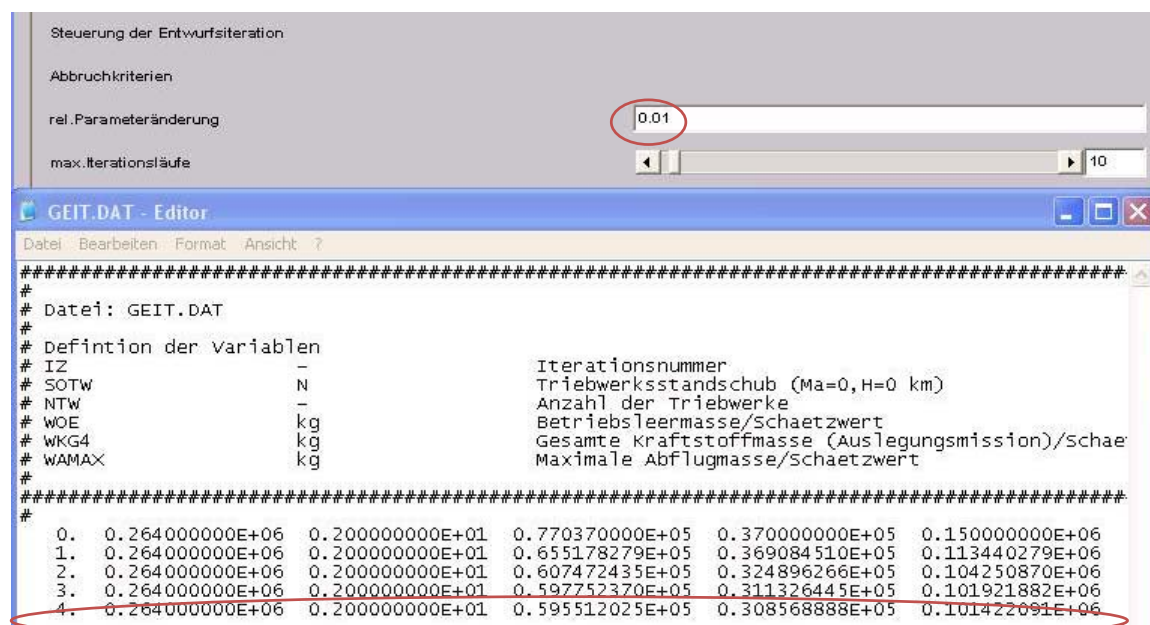


Bild 3.2 GEIT.dat

GPVAR_A.dat:

Diese Datei ist wie die Datei GEIT.dat aufgebaut. Der Unterschied besteht darin, dass es sich nicht um die gespeicherten Informationen einer Einzelanalyse handelt, sondern um die einer Parametervariation.

GPVAR_B.dat:

Hierbei handelt es sich im Gegensatz zu der Datei GPVAR_A.dat um mehrdimensionale Parametervariationen.

GOVAR_A.dat:

Diese Datei ist wie die Datei GEIT.dat aufgebaut. Der Unterschied besteht darin, dass es sich nicht um die gespeicherten Informationen einer Einzelanalyse handelt, sondern um die einer Optimierung.

GOVAR_B.dat:

Hierbei handelt es sich im Gegensatz zu der Datei GOVAR_A.dat um mehrdimensionale Optimierungen.

KP.dat :

In dieser Datei sind alle wichtigen Entwurfsdaten zusammengefasst, wie z.B. die geometrischen Abmessungen, die Betriebsleermassen, usw.

KPO.dat:

In dieser Datei sind, wie in der Datei KP, alle wichtigen Entwurfsdaten zusammengefasst. Allerdings handelt es sich hierbei um die Entwurfsdaten nach einer Optimierung.

KPP.dat:

In dieser Datei sind, wie in der Datei KP, alle wichtigen Entwurfsdaten zusammengefasst. Allerdings handelt es sich hierbei um die Entwurfsdaten nach einer Parametervariation.

PRA-BASIS-dbi.dat:

Diese Datei stellt einen Teil der Datenbank dar. Sie enthält alle Entwurfsgrößen inklusive aller Befehlszeilen, die aus der Vorgabedatei und weiteren Berechnungen erzeugt wurden. Diese Datei wird während der Parametervariation oder der Optimierung nicht überschrieben.

PRA-dbi.dat:

Diese Datei stellt ein Teil der Datenbank dar. Sie enthält alle Entwurfsgrößen, die aus der Vorgabedatei und weiteren Berechnungen erzeugt wurden. Diese Datei wird während der Parametervariation oder der Optimierung immer wieder mit neuen Werten überschrieben.

PRADO-Protokoll.dat:

Bei der E/P/O wird in dieser Datei jeweils der gesamte durchgeführte Programmverlauf mit allen Einzelschritten hinterlegt.

TWKENF_TiFi.dat:

In dieser Datei ist das Triebwerkskennfeld abgebildet. Hierin finden sich folgende, voneinander abhängige Informationen wieder:

- der Drosselgrad in %
- die Flughöhe in km
- die Flugmachzahl
- der Triebwerksschub in N
- der spez. Verbrauch in kg/N/h
- der Emissionsindex NOx in g/kg
- der Emissionsindex CO in g/kg
- der Emissionsindex HC in g/kg
- der Emissionsindex H2O in g/kg

Für die Ausgabe der Emissionen müssen die Triebwerksmodelle allerdings erst noch angepasst werden.

Warnung:

Bei mehrfachem Starten von z.B. dem Einlesen der Vorgabedatei, der Parametervariation oder der Optimierung werden alle bis dahin im Projekt-Ordner enthaltenen Dateien, ohne Nachfrage, überschrieben!

3.3 GUI Oberflächen

Um das Programm über eine GUI-Oberfläche zu steuern, wurde als grafische Schnittstelle zum Anwender eine leicht zu modifizierende Java-Oberfläche gewählt.

In den bereits existierenden GUI-Oberflächen sind allerdings nicht alle in PrADO enthaltenen Unterprogramme integriert.

Hinweis:

Die Werte, die beim Starten in der *GUI* stehen, sind die Werte, die noch im Ordner *Daten*⁵ in den „.dat“-Dateien abgelegt waren.

⁵ Dieser Ordner befindet sich unter dem folgenden Pfad: C:\PrADOSYSTEM\PRADO\GUI

3.3.1 Vorhandene Benutzeroberflächen

Die nachfolgend aufgezählten und funktionierenden GUIs gibt es bereits:

- *PrADO* TA2-A
- Missionssimulation
- Optimierung Verwindung
- Reichweitenberechnungen
- Widerstandsberechnung
- Start-Berechnungen 2

Neben der „Start-Berechnungen 2“-GUI gibt es auch noch „Start-Berechnungen“ und „Start-Berechnungen1“. Bei diesen beiden Oberflächen handelt es sich jedoch zum einen um dieselbe aufgerufene Datei und zum anderen sind die damit ansprechbaren Funktionen auch in der „2er“ enthalten.⁶

PrADO TA2-A

PrADO TA2-A ist die eigentliche GUI-Oberfläche oder auch das „*Main Window*“ mit den meisten eingearbeiteten Funktionalitäten. Sie wurde im Kapitel 4.1 der Arbeit „*Methodisches Entwerfen von Verkehrsflugzeugen mit PrADO*“ von Thorsten Kiesel schon umfassend beschrieben und soll aus diesem Grund hier nicht weiter beschrieben werden.

Missionssimulation

Diese GUI-Oberfläche bietet die Möglichkeit einzelne Flugmissionen zu simulieren.⁷

Optimierung Verwindung

Diese GUI-Oberfläche (siehe Bild 3.3) spricht die Unterprogramme zur Berechnung einer Optimierung der Flügelverwindung an.

Mit dieser Startoberfläche lassen sich auch Informationen zu einzelnen Anstellwinkeln errechnen.

⁶ Diese Benutzeroberflächen befinden sich unter: C:\PrADOSYSTEM\PRADO\GUI\SCRIPT_JAVA

⁷ Diese GUI ist in einenem eigenen Verzeichniss zu finden. Der Ort lautet:
C:\PrADOSYSTEM\PRADO\GUI\SCRIPT_JAVAMISSION

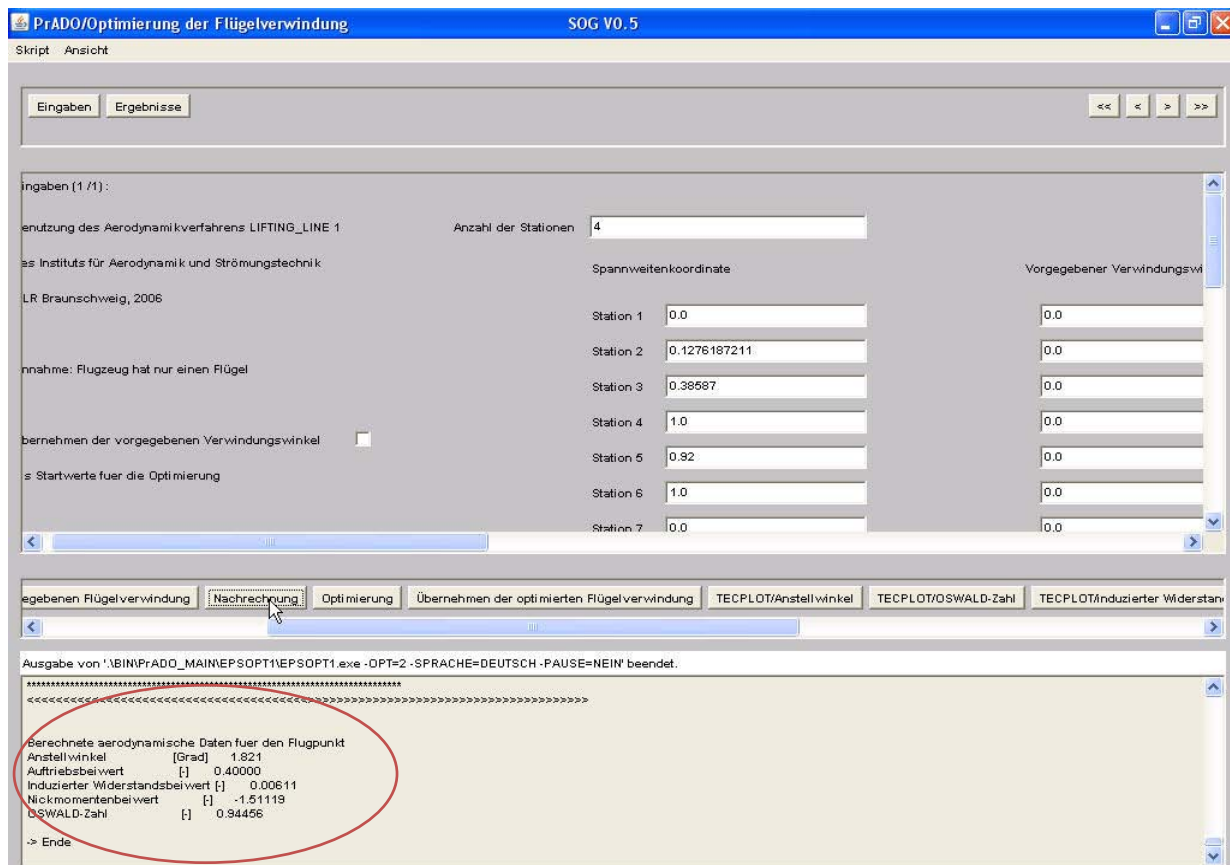


Bild 3.3 Optimierung der Verwindung

Desweiteren lässt sich die Auftriebsbeiwertverteilung, die Zirkulationsverteilung und sogar die Verwindungswinkelverteilung (siehe Bild 3.4) jeweils über die dimensionslose Spannweitenkoordinate darstellen.

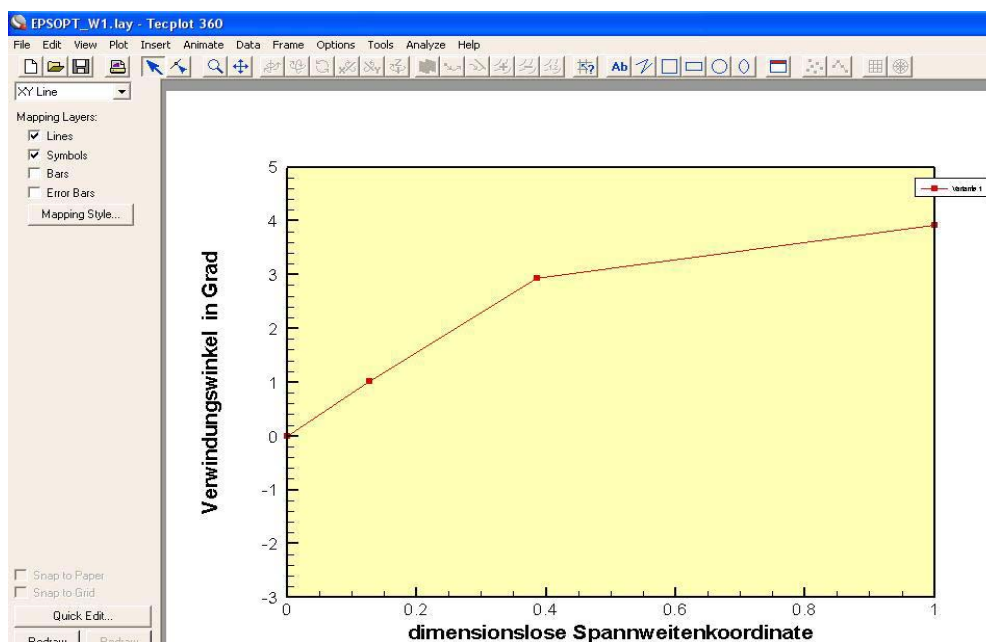


Bild 3.4 Verwindungswinkelverteilung

Reichweitenberechnungen

Diese GUI-Oberfläche dient zur Berechnung der Reichweite bei vorgegebener Flugzeugmasse.

Widerstandsberechnung

Diese GUI-Oberfläche spricht Unterprogramme zur Berechnung des Nullwiderstandes an. Für das Benutzen dieser Start-Oberfläche muss vorher die Vorgabedatei in der oben bereits beschriebenen „PrADO TA2-A“-GUI eingelesen worden sein, da sonst wie in Bild 3.5 die folgende oder eine ähnliche Fehlermeldung auftritt.

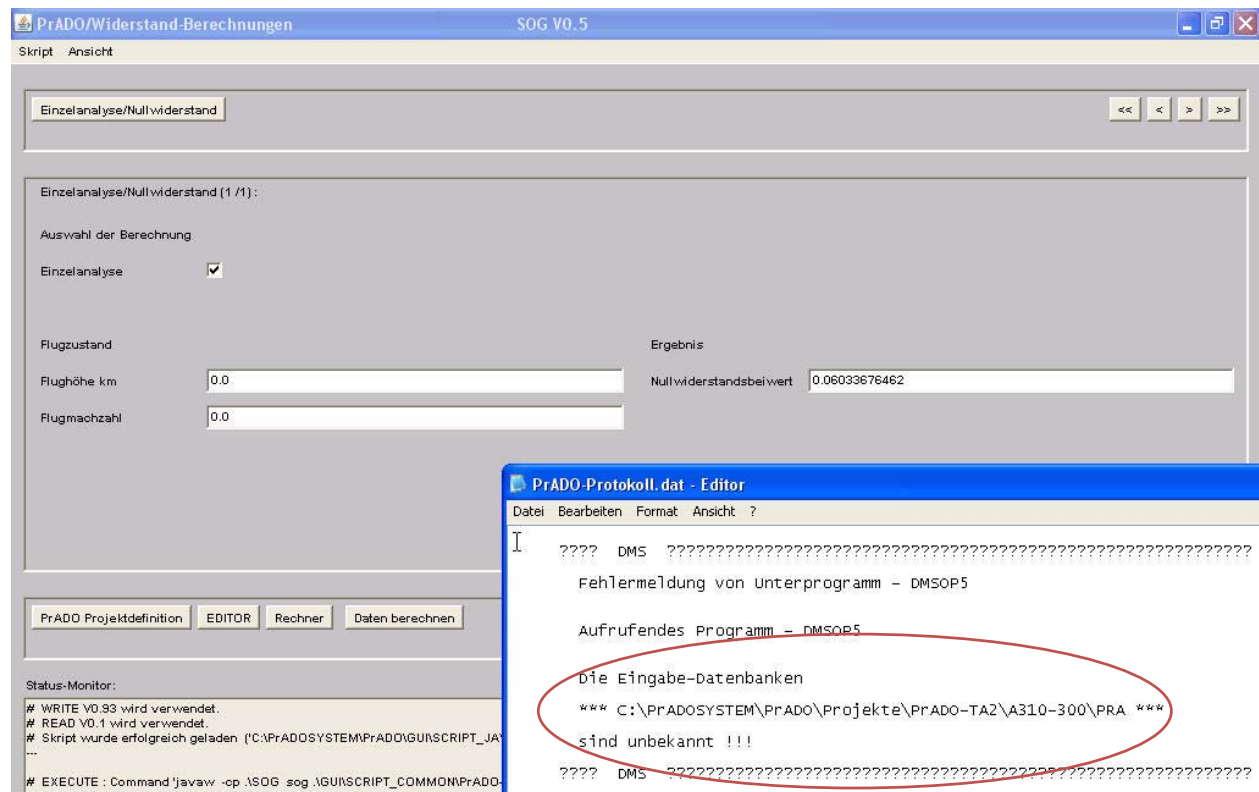


Bild 3.5 Widerstandsberechnung

Start-Berechnungen 2

Diese GUI-Oberfläche dient zur Ansteuerung von Unterprogrammen, die die Berechnung der Startbahnlänge durchführen. In dieser Oberfläche lassen sich auch Einzelanalysen durchführen.

3.3.2 Starten von Java-Oberflächen

Falls es noch keine Verknüpfungen zu den Java-Oberflächen-Dateien gibt, soll nachfolgend beschrieben werden, wie sich diese SCR-Dateien starten lassen.⁸

⁸ Diese SCR-Dateien befinden sich in dem Ordner: C:\PrADOSYSTEM\PRADO\GUI\SCRIPT_JAVA

1. An einer gewünschten Stelle mit der rechten Maustaste das Kontextmenü aufrufen und anschließend über „Neu“ und „Verknüpfung“ eine neue Verknüpfung erstellen.
2. Anschließend muss in die Zeile „Ziel“ in den Eigenschaften der Verknüpfung noch die folgende Zeile eingegeben werden:
`C:\WINDOWS\system32\javaw.exe -cp %PrADOHOME%\SOG sog %PrADOHOME%\GUI\SCRIPT_JAVA\PrADO-METHODEN\RANGE1-GUI.SCR`
3. Am Ende der Zeile fügt man den Namen der gewünschten Java-Oberfläche ein. Die im Beispiel (hier rot gedruckt) angesprochene Java-Oberfläche ist die der Reichweitenberechnung.

3.3.3 Anpassen der GUI-Oberfläche

Neben dem Masterfile für die Vorgabedatei gibt es auch eine Musterdatei für eine Java-Oberfläche.⁹

Diese Musterdatei ist sehr selbsterklärend geschrieben und lässt sich somit sehr leicht an die eigenen Bedürfnisse anpassen. Nach dem Öffnen beispielsweise mit dem Editor sieht man folgendes:

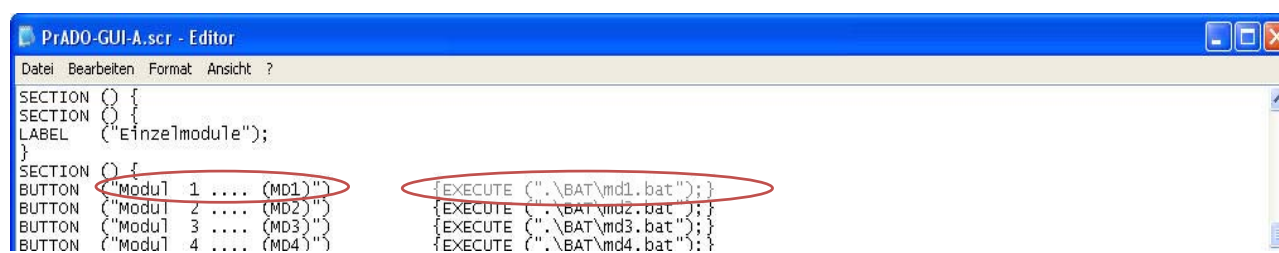


Bild 3.6 Musterdatei

In dem obigen Bild 3.6 sieht man:

- den Name der Schaltfläche, also *Modul 1 (MD1)*
- den Ort der beim Drücken der Taste zu startenden Stapeldatei, also *.\\BAT\\md1.bat*¹⁰

Durch beliebiges Ändern des Namens und der bei Betätigen der Taste von der Java-Oberfläche zu startenden Bat-Datei, lässt sich die Musteroberfläche anpassen. Die rohe Java-Oberfläche ist im Bild 3.7 zu sehen.

⁹ Diese Musterdatei ist am folgenden Speicherort abgelegt:
C:\PrADOSYSTEM\PRADO\GUI\SCRIPT_JAVA\SCR-MUSTER

¹⁰ Diese Bat-Datei ist eine der Stapeldateien aus dem folgenden Verzeichniss:
C:\PrADOSYSTEM\PRADO\BAT

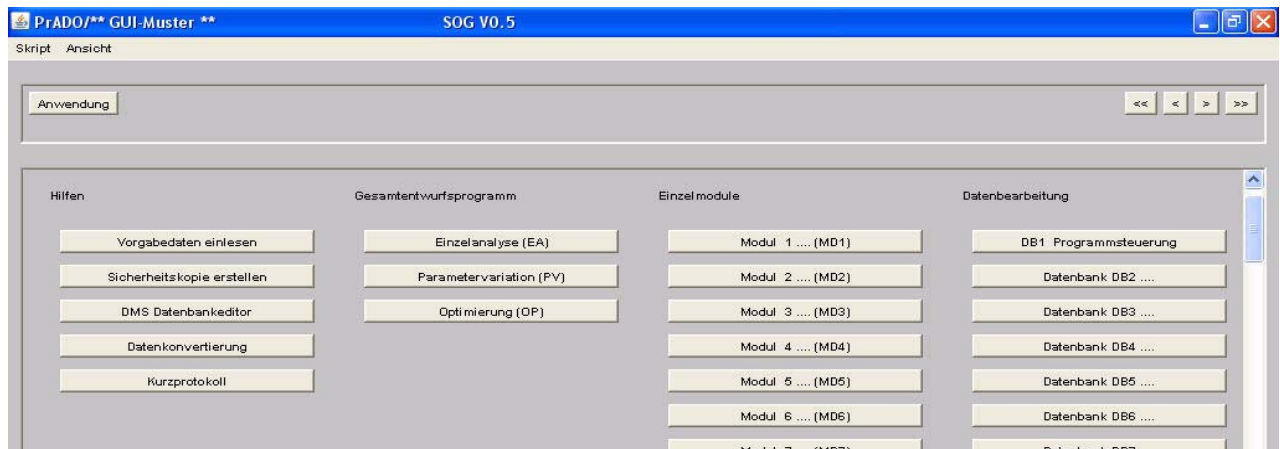


Bild 3.7 GUI-Masterdatei

Neben einer „Bat“-Datei, die z.B. im *MainWindow* von den Tasten der Einzelmodule gestartet wird, kann man auch andere Java-Oberflächen aufrufen (siehe Bild 3.8).

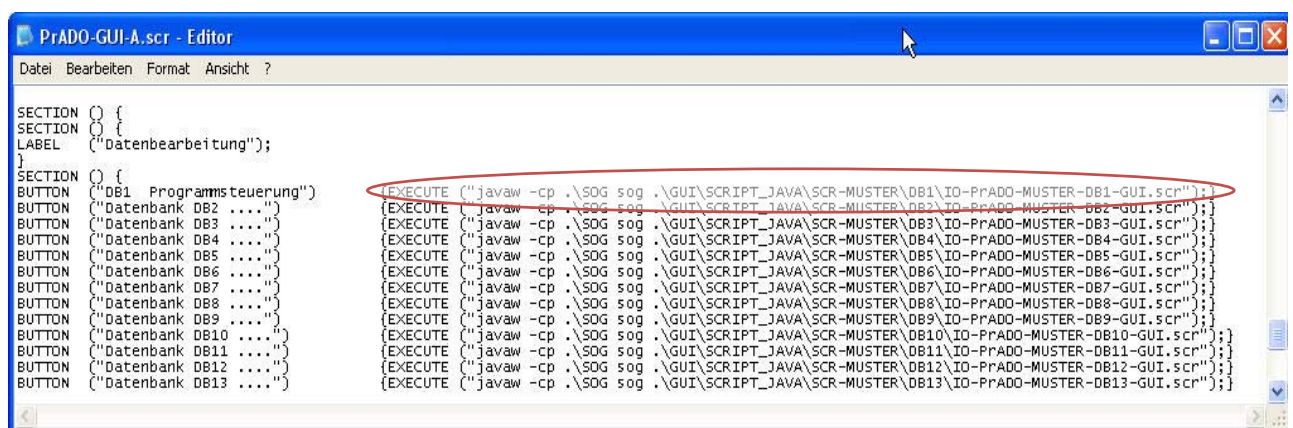


Bild 3.8 SCR-MUSTER

3.3.4 Programme ohne GUI ausführen

Die Unterprogramme lassen sich auch manuell ohne eine GUI starten.¹¹

Es lässt sich beispielweise das Programm „2DKURVEA.exe“ manuell ausführen. Die hierbei noch benötigten Informationen, z.B. für die gewünschte darzustellende Flughöhe, wird während des Programmlaufes abgefragt. Die zu tätigen Eingaben sind im Bild 3.9 durch Fragesätze zu erkennen.

¹¹ Die ausführbaren Programme befinden sich in dem Ordner:
C:\PrADOSYSTEM\PRADO\BIN\PrADO_MAIN

```

-----
Programm: 2DKURVEA
Projekt: AIR72-200
Triebwerk: General Electric CF34-3B1
-----

*** Diagramm: Masse-Reichweite ***
*** Diagramm: Nutzlast-Reichweite ***
*** Diagramm: Flugsimulation H = f(Ma) ***

Bitte waehlen Sie
(1) Flug mit max.Nutzlast
(2) Flug mit max.Kraftstoff
(3) Ueberfuehrungsflug
(4) Auslegungsmission
=> ? 4

*** Diagramm: WK,TF = f(R) ***
*** Diagramm: U-n-Darstellung ***
Flughoehe [km] => ? 6
Betriebsleermasse 23458.76 kg
maximale Abflugmasse 34037.46 kg
Startmasse [kg] => ? 30000
maximale Kraftstoffmasse 8036.08 kg
verbrauchte Kraftstoffmasse [kg] => ? 4000

Flugzeugmasse 26000.00 kg

FAR25-Daten
** Klappen eingefahren **
Lastvielfache
n+ : 2.500000000000000
n- : -1.000000000000000
Bemessungsgeschwindigkeiten (EAS)
US(CAMAX) [km/h] : 202.686029016949
US(CAMIN) [km/h] : 202.686029016949
UA(CAMAX) [km/h] : 320.474750794270
UA(CAMIN) [km/h] : 202.686029016949
UB [km/h] : 305.109704735531
UC [km/h] : 668.762694392892
UD [km/h] : 749.485607453784
Boengeschwindigkeit
UDE(UB) [m/s] : 20.0000000000000
UDE(UC) [m/s] : 15.0000000000000
UDE(UD) [m/s] : 7.600000000000000
** Klappen ausgefahren **
Lastvielfache
n+ : 2.000000000000000
Bemessungsgeschwindigkeiten (EAS)
US(CAMAX) [km/h] : 153.121806673985
UA(CAMAX) [km/h] : 216.546935693421
UF [km/h] : 394.074457060091

*** Diagramm: Massenaufteilung/leeres Flugzeug ***
*** Diagramm: Sichtwinkel der Piloten ***

Der Rumpf hat 2 Pilotensitze.
Bitte waehlen Sie den Pilotensitz aus: 1

*** Diagramm: Querschnittsflaechenverteilung ***

Anzahl der Stuetzpunkte [-] ? 10

2DKURVEA/2002 ... Ende ...
-----
Weiter nach RETURN

```

Bild 3.9 2DKURVEA.exe

3.4 Ergänzungen und Hinweise

3.4.1 Allgemeines

Der Name der Sicherheitskopie beginnt nicht wie in Kapitel „4.3.1 Hilfen“ von T. Kiesel beschrieben automatisch mit „S-Kop...“, sondern besteht ausschließlich aus dem Namen, den man eingibt.

3.4.2 Fehler während der Installation

Nach der Installation funktionierte das Lärmanalyse-Tool nicht, da die Namen der Dateien falsch geschrieben waren (siehe Bild 3.10). Nach dem Ändern der \ddot{a} in \ae funktionierte es einwandfrei.

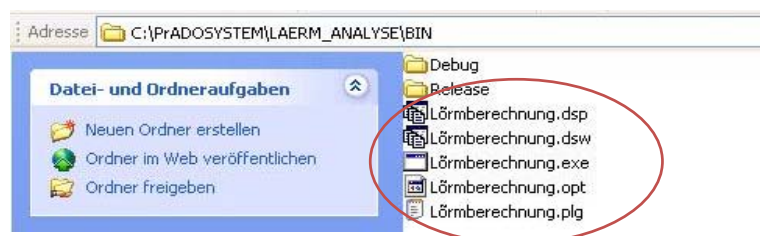


Bild 3.10 Lärmanalyse-Tool

Ein Fehler beim Installieren kann ggf. sein, dass die in den Systemordner *PRADOSYSTEM* kopierten Dateien nach dem Hineinkopieren noch einen Schreibschutz haben. Dieser wird unter Eigenschaften entfernt.

3.4.3 Anpassen des modularen Aufbaus

Es dauert einige Zeit, bis man den Aufbau des Programmes so verinnerlicht hat, dass man neue Module funktionstüchtig einfügen kann, ohne dass es zu Fehlern in anderen Bereichen kommt.

Bezüglich Anleitungen und Empfehlungen darüber, wie man in das „Getriebe“ von PrADO eingreift, um neue Module einzuarbeiten, möchte ich auf die sehr übersichtliche Darstellung in der Studienarbeit *“Structure and function of the aircraft design program PrADO“* von Linoel Salavin verweisen.

3.4.4 Der Sinn der Technologiefaktoren

Die häufig auftretenden Technologiefaktoren erlauben es, den Einfluss fortschrittlicherer Bauweisen oder neuer Werkstoffe zu berücksichtigen.

3.4.5 „BI“ und „AI“

Mit diesen Ergänzungen an der Kennzeichnung des Entwurfsmoduls (MDi) kann man das „Anstarten“ des Modules beeinflussen z.B.: BI*MD1
Somit lassen sich Freiräume in den Datenbank-Dateien vermeiden.

4 Die eigentliche Aufgabe von PrADO

In diesem Kapitel geht es um die wesentlichste Aufgabe von PrADO. Diese Aufgabe besteht aus der Einzelanalyse, der Parametervariation und der Optimierung (E/P/O). Es wird hier nur der Sinn und die Arbeitsweise dieser Funktionen beschrieben. Wie sich die Ergebnisse kontrollieren und darstellen lassen, wird im Kapitel 5 geschildert.

4.1 E/P/O allgemein

Alle für die E/P/O bedeutsamen Werte befinden sich, am Anfang der Vorgabedatei, in der ersten Datenbank.

4.2 Die Iteration

Wie bereits in dem Kapitel 2 erwähnt, ist der multidisziplinäre Entwurf nur mit iterativ ablaufenden Berechnungen möglich. Dies ist notwendig, da sich viele Ziele widersprechen bzw. von einander abhängig sind.

4.3 Die Einzelanalyse

Die Einzelanalyse überprüft, ob die vorgegebene Konfiguration, in Form der Vorgabedatei, nach dem heutigen Technologiestandard überhaupt machbar ist.

Dieser Vorgang läuft iterativ. Er wird also, unter Berücksichtigung der nach jedem Durchgang geänderten Iterationsvariablen, solange durchgeführt, bis das Ergebnis konvergiert. Sollten die Rechenschritte keine Konvergenz aufweisen, so ist die Konfiguration nicht realisierbar.

Vor der Einzelanalyse müssen die Variablen, die auf Konvergenz geprüft werden sollen, in der GUI- „Programmsteuerung“ oder direkt in die Vorgabedatei eingegeben werden. Ein solcher Datensatz muss folgendermaßen aussehen: SOTW 8 1.

Hierbei gibt die „8“ einen Hinweis darüber, in welcher Datenbank sich die Variable befindet. Die zweite Zahl, also in diesem Fall die „1“ beinhaltet die Information, dass es sich bei der Variablen „SOTW“ um einen Skalar handelt.

Nach dem Festlegen der Variablen läuft die Einzelanalyse folgendermaßen ab:¹²

¹² Die Schritte zur Beschreibung der Vorgänge während einer Einzelanalyse, wurden aus der „PrADO-Protokoll.dat“ abgeleitet. In dieser Datei sind alle Details der Einzelanalyse aufgezeichnet.

1. Aus den Eingaben werden die wichtigsten voneinander abhängigen Entwurfsgrößen aufgebaut. Desweiteren werden die Anforderungen überprüft und ergänzt.
2. Aus den eben gewonnenen Daten wird die dreidimensionale Geometrie des gesamten Flugzeuges berechnet.
3. Für die jetzt vorliegenden Werte wird, durch die Beschreibung des Kreisprozesses, der benötigte Standschub und das Triebwerkskennfeld ermittelt. Die Ergebnisse hieraus sind die Massen und der spezifische Verbrauch des Triebwerks.
4. Auf Basis der bis hierher vorliegenden Werten werden jetzt Flugsimulationen für die verschiedenen Missionen durchgeführt. Die verschiedenen Missionen bestehen neben der Auslegungsmission, aus dem Flug mit maximaler Nutzlast, maximalem Kraftstoff und aus dem Überführungsflug.
5. Aus den dazugehörigen verschiedenen Flugphasen wird neben den Flugleistungen auch der Kraftstoffverbrauch berechnet.
6. Jetzt wird die Struktur des gesamten Flugzeuges dimensioniert.
7. Nun folgt die Berechnung der Massen und der dazugehörigen Schwerpunkte der einzelnen Komponenten.
8. Jetzt werden die globalen Massen des Flugzeuges ermittelt.
9. Überprüfung der Landebahn- und Startbahnlänge.
10. Berechnung der Betriebskosten inklusive der Personal-, Wartungs-, Flugzeug- und Treibstoffkosten und den Gebühren
11. Jetzt erfolgt eine Prüfung der Variablen auf Konvergenz.

Diese Schritte werden so oft mit den immer wieder neuen Startwerten wiederholt, bis die abhängigen Variablen für die Entwurfsiteration konvergieren.

Sobald die beiden in den Anforderungen gewählten Iterationskriterien (siehe Bild 4.1) erfüllt sind, bricht die Einzelanalyse ab.

Variable	Wert	Wertaenderung i-1 -> i Iteration		Konvergenz
		absolut	relativ	
SOTW	0.264000E+06	0.000000E+00	0.000000E+00	<- ja
NTW	0.200000E+01	0.000000E+00	0.000000E+00	<- ja
WOE	0.777157E+05	0.678662E+03	0.880956E-02	<- ja
WRG4	0.371778E+05	0.177834E+03	0.480633E-02	<- ja
WAMAX	0.149856E+06	-0.143504E+03	-0.956693E-03	<- ja
*** Iterationsabbruch ***				
Summe/Variablen-differenzen	DDD =	0.00000E+00		<- erfuehlt
Iterationsschritte	IZI =	1	< Maximum = 50	<- erfuehlt
*** Rechenzeitstatistik ***				

Bild 4.4.1

Iterationskriterien

Zu den benutzbaren Entwurfsvariablen gehören z. B. die folgenden Werte:

- Triebwerksstandsdruck
- Anzahl der Triebwerke
- Betriebsleermasse (beginnt im ersten Iterationsschritt mit einem Schätzwert)
- gesamte Kraftstoffmasse (beginnt im ersten Iterationsschritt mit einem Schätzwert)
- maximale Abflugmasse (beginnt im ersten Iterationsschritt mit einem Schätzwert)

Sollten die Iterationsschritte keinen konvergierenden Verlauf erzeugen, sind die in der zweiten Datenbank vorgegebenen Anforderungen bei der gewählten Konfiguration und den restlichen vorgegebenen Werten nicht machbar.

Bei diesem nicht konvergierenden Verlauf steigt beispielsweise der Wert einer Entwurfsvariable von Iterationsschritt zu Iterationsschritt an und weist somit keine Konvergenz auf. In solchen Fällen ist die Machbarkeitsstudie des vorgegebenen Flugzeuges gescheitert.

4.4 Die Parametervariation

Die eben beschriebene Funktion der Einzelanalyse lässt sich im Rahmen einer Parametervariation nutzen. In diesem Fall wird, durch das Durchlaufen eines bestimmten Wertebereiches, eine Variation vorgegebener unabhängiger Entwurfsvariablen durchgeführt. Als Ergebnis erhält man eine Reihe von machbaren Entwürfen. Bei der Parametervariation ist die Anzahl der zu variierenden Parameter theoretisch nicht begrenzt.

Als Beispiel für mögliche Entwurfparameter für die Parametervariation sind hier nachfolgend die benötigten Eingaben für die Streckung des Flügels und die Flügelbezugsfläche beschrieben.

Hierbei handelt es sich um die Informationszeilen, die in die GUI „Programmsteuerung“ eingegeben werden müssen.

Streckung des Flügels:

LAMDAF:3 5. 12. 8.

Diese Eingabe bedeutet, dass sich die Variable LAMDAF in der dritten Datenbank befindet. Desweiteren soll die Parametervariation mit dem Startwert 5 beginnen und nach 8 Rechenpunkten bei einer Streckung von 12 abbrechen. Die zu optimierende Streckung des Flügels betrug in diesem Fall 8.82.

Flügelbezugsfläche:

FF:3 200. 300. 5.

Diese Eingabe bedeutet, dass sich die Variable FF in der dritten Datenbank befindet.

Desweiteren soll die Parametervariation mit dem Startwert 200 beginnen und nach 5 Rechenpunkten bei einer Flügelbezugsfläche von 300 abbrechen. Die zu optimierende Flügelbezugsfläche betrug in diesem Fall 218.5 m².

4.5 Die Optimierung

Optimierung bedeutet, dass auf Basis der Anforderungen und Vorgaben für eine große Anzahl von Parametern die optimalen Werte gefunden werden müssen.

Der prozedurale Ablauf der Optimierung baut, wie auch die Parametervariation, auf einer großen Anzahl von aneinander geketteten Einzelanalysen auf. Der Unterschied ist allerdings, dass die verschiedenen Variablen nicht einfach monoton einen Wertebereich durchlaufen, sondern über einen Optimierungsalgorithmus gesteuert werden. Dieser Optimierungsalgorithmus versucht, möglichst direkt, die optimale Lösung ausfindig zu machen.

Vor einer Optimierungsbetrachtung werden Variablen gewählt, mit denen das Ergebnis der *einzelnen* Einzelanalysen bewertet werden soll. Diese Variablen bilden somit die Gütefunktion der Optimierung.

Durch die Wahl unterschiedlicher Gütefunktionen für dasselbe Entwurfsproblem, kann es zu sehr unterschiedlichen Lösungen kommen. Hierbei ist zu bedenken, dass auch die Wahl unterschiedlicher Berechnungsverfahren, z.B. zur Aerodynamik oder zu den Betriebskosten, zu Abweichungen führen wird.

Desweiteren sei hier darauf hingewiesen, dass es bei der Optimierung möglich ist, die Anzahl der zu variierenden Parameter beliebig zu wählen. Allerdings sollte man bedenken, dass mit einer steigenden Anzahl an Variablen und der daraus resultierenden höheren Anzahl an Freiheitsgraden, ein Nachvollziehen der gewonnenen Ergebnisse immer komplizierter oder sogar unmöglich wird.

Wie die Optimierung der Konfiguration funktioniert, lässt sich momentan über drei verschiedene Optimierungsalgorithmen¹³ bestimmen:

- Das Verfahren von JACOB (1980)
- Das Verfahren von HADENFELD (1993)
- Das Verfahren von WANG (1990)

¹³ Das hierfür in PrADO verantwortliche Fortran-Programm lautet OPTIM und ist in dem folgenden Verzeichniss abgelegt: C:\PrADOSYSTEM\PRADO\SOURCE\LBY\ITPROG_LBY

Das Verfahren von H.G.Jacob

Das Verfahren von H.G.Jacob funktioniert über die Extremwertsuche an einer beschränkten multivariablen Funktion ohne Kenntnis ihrer Ableitungen.¹⁴

Das Verfahren von HADENFELD

Die zweite Möglichkeit ist das Programm von HADENFELD (1993) mit dem Namen: „darwin“.

Für dieses Verfahren muss man zusätzlich entscheiden, ob die Gütefunktion maximiert oder mini-miert werden soll.

Dieses Programm wurde von Mathias Hadenfeld in PrADO integriert.

Da das Funktionsprinzip dieses Programmes von Mathias Hadenfeld sehr gut in den Kommentarzeilen seines Fortran-Programms beschrieben wurde, wird diese Beschreibung hier zitiert:

Im Unterprogramm 'darwin' wird ein vorgegebener Satz von Parametern in Richtung auf ein Optimum hin variiert. Das Optimum wird durch den kleinsten Wert dargestellt, der im Hauptprogramm aus den Parameterwerten ermittelt werden kann. Der im Hauptprogramm aus den Parametern ermittelte Wert wird als Gütewert bezeichnet.

Zum Optimieren verwendet das Programm Elemente der Evolution. Die im Programm verarbeiteten Einheiten werden als 'Individuen' bezeichnet, je nach Funktion als 'Eltern' oder 'Nachkommen'. Ein Individuum besteht aus vier Anteilen:

- einem Satz von Parameterwerten
- einem Satz von Schrittweiten für die einzelnen Parameter
- einem Speicher für den Wert der Gütefunktion 'goal' mit dem Parametersatz
- einem Speicher für das 'Alter' des Individuums

Dem Unterprogramm werden beim ersten Aufruf die Startwerte der Parameter und der dazugehörige Gütewert übergeben. Mit diesen Werten werden die ersten Eltern initialisiert. Aus den Eltern ergeben sich die Nachkommen. Parameterwerte der Eltern werden auf die

Nachkommen kopiert und dabei verändert. Dabei werden normalverteilte Zufallszahlen mit dem Mittelwert 0 und der zu den Parametern gehörenden Schrittweite als Standardabweichung erzeugt und zu den Parametern addiert. Der Vorgang wird entsprechend der biologischen Evolution als 'Mutation' bezeichnet.

Die einzelnen Parameter eines Nachkommen stammen von mehreren Eltern, die mit einem Zufallsgenerator ausgewählt werden. Bei dieser 'Rekombination' kommt es zu einer

Durchmischung der Parameterwerte der Eltern, was sich vorteilhaft auf die Konvergenzgeschwindigkeit auswirkt.

Um ein Anpassung der Schrittweiten an die Gütefunktion zu ermöglichen, werden sie ebenfalls einer Mutation unterworfen. Die dafür notwendige Standardabweichung wird fest

vorgegeben.

Da die Nachkommen von Eltern mit günstigen Schrittweiten mit einer höheren Wahrscheinlichkeit überleben als die Nachkommen von Eltern mit ungünstigen Schrittweiten, setzt sich die günstigste Schrittweite automatisch nach einiger Zeit durch.

¹⁴

Weitere Informationen hierzu findet man in dem Buch von H.G.Jacob (1975):
 FORTRAN-Programm zur Ermittlung eines lokalen Optimums einer
 beschränkten multivariablen Gütefunktion ohne Kenntnis ihrer Ableitungen

Ist ein Nachkomme erzeugt, so werden seine Parameterwerte an das Hauptprogramm zurückgegeben. Dort wird der dazugehörige Gütewert berechnet, der dann beim nächsten Aufruf wieder an 'darwin' übergeben und dort gespeichert wird.

Wurde eine vorgegebene Anzahl von Nachkommen erzeugt, folgt die 'Selektion'. Dabei werden aus den Eltern und den Nachkommen die Individuen herausgesucht, die den kleinsten Gütewert haben. Sie werden zu Eltern der nächsten Generation, der Rest der Individuen wird gelöscht. Das Programm springt wieder an den Anfang der Schleife zum Erzeugen von Nachkommen.

Bei der Selektion wird das Alter der Individuen erhöht. Fällt ein Individuum über mehrere Generationen nicht bei der Selektion heraus, so wird es bei Erreichen eines vorgegebenen Alters vom Programm herausgenommen. Dabei kommt es zwar zu einer Verschlechterung des Bestwertes der Gütefunktion, es wird aber eine schnellere Anpassung der Schrittweiten erreicht.

Als Abbruchkriterium stehen zwei verschiedene Kriterien zur Verfügung. Beim ersten Kriterium wird die Konvergenzgeschwindigkeit beobachtet. Ist sie zu klein, so bricht die Routine ab. Beim zweiten Kriterium wird die Differenz der Funktionswerte der Eltern gebildet. Unterschreitet die Differenz zwischen den besten und den schlechtesten Eltern eine bestimmte Grenze, so wird abgebrochen. Beide Kriterien lassen sich in ihrer Empfindlichkeit über den Wert von 'tolrnz' einstellen.

Das Verfahren von Heping Wang

Die dritte Möglichkeit, eine Optimierung durchzuführen, ist das Optimierungsverfahren von Heping Wang.

Dieses Verfahren funktioniert über einen Zufallsgenerator und ist zurzeit bis maximal 50 Optimierungsvariablen geeignet. Auch hier muss man sich entscheiden, ob die Gütefunktion maximiert oder minimiert werden soll.¹⁵

¹⁵ Das hierfür verantwortliche Unterprogramm findet sich unter der folgenden Speicherort:
C:\PRADOSYSTEM\PRADO\SOURCE\LBYEX_LBY\exv4.for

5 Die Ergebnisse von PrADO

Hier geht es darum, welche Informationen PrADO liefern kann und wie sich diese Informationen überprüfen lassen.

5.1 Ergebnismöglichkeiten

5.1.1 Ergebnisse aus der E/P/O

Die, im vorhergehenden Kapitel 4 beschriebenen, bedeutsamsten Funktionen von PrADO können eine Aussage treffen über:

- die Umsetzbarkeit einer Konfiguration durch die Einzelanalyse
- die Parameterempfindlichkeit durch die Parametervariation
- eine vorteilhafte Konfiguration durch die Optimierung

Alle hierzu ausgegebenen Informationsdateien sind in den Dateien „GEIT.DAT“, „GPVAR.dat“ und „GOVAR“ innerhalb des Projektordners abgelegt.

Diese Dateien lassen sich sehr einfach mit Excel in optisch ansprechendere Diagramme umwandeln, da hier die variablen Werte der einzelnen Schritte spaltenweise angeordnet sind. Als Beispiel für die gleich aufgebauten Dateien ist in Bild 5.1 die Datei der Parametervariation dargestellt.

Die im Kopf der Datei von oben nach unten aufgeführten Variablen, bilden in der gleichen Reihenfolge die Spalten von links nach rechts.

#	Definition der Variablen	Nummer der Variante	Streckung/Fluegel	Bezugsflaeche/Fluegel	Betriebsleermasse (OWE)	Kraftstoffmasse/Summe/Flug mit maximaler Nutzlast	Maximale Abflugmasse	Erforderlicher Triebwerksstandsdruck (Ma=0, H=0km)	Direkte Betriebskosten (sitzkilometer)	Vektor/Randbedingungen	Vektor/Randbedingungen	Vektor/Randbedingungen	Vektor/Randbedingungen	Vektor/Randbedingungen	Vektor/Randbedingungen	Vektor/Randbedingungen	Vektor/Randbedingungen	Vektor/Randbedingungen	Vektor/Randbedingungen
1.	0.60000000E+01	0.30000000E+02	0.95500908E+04	0.37685722E+04	0.214241937E+05	0.461124453E+05	0.118953971E+00	0.728704578E+02											
2.	0.70000000E+01	0.30000000E+02	0.961618608E+04	0.296368244E+04	0.206794044E+05	0.439155555E+05	0.115995462E+00	0.143357644E+03											
3.	0.80000000E+01	0.30000000E+02	0.966698140E+04	0.243127123E+04	0.201977730E+05	0.423673851E+05	0.114105618E+00	0.143213990E+03											
4.	0.90000000E+01	0.30000000E+02	0.977358129E+04	0.206901192E+04	0.199421461E+05	0.414201657E+05	0.113050833E+00	0.143140362E+03											
5.	0.10000000E+02	0.30000000E+02	0.987854741E+04	0.180372094E+04	0.197819562E+05	0.409350723E+05	0.112369473E+00	0.143095249E+03											
6.	0.11000000E+02	0.30000000E+02	0.989541599E+04	0.160970251E+04	0.197048554E+05	0.406092678E+05	0.112991029E+00	0.142915196E+02											
7.	0.12000000E+02	0.30000000E+02	0.101235650E+05	0.145217880E+04	0.19675072E+05	0.404204470E+05	0.111778660E+00	0.143066108E+03											
8.	0.60000000E+01	0.35000000E+02	0.964726818E+04	0.307779332E+04	0.208257160E+05	0.431952602E+05	0.116513380E+00	0.142971691E+03											
9.	0.70000000E+01	0.35000000E+02	0.970790516E+04	0.246097970E+04	0.202684610E+05	0.412764763E+05	0.114312602E+00	0.142833452E+03											
10.	0.80000000E+01	0.35000000E+02	0.981154201E+04	0.205736233E+04	0.199685126E+05	0.400430798E+05	0.113098913E+00	0.142761819E+03											
11.	0.90000000E+01	0.35000000E+02	0.989394976E+04	0.175819849E+04	0.197518760E+05	0.393374967E+05	0.112227087E+00	0.142711395E+03											
12.	0.10000000E+02	0.35000000E+02	0.100303279E+05	0.156123761E+04	0.196911387E+05	0.389405674E+05	0.111896129E+00	0.142697444E+03											
13.	0.11000000E+02	0.35000000E+02	0.101598980E+05	0.139463677E+04	0.196541153E+05	0.386431397E+05	0.111655719E+00	0.142684876E+02											
14.	0.12000000E+02	0.35000000E+02	0.103028125E+05	0.126633623E+04	0.196688373E+05	0.384802051E+05	0.111599094E+00	0.142622512E+02											

Bild 5.1

GPVAR.dat

Desweiteren lassen sich die Daten, die während einer Parametervariation oder einer Optimierung nach jedem Iterationsschritt in die Datei „GPVAR.dat“ bzw. „GOVAR“ geschrieben werden, können über die GUI: „Datenkonvertierung“ in *Tecplot* Dateien (plt-Dateien) umgewandelt werden.

5.1.2 Daten zu den einzelnen Disziplinen

Mit PrADO lässt sich eine Vielzahl verschiedener Informationen aus den einzelnen Disziplinen des Flugzeugbaus berechnen. Anschließend lassen sich diese Informationen grafisch, in Form von Diagrammen, darstellen.¹⁶ In der folgenden Tabelle 5.1 sind alle in der benutzten Programmversion vorhandenen Diagrammvarianten zusammengefasst. Hierbei sind die folgenden Punkte zu beachten:

- Es lassen sich *nur* Diagramme darstellen, deren zugehörigen Module bereits durchlaufen wurden, da die „Tecplot/Diagramme-GUI“ keine Berechnungen durchführt. Tecplot ruft lediglich die bereits in dem Ordner „Plot“ vorhandenen Dateien auf.
- Nicht alle aufgeführten Möglichkeiten sind in den vorhandenen und unveränderten GUIs integriert. Sie lassen sich dort somit weder wiederfinden noch komfortabel über eine Schaltfläche ansprechen.

Tabelle 5.1 Diagrammvielfalt

Bereich	verantwortliches Unterprogramm	Beschreibung der darzustellenden Information
Geometrie	GCEO1	stellt die Sichtwinkel der Piloten dar
	GCEO2	stellt die Querschnittsverteilung des Flugzeugs als Funktion der Flugzeug-Längsachse dar
Flugmechanik	NRD1	Darstellung des Massen-Reichweiten-Diagramms
	NRD2	Darstellung des Nutzlast-Reichweite-Diagramms
	FLENV1	stellt die Flugenveloppe des Flugzeugs dar.
	PENAUD1	stellt für einen Flugzustand das PENAUD-Diagramm dar
	GKURVE1	stellt die horizontalen Kurvenleistungen des Flugzeugs dar
	GFLM1	dient zur Darstellung des Höhen-Machzahl-Diagramms
	GFLM2	Darstellung der Änderung des Maximalschubes, des erforderlichen Schubes sowie des Widerstandes über der Flugzeit

¹⁶ Die hierfür verantwortlichen Fortran-Programmtexte sind unter dem folgenden Speicherort zu finden: C:\PrADOSYSTEM\PRADO\SOURCE\LBY\2DKURVE_LBY

GFLM3	Darstellung der Änderung der Flugzeug- und Kraftstoffmasse über der Flugzeit
GFLM4	Darstellung der Änderung der Vertikal- und Horizontalbeschleunigung über der Flugzeit
GFLM5	Darstellung der Änderung der auf die maximale Höhe normierten Höhe, der auf die maximale Machzahl normierten Machzahl und der auf die maximale Reichweite normierten Reichweite
GFLM6	Darstellung der Änderung des Anstell-, Bahn- und Schubwinkels über der Flugzeit
GFLM7	Darstellung der Änderung des Auftriebsbeiwertes über der Flugzeit
GFLM8	Darstellung der Änderung des Widerstandsbeiwertes über der Flugzeit
GFLM9	Darstellung der Änderung der Gleitzahl über der Flugzeit
GFLM10	Darstellung der Änderung der Staupunkttemperatur über der Flugzeit
GFLM11	Darstellung der Änderung des spez. Kraftstoffverbrauchs über der Flugzeit
GFLM12	Darstellung der Änderung des Kraftstoffverbrauchs über der Flugzeit
GFLM13	Darstellung der Änderung des Reichweitefaktors über der Flugzeit
GFLM14	Darstellung der Änderung der spez. Reichweite über der Flugzeit
GFLM15	Darstellung des SEP-Parameters über der Flugzeit.
GFLM16	Darstellung der Übergrund- und Steiggeschwindigkeit als Funktion der Flugzeit
GFLM17	Darstellung der Flughöhe als Funktion der Reichweite
GFLM18	Darstellung der Änderung der Kräfte in Bahnrichtung als Funktion der Flugzeit
GFLM19	Darstellung der Änderung der Kräfte senkrecht zur Bahn als Funktion der Flugzeit
GFLM20	Darstellung der Schwerpunktswanderung als Funktion der Flugzeit
GFLM21	Darstellung der Flugzeugmasse als Funktion der Schwerpunktswanderung in X-Richtung
GFLM22	Darstellung der Flugzeugmasse als Funktion der Schwerpunktswanderung in Z-Richtung
GFLM25	Darstellung der Startbahnlänge als Funktion der Abflugmasse
GFLM26	Darstellung der Landebahnlänge als Funktion der Landemasse
GFLM27	stellt die erreichbare Flughöhe als Funktion der Flugzeugmasse dar
GFLM28	Darstellung des Flugzeit/Kraftstoff-Reichweite-

		Diagramms
	VN1	stellt das V-n-Diagramm nach FAR25 dar
Antrieb	GTK1	stellt den Triebwerksschub als Funktion von Flugmachzahl, -höhe, Anstellwinkel und Drosselgrad dar
	GTK2	stellt den spezifischen Kraftstoffverbrauch als Funktion von Flugmachzahl, -höhe, Anstellwinkel und Drosselgrad dar
	GTK3	stellt den zeitlichen Kraftstoffverbrauch als Funktion von Flugmachzahl, -höhe, Anstellwinkel und Drosselgrad dar
	GTK4	stellt den spezifischen Kraftstoffverbrauch als Funktion des Schubes für verschiedene Flugzustände dar
	GKTW1	stellt das STW-Kennfeld (maximale Schubwerte) dar
	GKTW2	stellt das SFC-Kennfeld dar
	GKTW3	stellt das MTW-Kennfeld (Nicken) dar
	GKTW4	stellt das SIGMA-Kennfeld (Nicken) dar
Aerodynamik	GPAERO1	stellt die aerodynamischen Eigenschaften des Flügelprofils dar (Verfahren: AEROPRO3)
	GPAERO2	stellt die aerodynamischen Eigenschaften des Höhenleitwerksprofils dar (Verfahren: AEROPRO3)
	GPAERO3	stellt die aerodynamischen Eigenschaften des Seitenleitwerksprofils dar (Verfahren: AEROPRO3)
	GFAERO1	stellt die aerodynamischen Eigenschaften des Flügels dar (Verfahren: P1PANELF)
	GFAERO2	stellt die aerodynamischen Eigenschaften des Flügels dar (Verfahren: LIFTING_LINE1)
	GHAERO1	stellt die aerodynamischen Eigenschaften des HLW dar (Verfahren: P1PANELF)
	GAERO1	stellt die Gleitzahl als Funktion der Auftriebsbeiwertes dar
	GAERO2	Nullwiderstand als Funktion der Flugmachzahl dar.
	GAERO3	stellt den induzierten Widerstand als Funktion des Anstellwinkels dar
	GAERO4	stellt die aerodynamische Polare dar
	GAERO5	stellt die Auftriebskurve dar
	GAERO6	stellt den Nullanstellwinkel als Funktion der Machzahl dar
	GAERO7	stellt die Gleitzahl als Funktion der Flugmachzahl dar
	GAERO8	stellt die maximale Gleitzahl als Funktion der Flugmachzahl und Flughöhe dar
GAERO9	stellt den Auftriebsbeiwert für die maximale Gleitzahl als Funktion der Flugmachzahl und Flughöhe dar	
GAERO10	stellt den Anstellwinkel für die maximale Gleitzahl	

		als Funktion der Flugmachzahl und Flughöhe dar
	GAERO11	stellt den Widerstandsbeiwert für die maximale Gleitzahl als Funktion der Flugmachzahl und Flughöhe dar
	GAERO12	stellt den Auftriebsbeiwert als Funktion der Flugmachzahl und des Anstellwinkels dar
	GAERO13	stellt den Widerstandsbeiwert als Funktion der Flugmachzahl und des Anstellwinkels dar
	GKAERO1	stellt das CA-Kennfeld dar
	GKAERO2	stellt das CWi-Kennfeld dar
	GKAERO3	stellt das CM-Kennfeld dar
	GKAERO4	stellt das CWO-Kennfeld ohne den Anteil aus den Profil-polaren dar
	GKAERO5	stellt den CWO-Profilpolarenanteil dar
	GKAERO6	stellt das CA-Kennfeld des Höhenleitwerks dar
	GKAERO7	stellt das CWi-Kennfeld des Höhenleitwerks dar
	GKAERO8	stellt das CM-Kennfeld des Höhenleitwerks dar
	GKAERO9	stellt dan CWO-Profilpolarenanteil des Höhenleitwerks dar
Masse und Schwerpunkt	GWEIGHT1	stellt die Massenaufteilung des leeren Flugzeugs dar
	GWEIGHT2	stellt die Massenaufteilung des Flugzeugs dar (Flug mit maximaler Abflugmasse)
	GFLM20	Darstellung der Schwerpunktwanderung als Funktion der Flugzeit
	GFLM21	Darstellung der Flugzeugmasse als Funktion der Schwerpunktwanderung in X-Richtung
	GFLM22	Darstellung der Flugzeugmasse als Funktion der Schwerpunktwanderung in Z-Richtung
Lasten	GWEIGHT3	stellt die Schnittlasten der Flügeldimensionierung dar
	GWEIGHT4	stellt die Schnittlasten der Rumpfdimensionierung dar
Betriebskosten	GDOC1	stellt die Betriebskostenaufteilung des Flugzeugs dar (DOC-Berechnungspunkt)
	GDOC2	Darstellung des DOC-Reichweiten-Diagramms
	GDOC7	Darstellung des jährliche Ausnutzung-Reichweiten-Diagramms

5.1.3 Informationen zur Lärmausbreitung

Die Genauigkeit der integrierten Lärmanalyse beträgt laut **Westphal 2004** ca. 5 EPNdB für den Start und 6 EPNdB für die Landung.

Die Lärmausbreitung der einzelnen Flugphasen lässt sich nach dem Durchlaufen des Lärm-Moduls „MD28“, über die Schaltfläche „Diagramme“ im *Main Window*, darstellen.

In Bild 5.2 ist beispielsweise der Lärmfußabdruck während des Startvorgangs mit dem Programm *Tecplot* dargestellt.

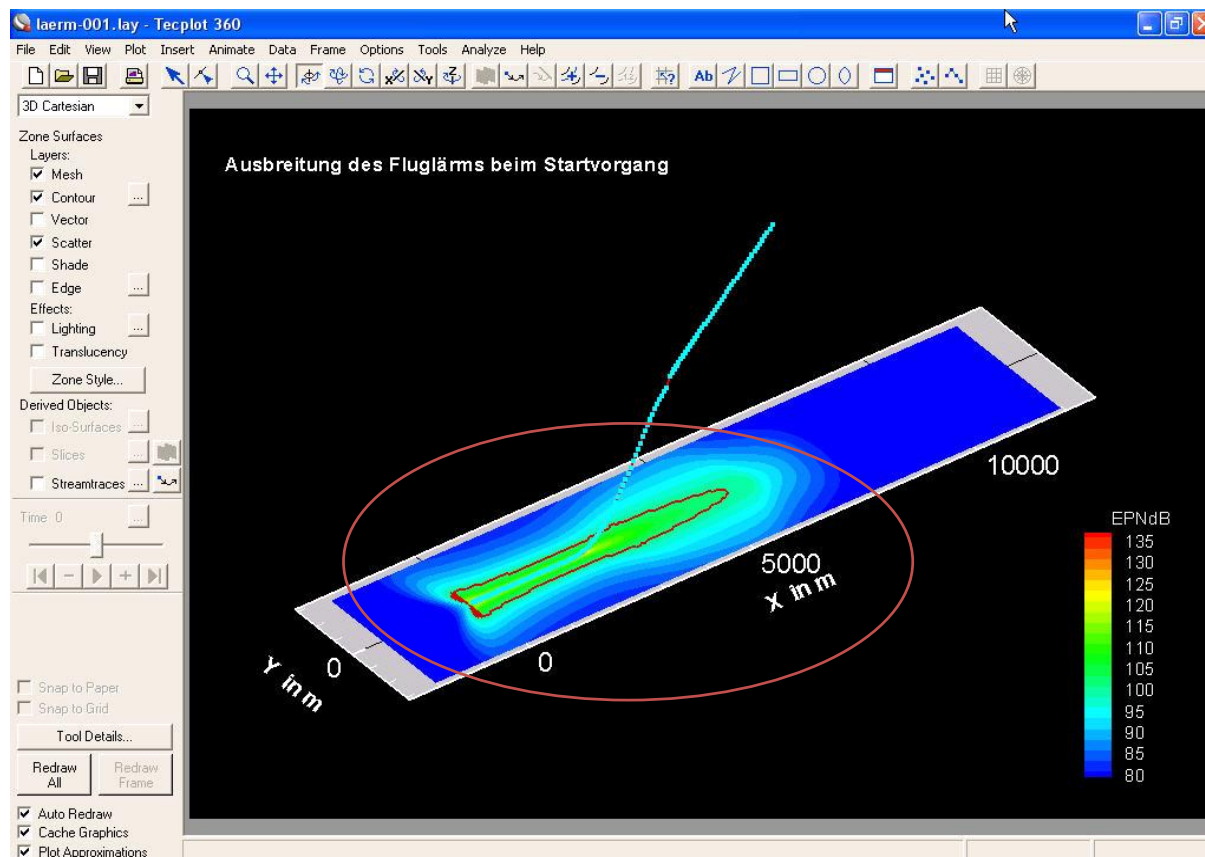


Bild 5.2 Fußabdruck

5.2 Kontrolle der gewonnenen Informationen

5.2.1 Überprüfung der eingegebenen Daten

Zur Kontrolle der in die Vorgabedatei eingegebenen Daten, lässt sich das Flugzeug sowie alle Komponenten zwei- und dreidimensional darstellen.¹⁷ Die Darstellungsmöglichkeiten wurden in der Arbeit von **Kiesel 2007** im Kapitel 6.1 näher beschrieben.

Es lassen sich hiermit auch eher sekundäre erscheinende Gegebenheiten betrachten. Beispielsweise, ob die gewählten Container, die zur Erfüllung der Transportaufgabe in

¹⁷ Die hierfür verantwortlichen Dateien sind unter dem folgenden Pfad zu finden:
C:\PrADOSYSTEM\PRADO\SOURCE\LBV\2DLFZ_LBY
und für die 3D-Darstellung: ...3DLFZ_LBY

Bezug auf das erforderliche Frachtvolumen gewählt wurden, im Rumpf überhaupt ausreichend Platz finden (siehe Bild 5.3).

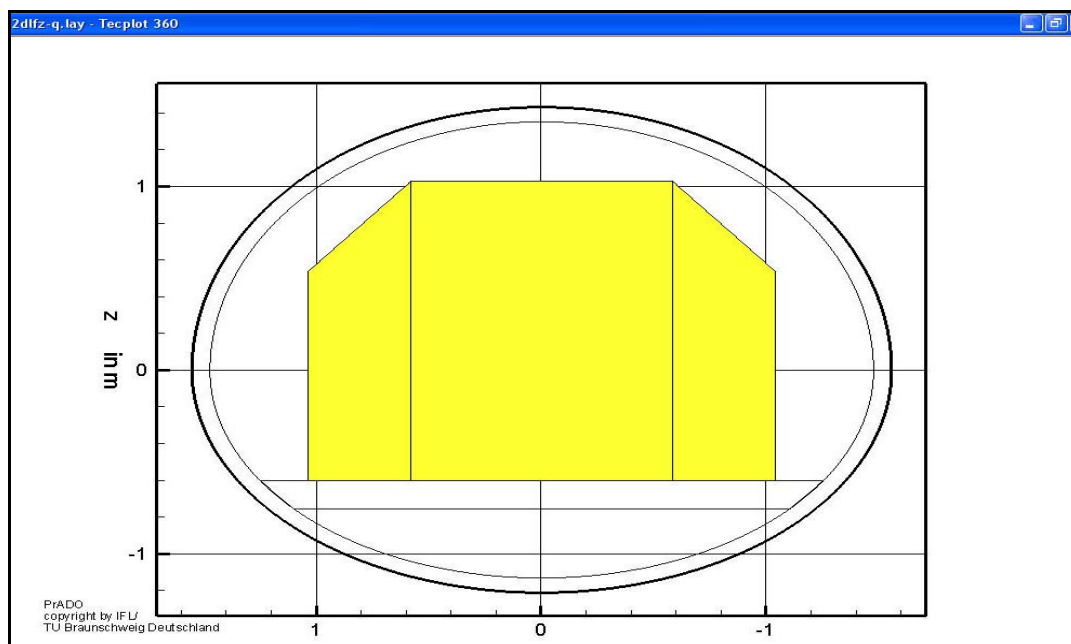


Bild 5.3 Container

5.2.2 Geometriekontrolle von Nachentwürfen

Wenn es sich bei dem zu berechnenden Flugzeug um einen Nachentwurf von bereits existierenden Flugzeugen handelt, bietet sich für die Überprüfung der erzeugten Geometrie, das Programm *Gimp* an. Mit diesem kostenlosen Grafikprogramm lassen sich die, von PrADO erzeugten, Tecplot-Darstellungen mit den originalen Geometrien überprüfen.¹⁸

Die nach dem Starten von *Gimp* durchzuführenden Schritte sind:

1. Öffnen der beiden Bilder, z.B. durch direktes „Hineinziehen“ der Bilder in die Gimp Oberfläche (siehe Bild 5.4).

¹⁸ Die aktuelle Gimp-Version und verschiedene Hilfen lassen sich unter dem folgendem Link herunterladen: <http://gimp.software-herunterladen.net>

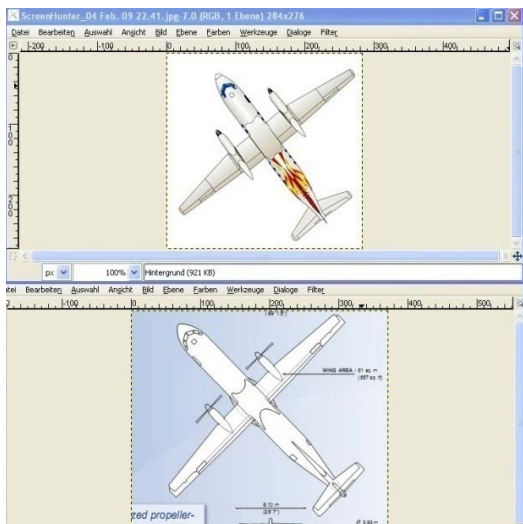


Bild 5.4 Öffnen der beiden Bilder

- Den gewünschten Bereich z.B. mit der Funktion „Rechteckige Auswahl“ auswählen (siehe Bild 5.5).

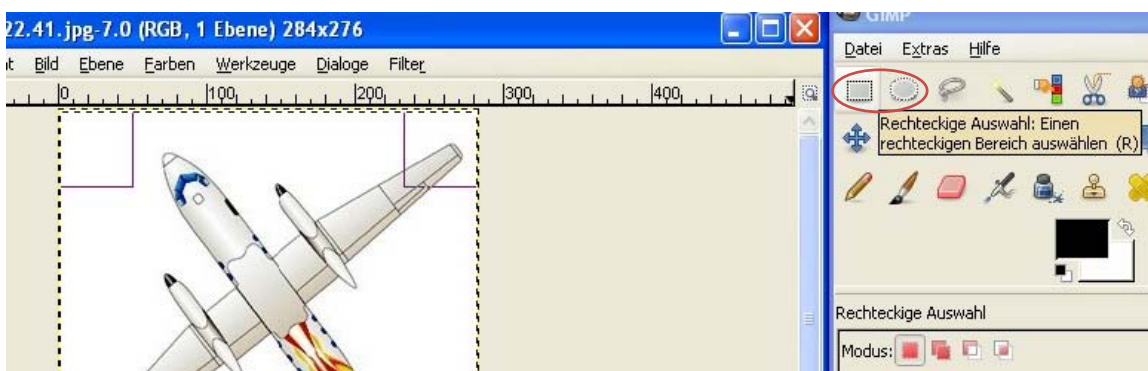


Bild 5.5 Rechteckigen Auswahl

- Anschließend wird die Auswahl über das Kontextmenü kopiert (siehe Bild 5.6).

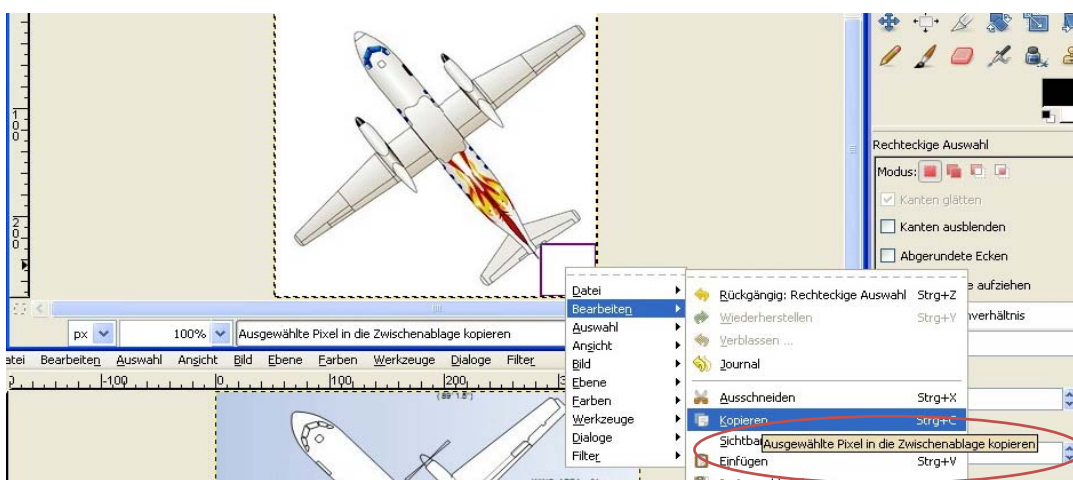


Bild 5.6 Kopieren

4. Das kopierte Bild in das andere einfügen (siehe Bild 5.7).

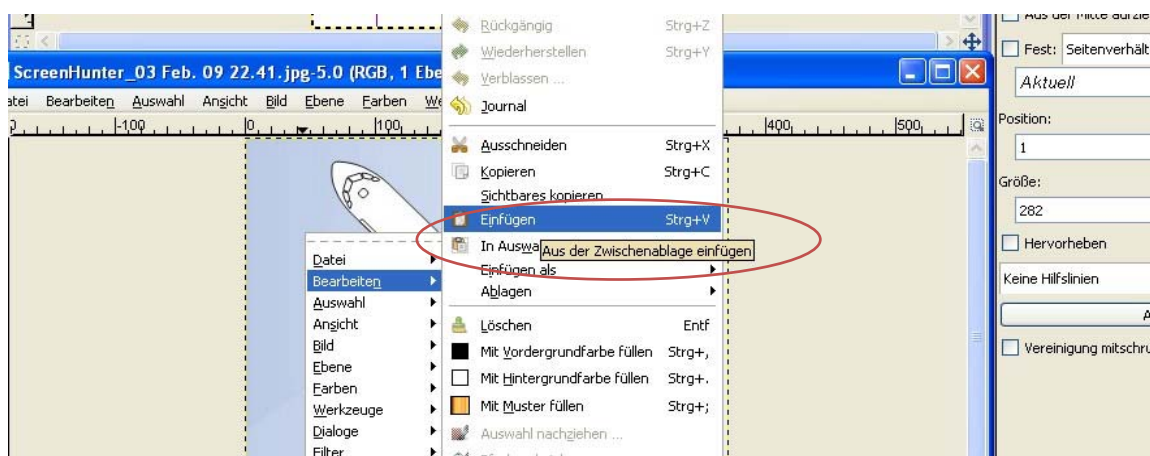


Bild 5.7 Einfügen

5. Das Fenster „Ebenen“ über einen rechten Mausklick aufrufen (siehe Bild 5.8).

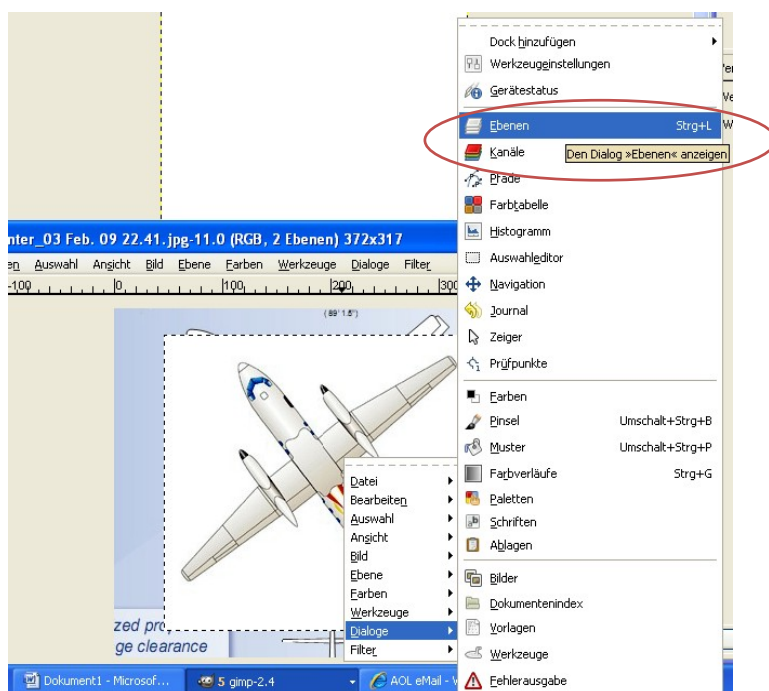


Bild 5.8 Ebenen

6. In diesem Fenster die Deckkraft herabsetzen (siehe Bild 5.9).

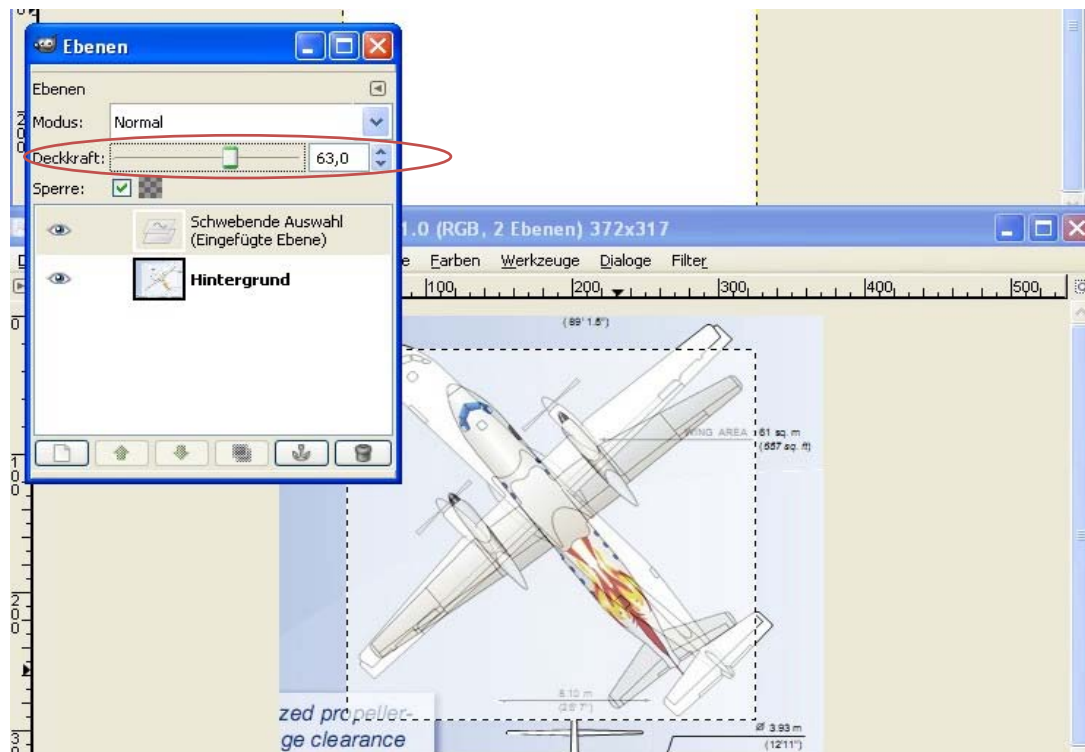


Bild 5.9 Herabsetzen der Deckkraft

7. Nun noch über die folgenden Funktionen (siehe Bild 5.10 bis 5.12) das Bild skalieren und positionieren.

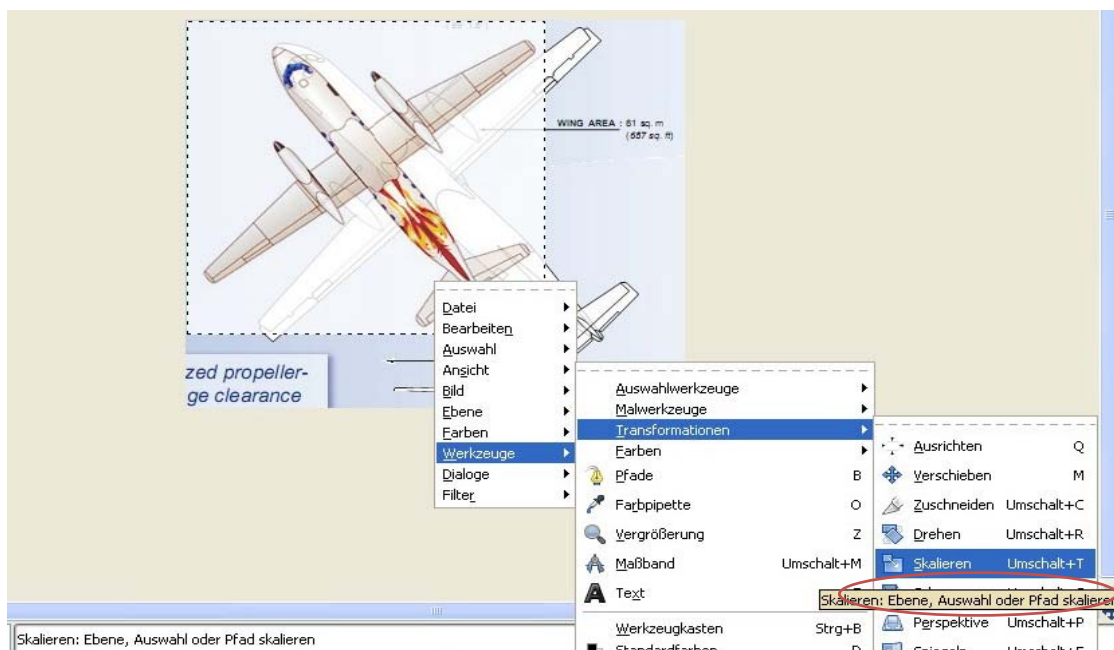


Bild 5.10 Skalieren

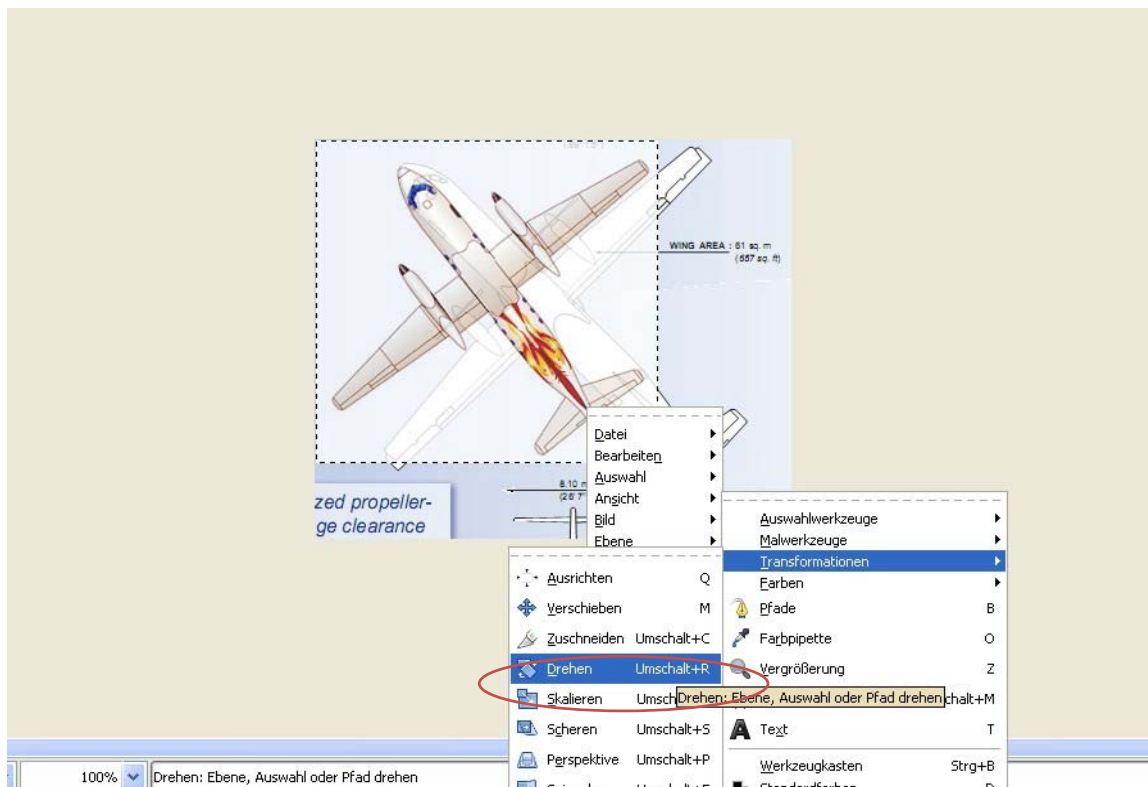


Bild 5.11 Transformation

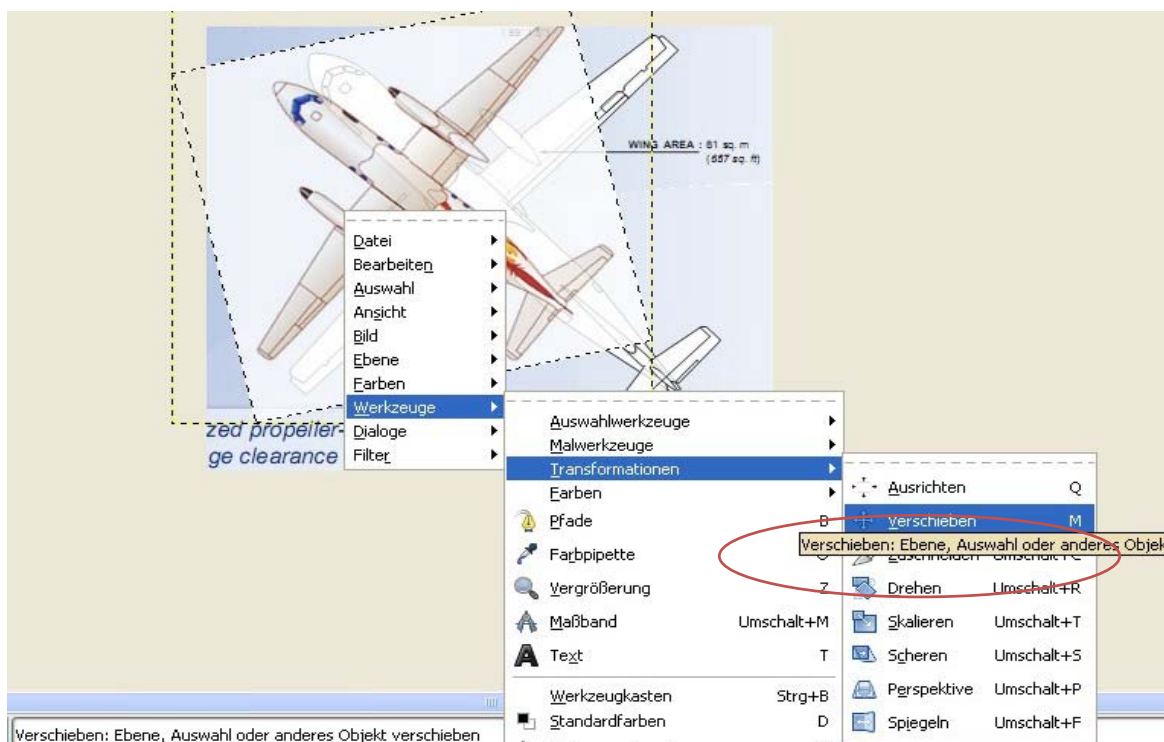


Bild 5.12 Verschieben

8. Zum Schluss lässt sich das Bild dann beispielsweise im „jpeg“-Format speichern. Dies geschieht über „Speichern unter“ und danach folgendem „Exportieren“ (siehe Bild 5.13 und 5.14).

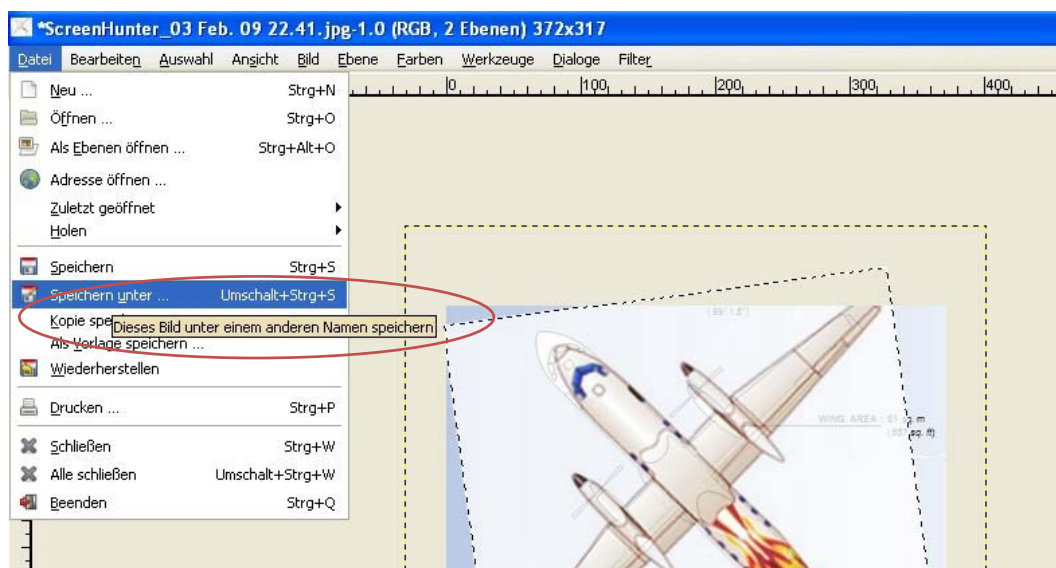


Bild 5.13 Speichern

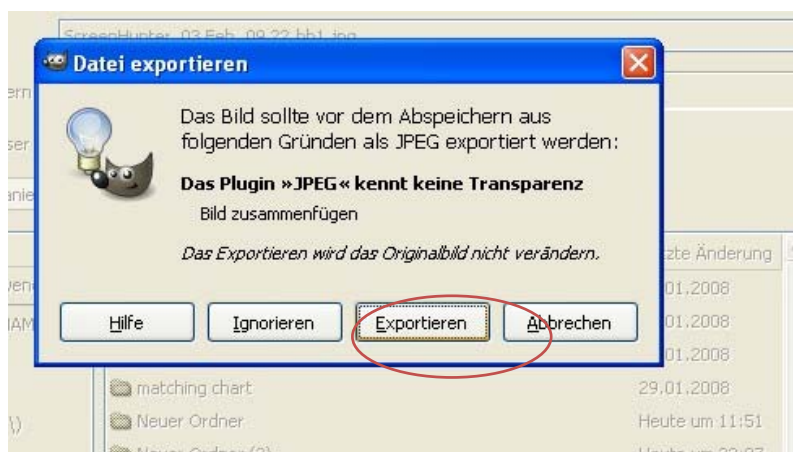


Bild 5.14 Exportieren

Beispiel an der ATR 72-200F

Im folgenden Beispiel wurde ein, von *Tecplot* ausgegebenes, SLW der ATR überprüft.¹⁹ In dem folgenden Bild 5.15 ist der türkise Umriss ein Bild von der, von *Tecplot* ausgegebenen, also aus der Vorgabedatei erzeugten Geometrie. Die roten Umriss sind aus einer maßstäblichen Dreiseitenansicht des ATR- SLWs. In diesem Bild lässt sich erkennen, dass kaum Abweichungen der gemachten Vorgaben von der realen Geometrie festzustellen sind.

¹⁹ Die Entstehung dieses SLW's wird im Kapitel XXX beschrieben.

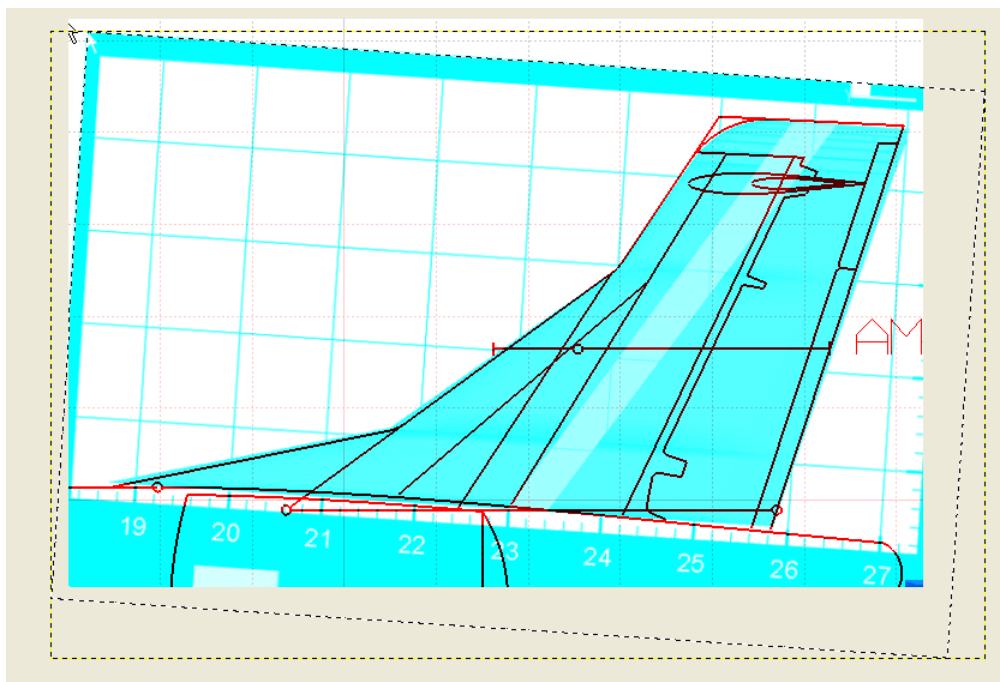


Bild 5.15 ATR- SLW

In den Vorgaben müssen, für diese Übereinstimmung, nicht nur die relativen Maßangaben und Positionen der verschiedenen Schnitte fehlerfrei eingetragen sein, sondern z.B. auch die Streckung. Solche Werte müssen ggf. noch angepasst werden, um das richtige Ergebnis zu erzielen. Weitere Informationen zum Seitenleitwerk sind im Kapitel 7 beschrieben.

5.2.3 Fehler während der E/P/O

In diesem Abschnitt geht es um die Frage, ob während der E/P/O Fehler aufgetreten sind. Hierzu lässt sich der Verlauf auf Unregelmäßigkeiten untersuchen. Desweiteren kann man überprüfen, ob unterschiedlichen Optima auftreten.

Unregelmäßigkeiten im Verlauf:

Um zu kontrollieren, ob der Wertverlauf Unregelmäßigkeiten aufweist, lassen sich die einzelnen ablaufenden Schritte, in einem Diagramm darstellen. Diese Möglichkeit lässt sich über das *Main Window* auf der linken Seite unter „Darstellung/Tecplot“, mit den Schaltflächen „Historie/...“ für die E/P/O aufrufen.

Im folgenden Beispiel (Bild 5.16) lässt sich der konvergierende Verlauf bei der Optimierung des Anstellwinkels verfolgen.

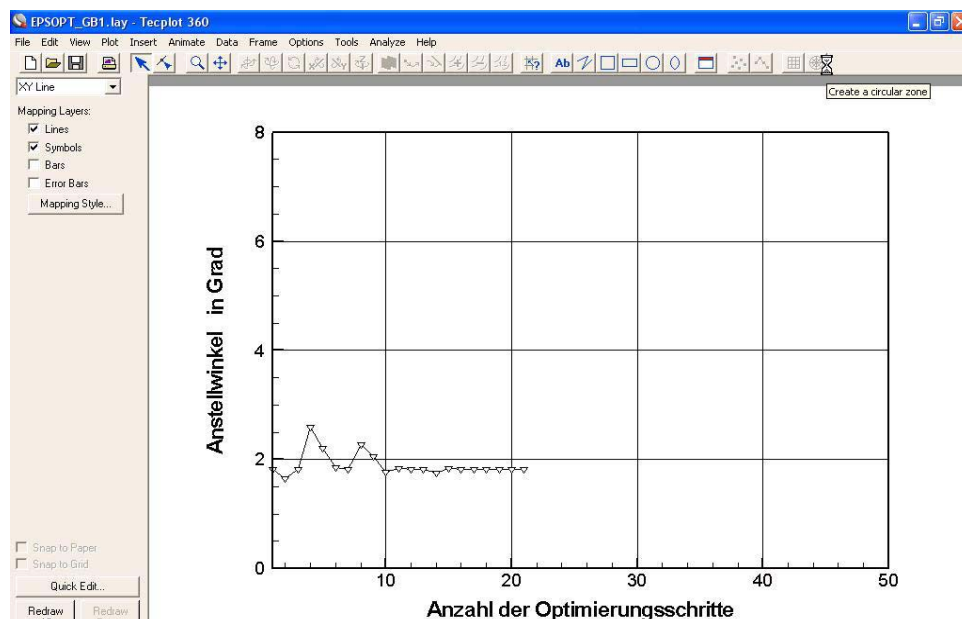


Bild 5.16 Optimierungsschritte

Desweiteren lassen sich, *auch während des Programmlaufes der Optimierung*, die Zwischenstände aller gewählten Variablen und Randbedingungen im *Main Window* über die Schaltfläche „db1 Programmsteuerung“ unter der zweiten Seite des Optimierungsbereiches, abrufen (siehe Bild 5.17). Diese Werte stammen aus der „GOVAR.DAT“-Datei im Projektordner und lassen sich bei Bedarf auch dort verfolgen.

Optimierung [2 / 2]:

Ergebnisse der Optimierung

Gütwert: 0.03890397795

Randbedingungsparameter (≤ 0 Randbeding erfüllt)

RB 1 Startbahnlänge	0.06813757543
RB 2 Landebahnlänge	-0.04252061775
RB 3 Tankvolumen	-0.2316542378
RB 4 Standschub	0.01223855312
RB 5 Flugzeuglänge	-0.6668232923
RB 6 Flugzeubreite	-0.6607169895
RB 7 Flugzeughöhe	-0.6882238282
RB 8 Landeanflugeschwindigkeit	0.01861171173
RB 9 Frachtraumvolumen	-0.2196761304
RB 10 max. Flughöhe	-0.03267189399
RB 11 LCN-Wert (starre Bahn)	-0.5911668702
RB 12 LCN-Wert (elastische Bahn)	-0.5504800296

Bild 5.17 Randbedingungen

Unterschiedliche Optima:

Auch wenn das Auftreten von unterschiedlichen optimalen Werten während des Projektes nicht vorgekommen sind und somit ein solches Phänomen eher unwahrscheinlich ist, lässt sich durch die Kontrolle mehr Sicherheit für die Richtigkeit des Ergebnisses gewinnen.

Um also zu prüfen, ob eine fehlerfreie Konvergenz entsteht, kann man das Programm mit verschiedenen Startwerten beginnen lassen und anschließend kontrollieren, ob sich trotzdem dasselbe Endergebnis einstellt (siehe Bild 5.18).

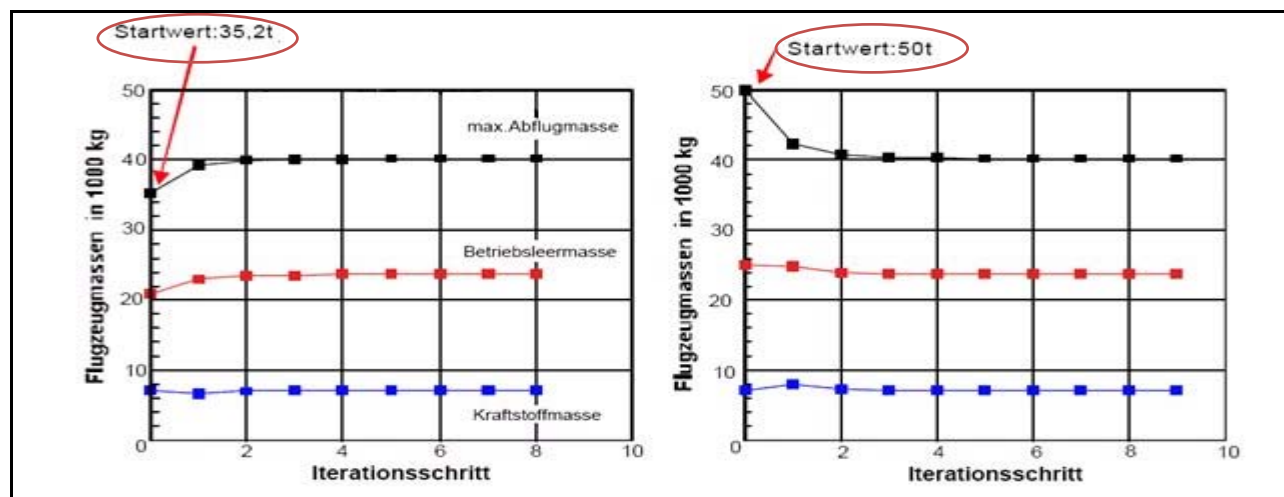


Bild 5.18 Startwert (nach HEINZE 2008)

5.3 Bewertung der Genauigkeit

Für Flugzeuge, die stark dem konventionellen Design entsprechen, dürfte die absolute Genauigkeit der Vorhersagen keine wesentlichen Abweichungen aufweisen, da viele in PrADO verwendete Formeln, durch ihre statistischen Wurzel, sehr genaue Informationen liefern.

Bei ungewöhnlichen Konfigurationen müssen die Verfahren möglichst kritisch analysiert werden. Desweiteren ist darauf hinzuweisen, dass mit den einfacheren Verfahren teilweise starke Abweichungen entstehen können.

6 Das Arbeiten mit PrADO

Der hilfreiche Charakter mancher hier aufgeführter Ratschläge wird ggf. erst während des Arbeitens mit PrADO erkennbar. Desweiteren werden einige der hier beschriebenen Hinweise für manche Leser als selbstverständlich angesehen werden.

6.1 Der Einstieg

Zum Beginn des Arbeiten mit PrADO bietet sich ein ausprobieren aller Programm-möglichkeiten mit einem möglichst einfachen aber dennoch lauffähigen Projektflugzeug an.

Es ist wichtig, dass das Projektflugzeug möglichst nicht zu komplex ist, um die Rechenzeit gering zu halten. Zum Verdeutlichen der Wichtigkeit sei hier auf eine Rechenzeit von bis zu 6 Stunden für eine Einzelanalyse verwiesen.

Folgende Dinge können unternommen werden, um eine vorhandene Vorgabedatei, von der bekannt ist, dass sie ohne Fehler durchläuft, schneller zu machen:

- Das Iterationskriterium auf ein hohen Wert setzen (z.B.: 0,01), da es für die ersten Tests nicht auf eine hohe Genauigkeit ankommt und die Anzahl der notwendigen Iterationen so sehr stark abnimmt.
- Den Spielraum der Iterationsvariablen möglichst klein wählen.
- Alle Elemente, auf die verzichtet werden kann, sollten ausgeschaltet werden (z.B. Winglets).
- Wahl einfacher Verfahren für die Berechnungen. Diese Verfahren benötigen wesentlich weniger Eingabewerte und basieren auf grundlegenden Parametern wie der anvisierten Abflugmasse.

Hier sind noch ein paar allgemeine Hinweise aufgezählt:

- Ein Verzeichnis, in dem man überprüfen kann, was die Variablen bedeuten, kann sehr hilfreich sein. Falls man keine „DMS Variablen Liste“ zur Hand hat, kann man sich eine Liste in der „DMS“-GUI erzeugen lassen.
- Die Tasten „Page up“ und „Page down“ sind in den oft langen Textdateien sehr hilfreich.
- Wenn man gerade eine rechnerleistungsintensive E/P/O durchführt, aber zwischendurch etwas anderes starten möchte, dann läßt sich der Programmablauf mit der Taste „Pause“ oder „Break“ gefahrlos unterbrechen.
- Die Möglichkeit, am Ende der Einzelanalyse, der Parametervariation oder der Optimierung die CPU-Zeit auszugeben zu lassen kann hilfreich sein (siehe Bild 6.1).

Hierdurch lassen sich zeitaufwendigen Modulen ausfindig machen, um diese dann ggf. auszuschalten oder zu verändern.

```

*** Rechenzeitstatistik ***
Programm          CPU-Zeit
-----
PrADO-It          2522.34
MD1                3.90
MD2               549.56
MD3               120.29
MD4                11.60
MD5               13.09
MD6               41.75
MD7                5.00
MD11              5.28
MD12              520.61
MD14              5.07
MD15              4.41
MD16              38.36
MD17              68.98
MD18              37.63
MD19              268.79
MD20              90.65
MD21              235.85
MD22              10.94
MD24              5.08
MD25              5.31
MD26              67.82
MD27              227.79
MD29              91.41
MD30              4.48
MDC                0.25
  
```

Bild 6.1 CPU-Zeit

- Es ist darauf zu achten, möglichst viele Integer zu benutzen, um die Rechnerzeit kurz zu halten.
- In manchen GUI-Oberflächen kann es sinnvoll sein, die Ansicht von „Status-Monitor“ auf „Ausgabe-Monitor“ zu ändern, um zu sehen, welche einzelnen Schritte das Programm gerade ausführt oder um zu erfahren, dass Probleme aufgetreten sind.

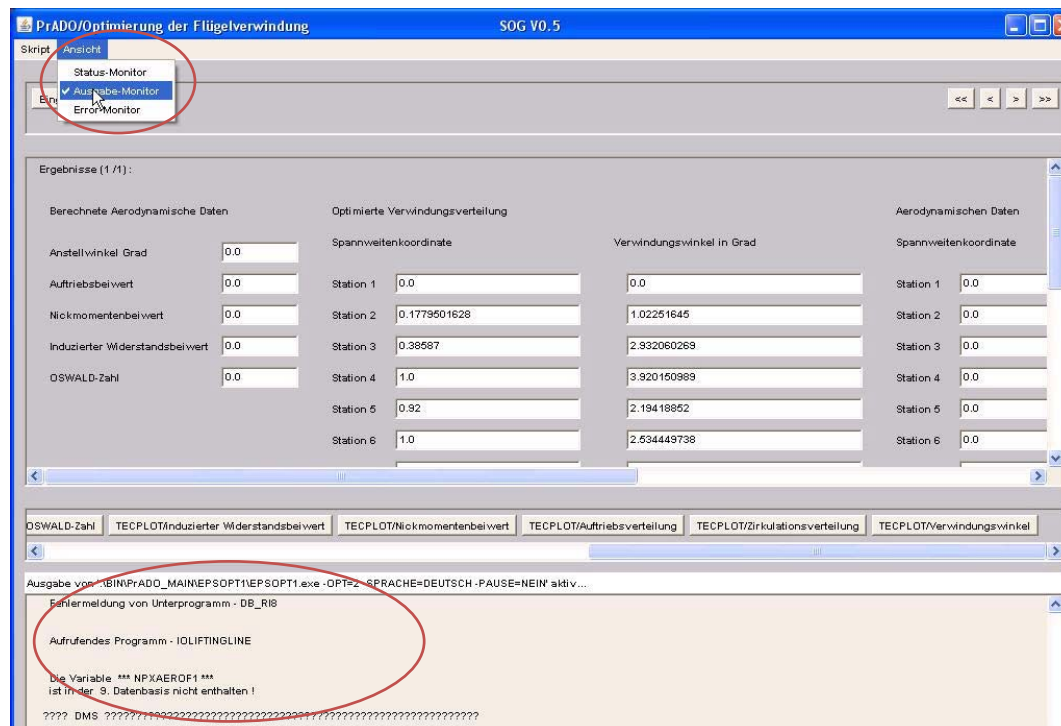


Bild 6.2 Ausgabe Monitor

- Nach einer abgebrochenen E/P/O kann es sein, dass die Datenbasis nicht mehr zu gebrauchen ist und sich auch die einzelnen Module nicht mehr ausführen lassen.

- Es ist möglich mehrere Instanzen von *PrADO* gleichzeitig laufen zu lassen.
- Markieren langer Texte im Programmlauf:
 1. Den Programmlauf mit der Taste „Pause“ unterbrechen.
 2. Das Kontextmenü, über die rechte Maustaste, innerhalb des Programmfensters aufrufen und hier drin „alles markieren“ auswählen.
 3. Anschließend *keine* Kopierfunktion beispielsweise durch die Tastenkombination „Strg+C“ aufrufen.
 4. Als Nächstes *Wordpad*, den *Editor* oder *Word* öffnen und dort im Kontextmenü durch Einfügen oder durch die Tastenkombination „Strg+V“ den Inhalt importieren.
- Wie lässt sich das Programmfenster anpassen?
 1. Aufrufen des Kontextmenüs, durch einen „Klick“ mit der rechte Maustaste auf den Rahmen (siehe Bild 6.3).

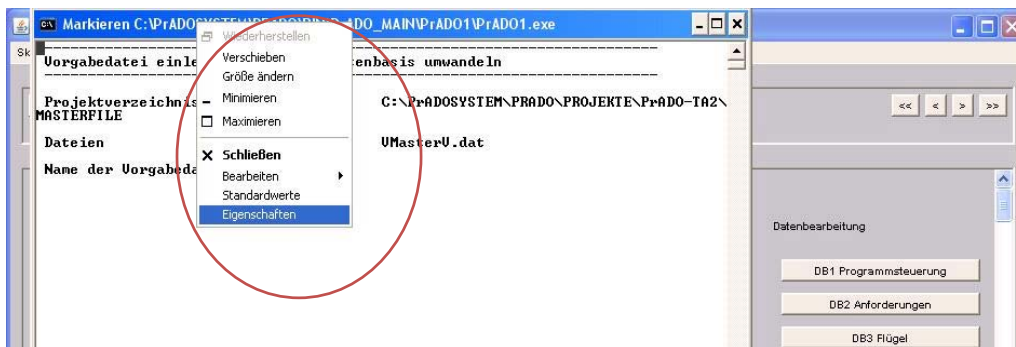


Bild 6.3 Programmfenster

2. Durch das Aufrufen des Fensters „Eigenschaften“ lassen sich dann die Farben, die Schrift und die Fenstergröße an die eigenen Bedürfnisse anpassen.(siehe Bild 6.3 und 6.4)

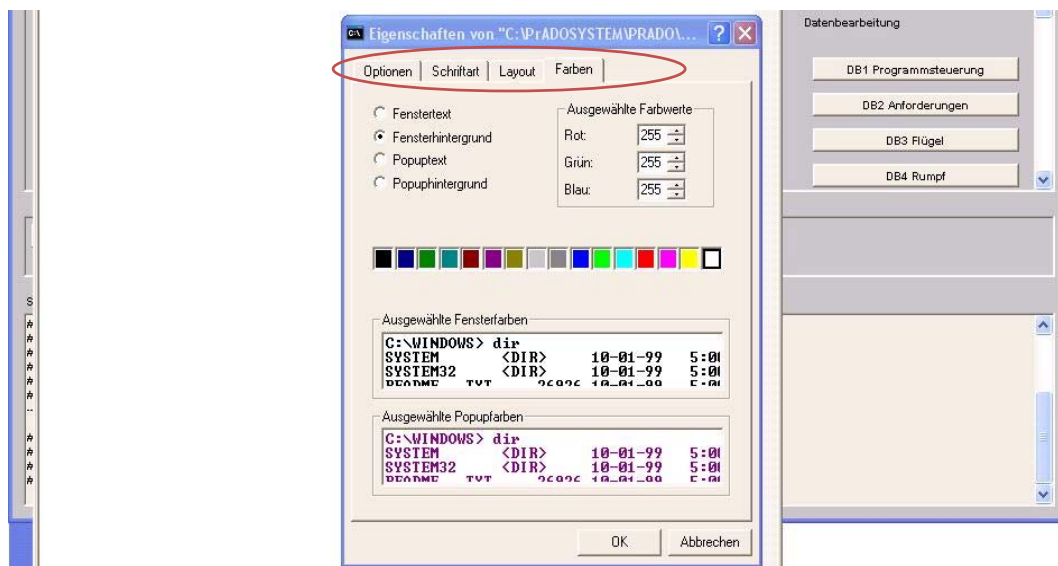


Bild 6.4 Eigenschaften

6.2 Die Handhabung der Vorgabedatei

6.2.1 Allgemeines:

Da der Aufbau der Vorgabedatei ausführlich in der Projektarbeit von Thorsten Kiesel geschildert ist, soll hier nur das Wichtigste kurz wiederholt und einige ergänzende Ratschläge gegeben werden:

- Die Datenbasis wird in der Vorgabedatei durch die Marken „**ANFANG-Dbi“ und „**ENDE-Dbi“ eingeschlossen.
- Der Eintrag der Marken muss immer in der ersten Spalte beginnen.
- Datensätze beginnen immer mit der Marke "<-", die ebenfalls in der ersten Spalte beginnt.
- Elemente die vor dem <_... z.B ein * haben sind funktionslos.
- Bei den komplizierteren Eingaben bietet es sich an, unter den Eingaben eine Kommentarzeile einzufügen, die einem die Position und somit auch die Funktion des einzugebenden Wertes verrät. z.B. relative Lage, Drehpunkt, usw.. Eine solche Kommentarzeile könnte folgendermaßen aussehen:

```
C.....1.....2.....3.....X2.....3....X2.....3
```

Zusätzlich ist eine Markierung z. B. ein "X", bei sich wiederholenden Eingaben innerhalb einer Zeile hilfreich.

- Mit dem folgenden Makro, für das Programm Excel, lässt sich die Vorgabedatei so verändern, dass man die Variablen schnell kontrollieren bzw. abzuändern kann. Hierfür blendet das Makro die Kommentarzeilen aus. So lassen sich offensichtliche Fehleingaben leicht finden und beheben. Dies ist natürlich nur sinnvoll, wenn kein Rechner mit einer PrADO-Installation zur Verfügung steht.

```

Makro von Sebastian Herda
Sub ausblenden()
For i = 2 To 10000
If Cells(i, 1) Like "C*" Then
    Rows(i).Select
    Selection.EntireRow.Hidden = True
End If
Next i
End Sub
Sub Spaltenbreite()

```

6.2.2 Sam und LLIE

Was ist der Unterschied zwischen den Vorgabedateien mit dem Anhang „SAM“ oder „LLIE“?

- **Keine Anhang** (z. B: V_A310-300.dat)
Hierbei handelt es sich um Vorgabedateien die für die Berechnungen die einfachen Verfahren verwenden.
- **LLIE** (z.B.: V_A310-300_LLIE.dat)
In diesen Vorgabedateien sind alle für SAM notwendigen Eingaben aus- und alle für Liftingline notwendigen Eingaben eingeschaltet.
- **SAM** (z. B: V_A310-300_SAM.dat)
In diesen Vorgabedateien sind alle für Liftingline notwendigen Eingaben aus- und alle für SAM notwendigen Eingaben eingeschaltet.

Je nachdem welches Verfahren verwendet wird, entstehen in den Ergebnissen Abweichungen. Zum Beispiel verändert sich die max. Flächenbelastung.

6.2.3 Auswahl der gewünschten Vorgabedatei

In dem Programmfenster muss der Name der gewünschten Vorgabedatei nicht abgetippt werden, sondern kann folgendermaßen mit der Maus ausgewählt werden:

1. Die Funktion „Markieren“ im Kontextmenü aufrufen.

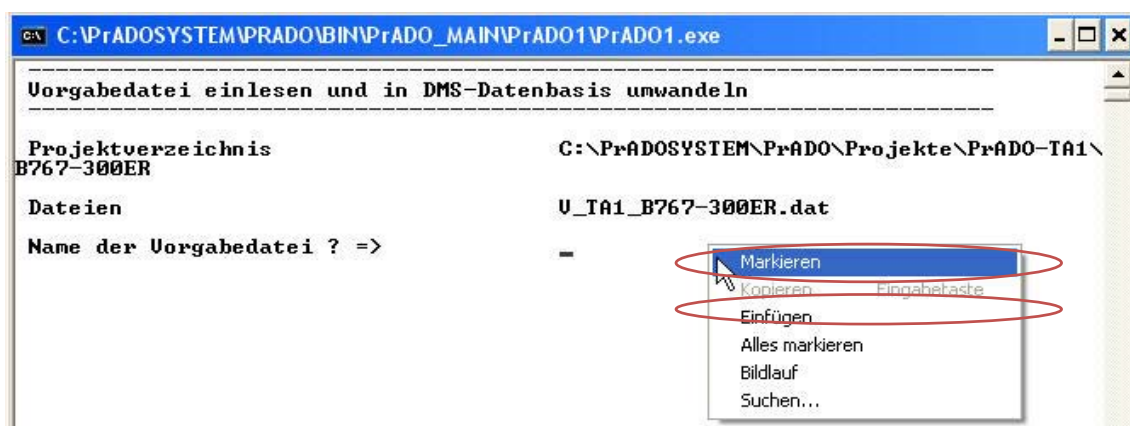


Bild 6.5 Kontextmenü

2. Jetzt durch einen „Doppelklick“ mit der linken Maustaste die gewünschte Vorgabedatei markieren (siehe Bild 6.5).
3. Anschließend durch die Funktion „Einfügen“ (siehe Bild 6.5) den Namen eintragen.

Nach dem „Einfügen“ wird das Einlesen der Vorgabedatei, durch das Bestätigen der Eingabe mit der Return-Taste, abgeschlossen.

6.3 Fehlerquellen

6.3.1 In der Vorgabedatei

Hier folgen die wichtigsten Punkte, die in der Vorgabedatei beachtet werden müssen:

- keine ä ,ü oder ö verwenden
 - keine Kommata verwenden
 - falls es sich bei den Eingaben um „Character“, also beliebige Zeichen, handelt, muss jedes Element in einer neuen Zeile stehen und kann nicht einfach durch Leerzeichen getrennt werden
 - alle „Character“-Einträge *müssen* in der ersten Spalte beginnen
 - Befehle, die funktionieren sollen, dürfen nicht durch ein vorgesetztes Zeichen vor das „<-“ z. B.: *<- deaktiviert sein
 - die Steuerzeile, die aus einer vierstelligen Zahlenkombination besteht, kann auch eine Fehlerquelle darstellen, die man leicht übersieht
- Die selten vorkommenden Möglichkeiten sind *grau* gedruckt:

Tabelle 6.1 Steuerzeile

0 (keine Optimierung)	1 (Character, z.B.: a, b...)	1 (Skalar oder Vektor)	N (Anzahl der einzugebenden Elemente)
1 (Optimierungsgröße)	2 (Integer, z. B.: 12, ...)	2 (Matrix)	-
-	3 (Real, z. B.: 1.23)	-	-

- die Zahlen der Steuerzeile sind mindestens durch ein Leerzeichen voneinander getrennt.

6.3.2 In Tecplot

Wenn sich *Tecplot* bei dem Versuch ein Diagramm darzustellen öffnet, aber kein Graph zu sehen ist und auch keine Fehlermeldung erscheint, kann es an der falschen Ansicht im Grafikprogramm liegen. Ein Ändern der Ansicht von beispielweise „XY Linie“ auf „2D Cartesian“ kann dieses Problem lösen (siehe Bild 6.6).

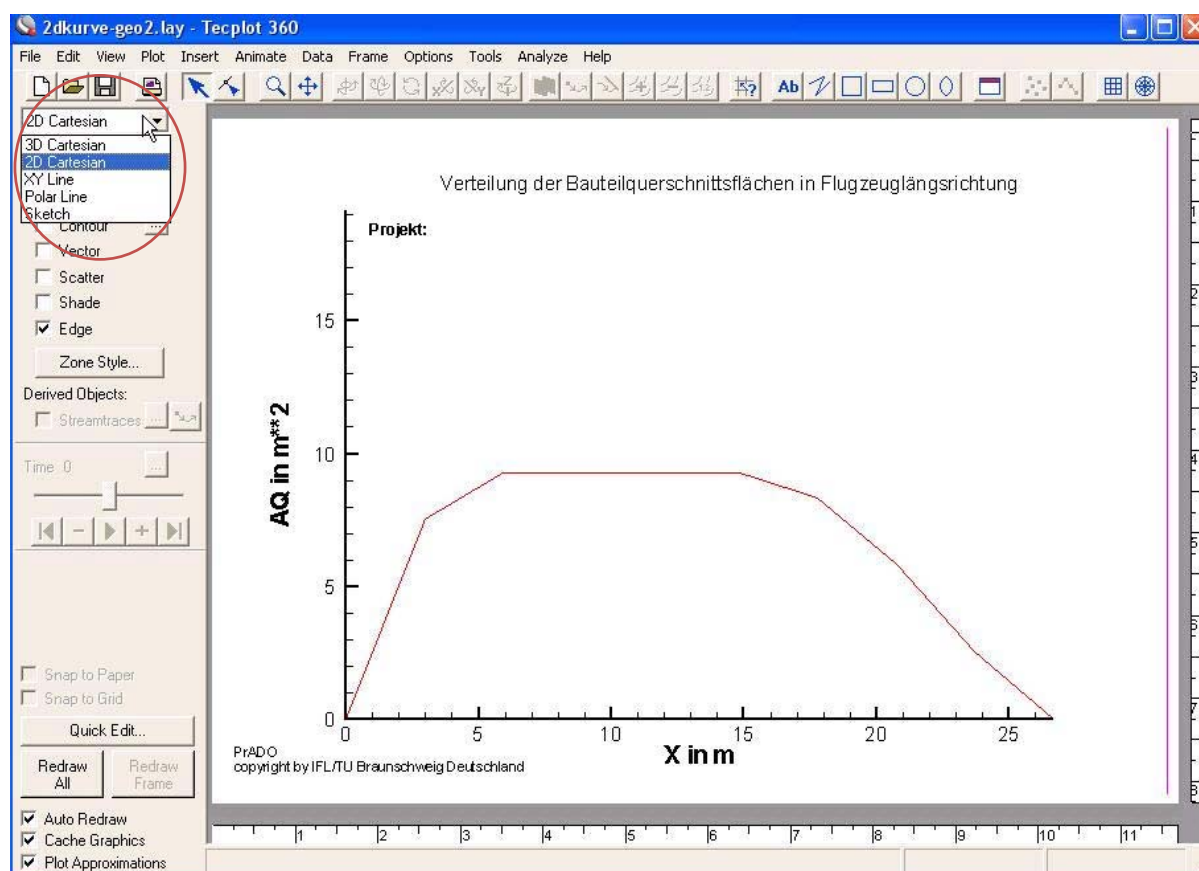


Bild 6.6 Tecplot

6.3.3 Fehlermeldungen während des Programmlaufs

Wenn der Programmlauf beispielsweise bei einer Einzelanalyse abbricht, und man keine Möglichkeit hat, den Fehler oder die Stelle, an der das Programm abbricht, zu erkennen, da sich das Programmfenster sofort nach dem Abbruch schließt, bieten sich folgende Lösungen an:

- sobald man eine Fehlermeldung im Programmlauf entdeckt, sollte man den weiteren Programmlauf mit der Taste Pause bzw. Break schnellst möglich unterbrechen
- das Unterprogramm²⁰, in der das Programm abbricht, mit dem Fortran Compiler öffnen und dort ausführen. In diesem Fall bleibt das Programmfenster auch nach dem Abbruch geöffnet und ermöglicht so die Suche nach dem Fehler.

Bei der Suche nach fehlenden Variablen muss man beachten, dass in der durch *PrADO* aus der Vorgabedatei erzeugten Datenbasis-Datei Variablen enthalten sind, die die Vorgabedatei nicht enthält und die erst durch *PrADO* erzeugt werden. Aus diesem Grund lassen sich manche Variablen nicht in der Vorgabedatei finden. Zum Beispiel ist die Variable „FREF“ noch nicht nach dem Einlesen der Vorgabedatei in den einzelnen „db“-Dateien vorhanden, sondern wird erst nach dem Durchlaufen der Module erzeugt.

Unterprogramme, die im Programmlauf Fehler verursachen, lassen sich am einfachsten mit der in Windows integrierten Suchfunktion ausfindig machen (siehe Bild 6.7). Anschließend lassen sich die FORTRAN-Programme z.B. mit dem Editor oder dem Compiler öffnen.

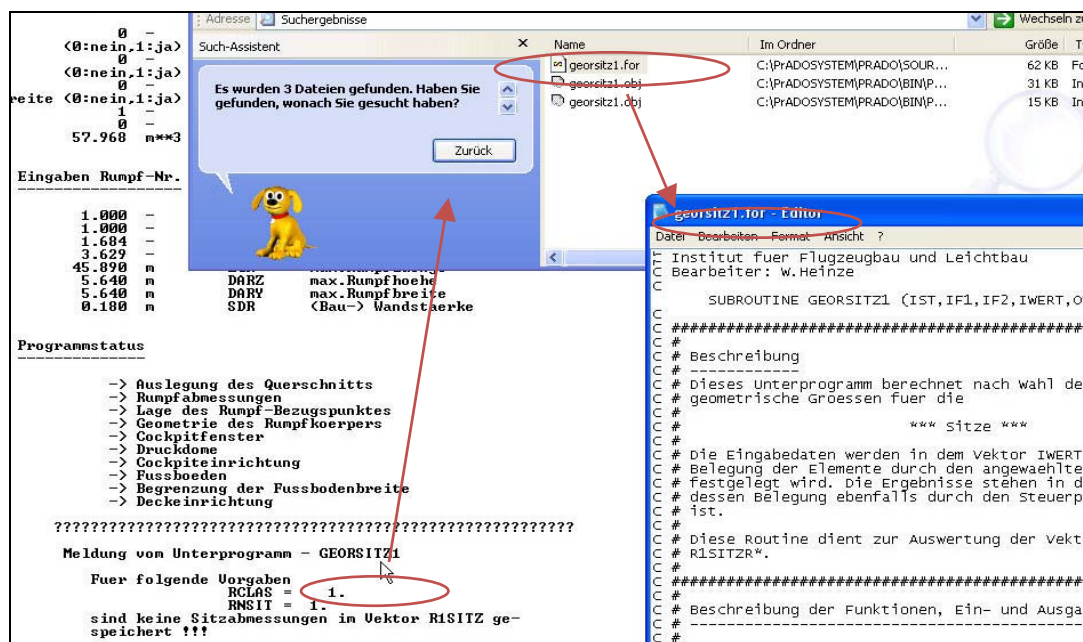


Bild 6.7 Unterprogramme

²⁰

Die ausführbaren Dateien, also die Exe-Dateien, befinden sich in dem Ordner: C:\PrADOSYSTEM\PRADO\BIN\PrADO_MAIN. In diesem Ordner finden sich die ausführbaren Dateien aller Module wieder.

6.4 Hilfreiche Tools für das Arbeiten mit PrADO

Neben dem Programm *Gimp*, das im Kapitel 5.2.1 zur Überprüfung der Geometrie benutzt wird, lässt sich die Arbeit mit PrADO durch die Nutzung der im folgenden Abschnitt vorgestellten kleinen Tools, die sich während des Arbeitens mit PrADO als sinnvoll und nützlich herausgestellt haben, vereinfachen. Bei allen folgenden Programmen handelt es sich um Freeware, die im Internet auf verschiedenen Plattformen zum Download zur Verfügung steht.

6.4.1 Ermittlung der relativen Geometriedaten

Dadurch, dass man von den Flugzeugherstellern oftmals die Flugzeuggeometrie als „dwg“-Datei bekommt (z.B.: Boeing, Airbus, ...), bietet es sich an, diese Dateien auch digital weiter zu bearbeiten.

Der eben erwähnte Dateityp lässt sich auch mit Catia V5 in der Arbeitsumgebung „Drafting“ öffnen. Diese Arbeitsumgebung befindet sich unter „Start“ und „Mechanische Konstruktionen“.

Hier soll mit Catia V5 beschrieben werden, wie man die Geometrie z.B. auf die Länge 1 m (oder jede andere beliebige Einheit) skalieren kann, wodurch man dann in der Lage ist, alle, für die Vorgabedatei benötigten, relativen geometrischen Größen einfach heraus zu messen.

Diese Informationen sind auch für Benutzer die noch keine Erfahrung mit Catia V5 haben sinnvoll. Nach etwas Übung lassen sich mit den folgenden Schritten alle notwendigen *normierten* Maße präzise gewinnen, ohne die Gefahr der Abweichungen durch falsches Messen oder das manuelle Umrechnen in den normierten Wert zu haben. Die folgenden Schritte müssen durchgeführt werden:

1. Öffnen der Arbeitsumgebung (siehe Bild 6.8).

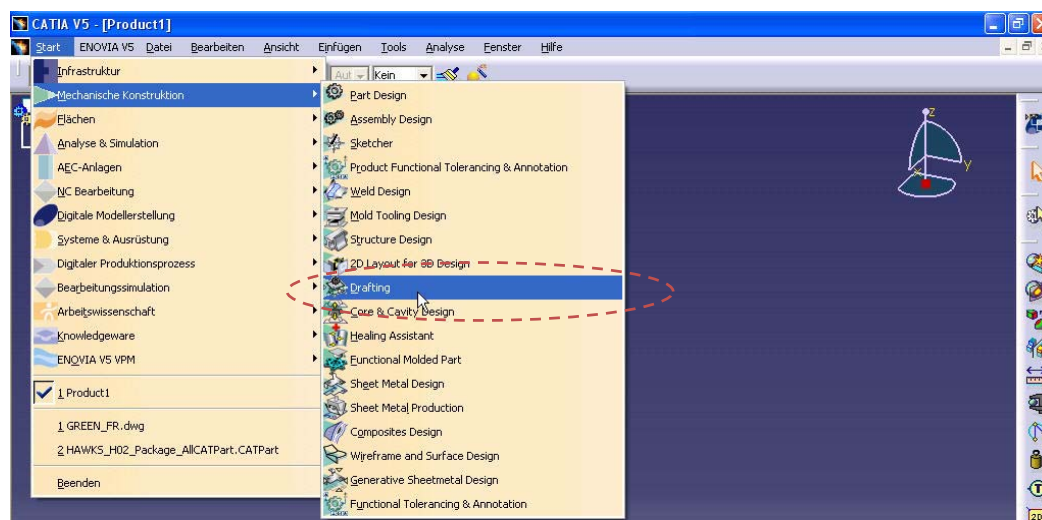


Bild 6.8 Arbeitsumgebung

2. Öffnen der „dwg“-Datei (siehe Bild 6.7).

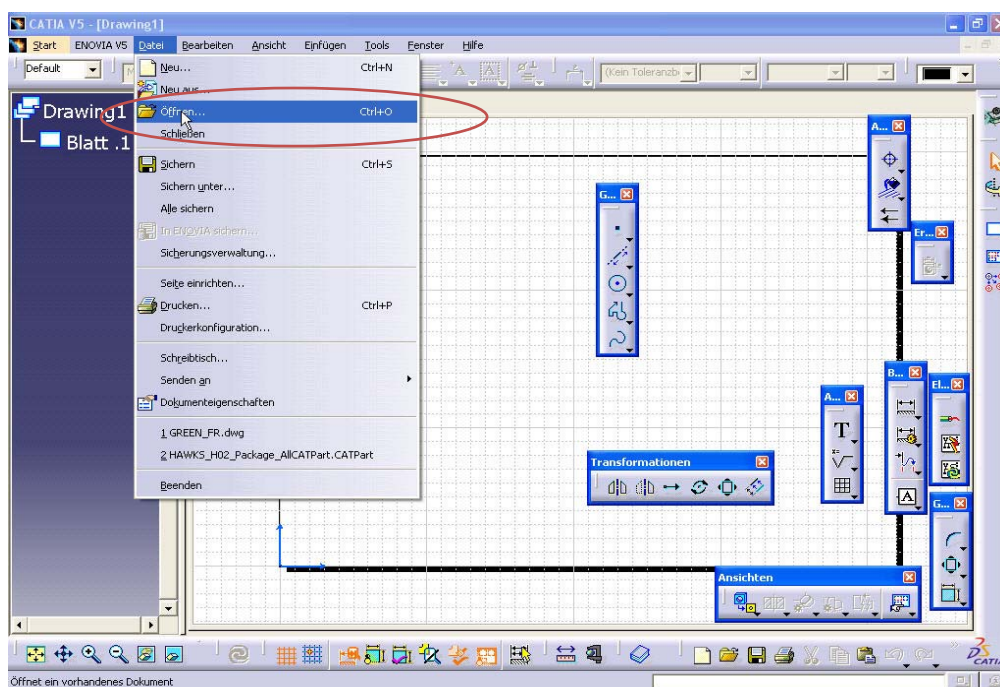


Bild 6.9 Öffnen einer Datei in Catia

3. Die Funktion „Unendlich Linien“ zum Begrenzen beim Skalieren einsetzen (siehe Bild 6.10).

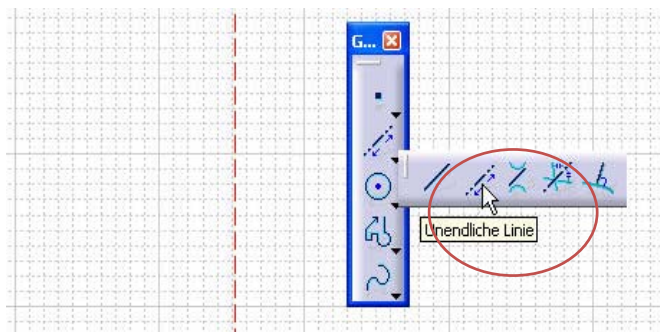


Bild 6.10 Unendlich Linien

4. Mit dieser Funktion zwei Begrenzende Linien einfügen (siehe Bild 6.11).

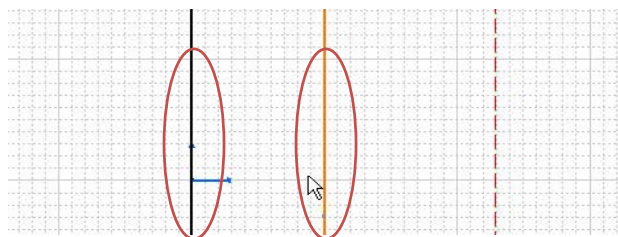


Bild 6.11 Begrenzende Linien

5. Mit der linken Maustaste die gewünschte Ansicht *markieren* (siehe Bild 6.12).



Bild 6.12 Ansicht markieren

6. Die gewünschte Ansicht zu den, im vorletzten Schritt erstellten, Grenzlinien verschieben (siehe Bild 6.13 und 6.14). Hierbei bietet es sich an, das Flugzeug so zu verschieben, dass sich beispielsweise die Rumpfspitze schon mit einer Grenzlinie überschneidet (siehe Bild 6.14).



Bild 6.13 Verschieben der Ansicht

7. Bei markiertem Modell die Funktion „Maßstab“ aktivieren (siehe Bild 6.14).

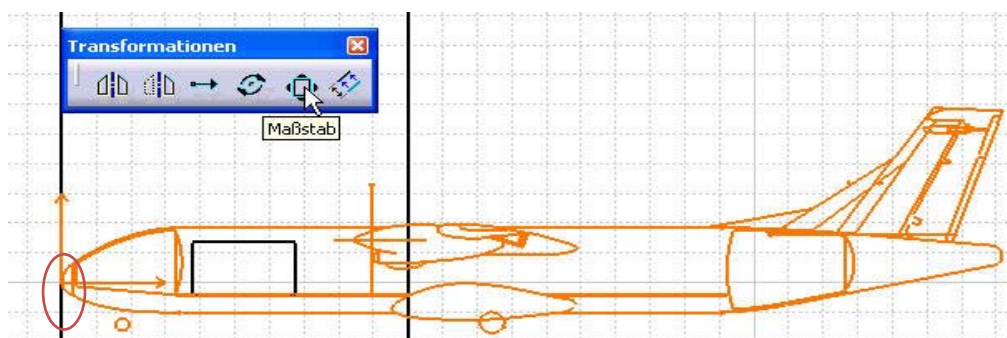


Bild 6.14 Maßstab verändern

8. Anschließend *einmal* auf den bereits überschneidenden Punkt an der Rumpfspitze „klicken“. Hierdurch ist dieser Punkt fixiert und die Geometrie wird um diesen Punkt skaliert.
9. Den Skalierungswert so einstellen, dass die gewünschte Länge erreicht wird. Erst einen ungefähren Schätzwert in das Feld eintragen, dann mit den Pfeilen (rechts neben dem Eingabefeld) präzisieren. Hierfür muss der Haken aus dem Feld Schrittmodus entfernt werden. Desweiteren sollte das Feld „duplizieren“ deaktiviert sein.
10. Sobald der richtige Maßstab gefunden wurde, mit „O.K.“ bestätigen (siehe Bild 6.15).

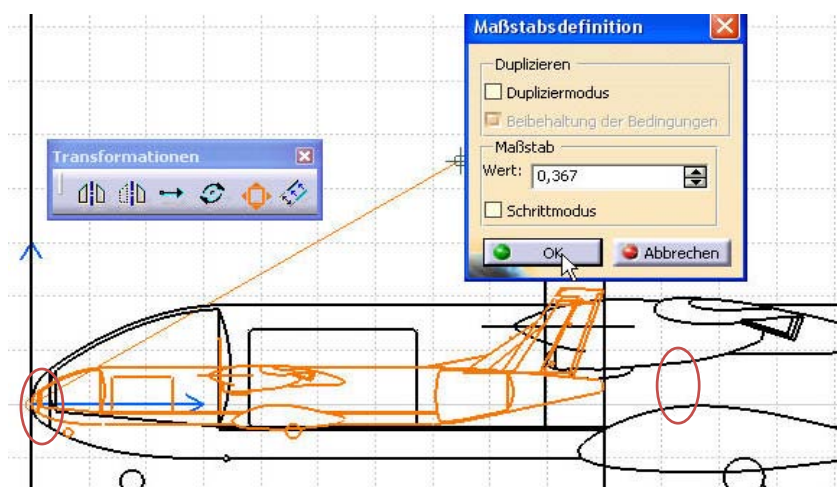


Bild 6.15 Maßstabfunktion

11. Jetzt lassen sich die normierten Maße jeder beliebigen horizontalen Koordinate in dieser Ansicht herauslesen. Zusätzlich kann man, zum schnellen Wiederfinden bestimmter Stellen, Punkte setzen (siehe Bild 6.16). In dem gezeigten Bild ist beispielsweise das Ende des Fußbodens bei 71 % der gesamten Rumpflänge.

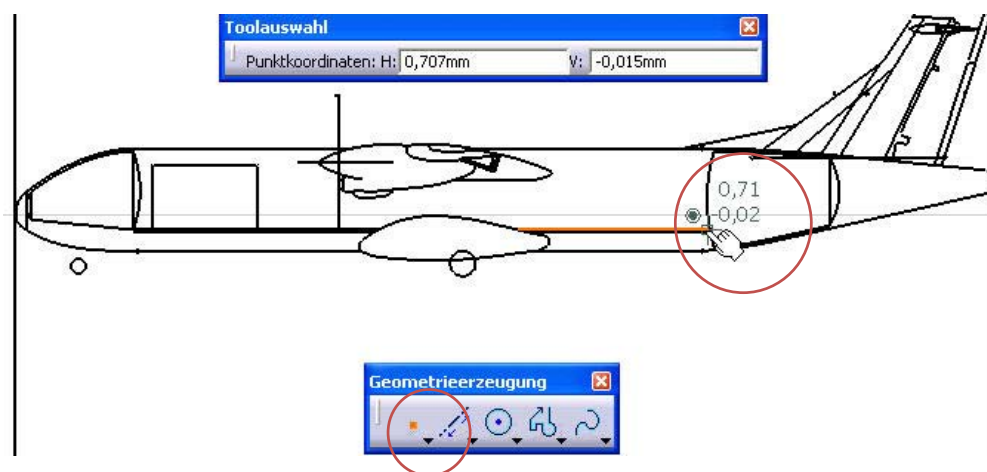


Bild 6.16 Endergebniss

6.4.2 Vergleichen von Texten

Mit dem kostenlosen Programm *WinMerge* lassen sich Texte miteinander vergleichen.²¹ Das Programm vergleicht die Texte und stellt die abweichenden Textpassagen farbig dar. Dies ist für das Arbeiten mit *PrADO*-Dateien eine sehr große Hilfe.

So lassen sich beispielsweise die Vorgabedateien, die „PRA-dbx.dat“-Dateien oder auch die einzelnen Iterationsschritte aus den E/P/O miteinander vergleichen.

Das Programm *WinMerge* lässt sich simpel und intuitiv bedienen.

Nach dem Starten lassen sich die beiden Textdatei einfach in die Bedienoberfläche hineinziehen oder über „Suchen“ importieren (siehe Bild 6.17).

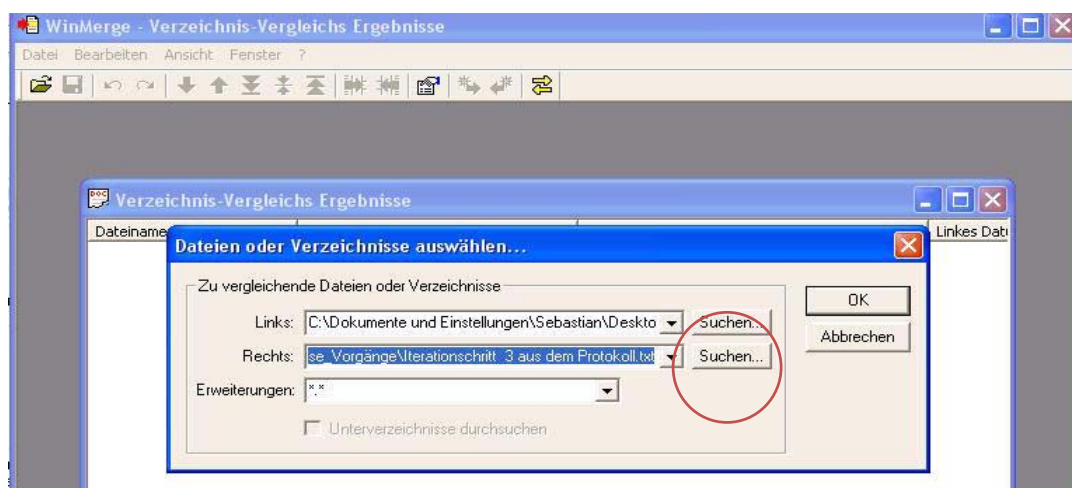


Bild 6.17 Importieren

Durch das Benutzen der gelben Pfeile in der Kopfzeile ist es möglich, zur nächsten Abweichung zu springen (siehe Bild 6.18).

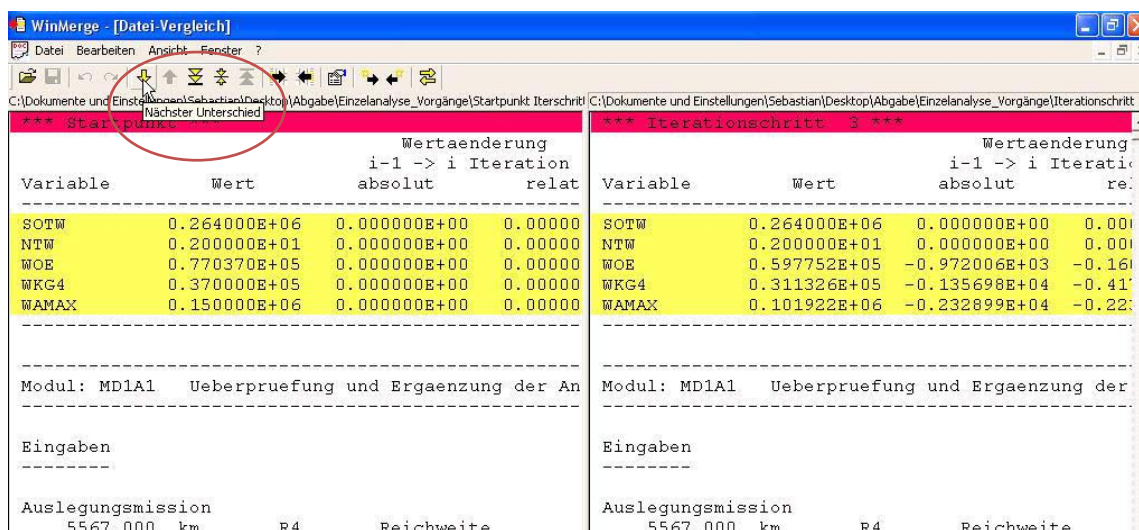


Bild 6.18 Zur nächsten Abweichung springen

²¹ Das Programm lässt sich unter der Internetadresse: <http://winmerge.softonic.de> herunterladen.

6.4.3 Ermitteln von Flächen

Wenn der Nachentwurf eines Flugzeuges erstellt werden soll und über dessen Geometrie neben einigen Bildern nur die Länge und die Spannweite bekannt sind, ist es sehr aufwendig und ggf. ungenau, die Größen der einzelnen Flächen auszurechnen. Dies trifft besonders bei geschwungenen Flächen zu.

In solchen Fällen kann das kleine Programm *A.C.Calculator*²² von Martin Heppler weiterhelfen.

Der eigentliche Zweck des Programms, der allerdings nicht weiter gebraucht wurde, besteht darin, den aerodynamischen Schwerpunkt eines Flugzeuges annähernd zu bestimmen.

Dieses Programm wurde, im Rahmen dieser Projektarbeit, benutzt um die Leitwerksfläche zu bestimmen.

Desweiteren wurden die einzelnen Flächenangaben für die Class1 Methode von **RAYMER 1989** verwendet, um den ungefähren Wert der Betriebsleermassen zu überprüfen und dadurch grobe Programmfehler auszuschließen.

Wie sich die Fläche einer Komponente aus einer beliebigen Ansicht bestimmen lässt, ist nachfolgend am Beispiel der projizierten Rumpfsseitenfläche beschrieben:

1. Zu Beginn muss die Zeichnung oder notfalls ein möglichst senkrecht aufgenommenes Foto von der zu ermittelnden Fläche mit einem Grafikprogramm wie z. B. dem in den meisten Windowsbetriebssystemen enthaltenen Programm *Paint* geöffnet werden (siehe Bild 6.19).

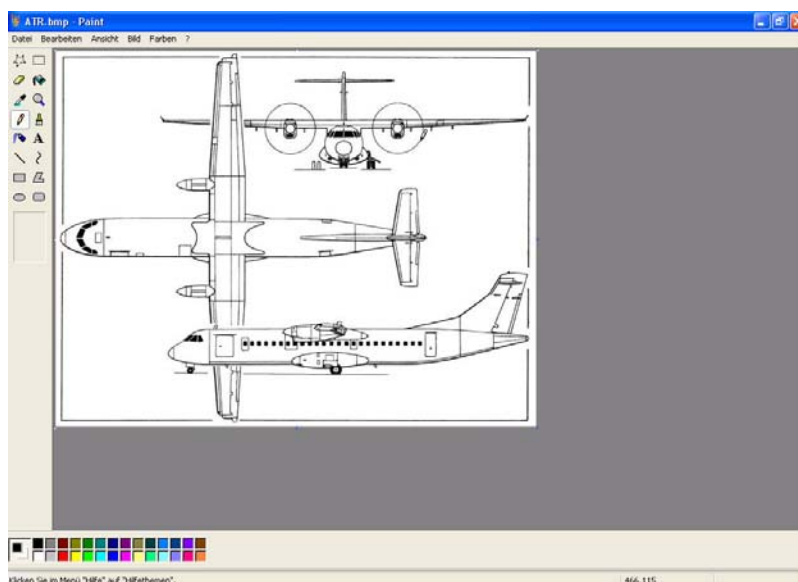


Bild 6.19 Paint

²²

Das Programm *A.C.Calculator* lässt sich kostenlos unter der folgenden Internetadresse herunterladen: <http://www.mh-aerotoools.de/airfoils/software.htm>.

2. Jetzt müssen die benötigten Flächen mit einer der Farben, die in dem Programm *A.C. Calculator* rechts oben abgebildet sind (siehe Bild 6.20), in dem Grafikprogramm eingefärbt werden.
3. Nach dem Abspeichern des eingefärbten Bildes wird der *A.C. Calculator* gestartet und das Bild wird über die Taste „Öffnen“ ausgewählt und hinein geladen.

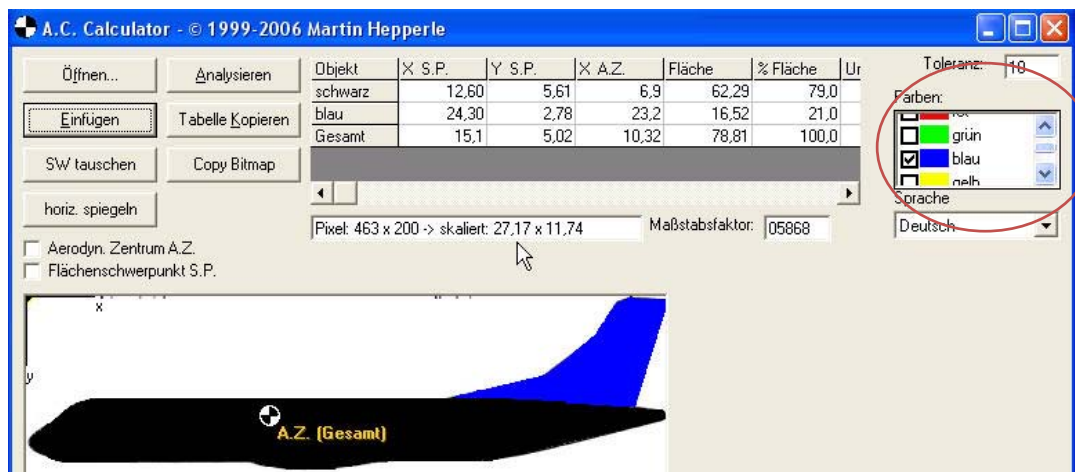


Bild 6.20 A.C. Calculator

4. Nun wird die Pixellänge in die gewünschte Einheit z. B. in Meter skaliert. Hierzu wird die Pixellänge, der bekannten geometrischen Länge, durch das Anfahren mit dem Mauszeiger ermittelt. Anschließend lässt sich durch den Vergleich mit dem tatsächlichen Maß der benötigte Maßstabfaktor bestimmen.
5. Nach dem Betätigen der Schaltfläche *Analysieren* erscheint oben in der Mitte des Programmfensters eine Tabelle mit den gesuchten Flächenangaben. In dem abgebildeten Fall beträgt die Seitenprojektionsfläche des ATR72-Rumpfes 62,29 m² und die des HLWs 16,52 m². Diese genauen Werte lassen sich manuell nur bei sehr häufigem und präzisiertem Unterteilen der Zeichnung und im Verhältnis sehr zeitaufwendigem Rechenaufwand ermitteln.

6.4.4 Verbinden von Dateien

Von *PrADO* werden nur die „PRA-dbX.dat“-Dateien, also die Dateien der Datenbasis im Projektordner, für die Dateneingabe und -ausgabe benutzt.

Die Vorgabedatei bleibt von *PrADO* immer unverändert. Aus diesem Grund muss man alle dauerhaften Änderungen umständlich manuell in der Vorgabedatei verändern. Dies ist vor allem bei komplett neuen Projekten, die aus dem Masterfile oder der Vorgabedatei eines anderen A/C nach und nach erstellt werden, sehr aufwendig. Mit dem Programm

*Simple File Joiner*²³ lassen sich die einzelnen „PRA-dbX.dat“-Dateien schnell und unkompliziert

²³

Das kostenlose Programm „FileJoin“ lässt sich unter der folgenden Internetadresse herunterladen:
<http://www.peretek.com>

zu *einer* Vorgabedatei zusammensetzen. Wenn man zusätzlich am Anfang oder Ende der Datei die Beschreibung und Erklärung aller Variablen als Textfile hinzufügt, erhält man eine Vorgabedatei mit der man gut arbeiten kann und die so alle Änderungen sicher gespeichert hat.

Im Anschluss an das Erzeugen einer neuen Datei aus mehreren einzelnen Dateien, muss der Name der Datei ggf. noch mit der Endung „.dat“ ergänzt werden.

6.4.5 Eine Alternative zum Editor

Mit dem Programm *VrmIPad*²⁴ lassen sich Texte wie die Vorgabedatei leicht bearbeiten und übersichtlich darstellen. Besonders hilfreich an *VrmIPad* ist:

- das leichte Setzen von gut sichtbaren Markierungen über das Betätigen der blauen Fahne in der Mitte des Kopfes des Programmfeldes (siehe Bild 6.21)
- das schnelle hin und her springen zwischen den Markierungen
- die Möglichkeit Karteikarten anzulegen, um z.B. schnell und geordnet auf die Dateien anderer, ähnlicher Projekte zugreifen zu können (siehe Bild 6.21)

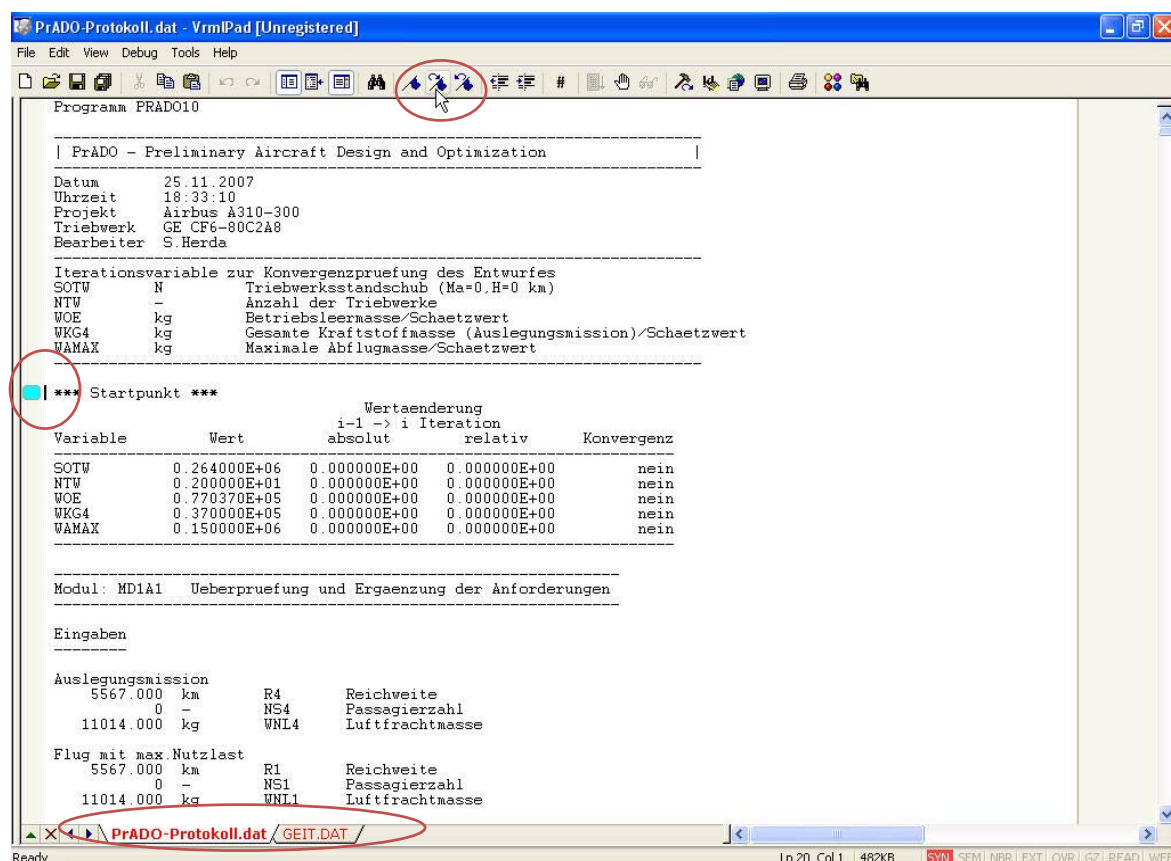


Bild 6.21 VrmIPad

7 Anwendung an der ATR 72-200F

Die Aufgabe bestand darin, die Vorgabedatei der ATR72-200 Frachtversion (siehe Bild 7.1) so zu modifizieren, dass PrADO damit, nach dem Programmupdate vom 01.12.06, nicht nur die Flugzeuggeometrie erzeugen kann, sondern auch bei einer E/P/O nicht abbricht.

Da nach diesen Arbeiten das Ende dieser Projektarbeit erreicht war, konnten an der erstellten ATR72-200F nicht noch die Möglichkeiten der Parametervariation oder der Optimierung getestet werden.

Desweiteren gehen viele in dieser Arbeit beschriebenen Erfahrungen aus dem Arbeiten an der ATR72-200F hervor. In diesem Kapitel folgen nur die konkreten Änderungen an der ATR72-200F-Vorgabedatei von Thorsten Kiesel.



Bild 7.1 Frachtraum der ATR72F (ATR 2007)

7.1 Allgemeines zur ATR

Da es während dieses Projektes mehr um das Arbeiten mit PrADO ging, als um die ATR72, die während des Arbeitens mit PrADO benutzt und verfeinert wurde, macht es an dieser Stelle keinen Sinn die ATR72-200 allzu detailliert zu beschreiben. Aus diesem Grund wurden in dieses Kapitel nur vereinzelte Informationen aufgenommen. Als Ergänzung zu den Informationen zur ATR in der Arbeit von Thorsten Kiesel, sei hier nur darauf hingewiesen, dass die ATR72 als weltweit erstes kommerzielles Flugzeug einen Außenflügelkasten aus Kohlefaserverbundwerkstoffen hatte, wodurch einige statistische Werte auf dieses Flugzeug nicht zutreffen. Der angesprochene Außenflügelkasten aus Kohlefaserverbundwerkstoffen ist im Bild 7.2 auf der linken Seite als graue Fläche zu erkennen.



Bild 7.2 Außenflügelkasten (ATR 2007)

7.2 Was war der Stand?

Die Vorgabedatei der ATR von Thorsten Kiesel funktionierte nach dem Update vom 01.12.06 nicht mehr, da

- zu den Variablen, die PrADO benötigt, neue hinzugekommene sind.
- die Gewichtsinformationen nicht mehr verstreut in vielen Modulen, sondern kompakt in der Datenbank Nr. 10 Massen und Schwerpunktlagen abgefragt werden.
- das Triebwerksmodul verändert wurde.

Nach dem Update ließen sich viele Module wie z.B. das SLW auch nicht mehr manuell starten. Somit waren an vielen Stellen Abänderungen notwendig.

Desweiteren beinhaltete die Vorgabedatei sowie die Template-Dateien für den Rumpf und das Profil noch einige Ungenauigkeiten.

7.3 Vorgenommene Veränderungen

7.3.1 Das SLW

Nach dem Update auf die neue Version sah das SLW wie in Bild 7.3 aus.

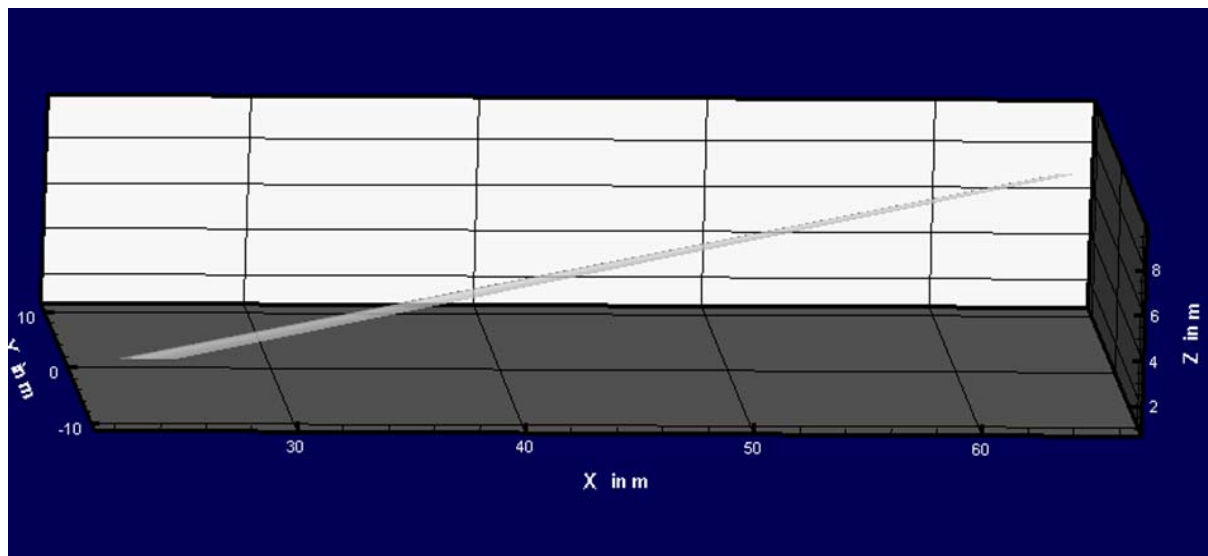


Bild 7.3 Vorheriger Stand

Die folgenden Änderungen wurden an den Vorgaben des Seitenleitwerks durchgeführt:

- es wurden die Holmlagen und die relativen Koordinaten der Schnitte, wie in Kapitel 5.2.2 beschrieben, mit dem Programm Catia ermittelt.
- die Streckung wurde so lange in feinen Schritten variiert, bis sich ein gleichmäßiger Verlauf der SLW-Hinterkante ergab. In den Vorgaben müssen hierfür nicht nur die relativen Maßangaben und Positionen der verschiedenen Schnitte fehlerfrei eingetragen sein, sondern auch die Streckung. Falls die Streckung nicht genau genug angegeben wird, sieht das SLW anschließend beispielsweise so wie auf den Bildern 7.4 und 7.5 aus.

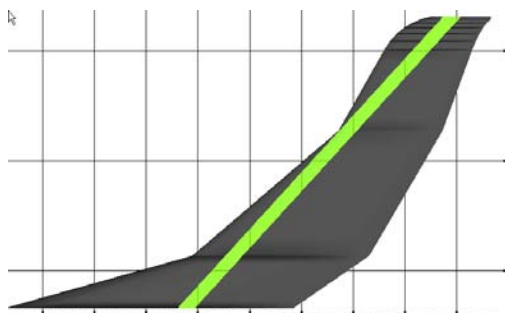


Bild 7.5 Hohe Streckung



Bild 7.4 Geringe Streckung

- Das anzugebende Gewicht der SLW-Hinterkante wurde mit 20 kg/m³ festgelegt.

In PrADO ist es zurzeit eigentlich noch nicht möglich, Ausgleichshörner nachzubilden. Ein Problem hierbei ist, dass alle Ruderschnitte mit einem Holm beginnen müssen. Dieses Problem wurde umgangen, indem ein imaginärer Holm an die Vorderkante des Ausgleichshorns gesetzt wurde.

Die Schwierigkeit hierbei bestand darin, dass zum einen viele Schnitte notwendig gewesen sind und zum anderen alle Schnitte eine andere, auf die jeweilige Profiltiefe bezogene, relative Position der Drehachse benötigten. Bei Abweichungen bezüglich der Drehachsen, fächerte sich das Seitenruder während des Ausschlagens auf. Hierfür wäre die Möglichkeit, zwei globale Punkte für die gewünschte Drehachse wählen zu können, eine wertvolle Verbesserung.

Ob alle Module mit dem so erschaffenen Ausgleichshorn fehlerfrei funktionieren, konnte noch nicht überprüft werden.

Das neue SLW ist, nach allen vorgenommenen Veränderungen, in Bild 7.6 abgebildet. In diesem Bild 7.6 ist ein Abschnitt aus der Vorgabedatei zu sehen. In diesem Abschnitt kann man die ermittelten Daten der Schnitte erkennen.

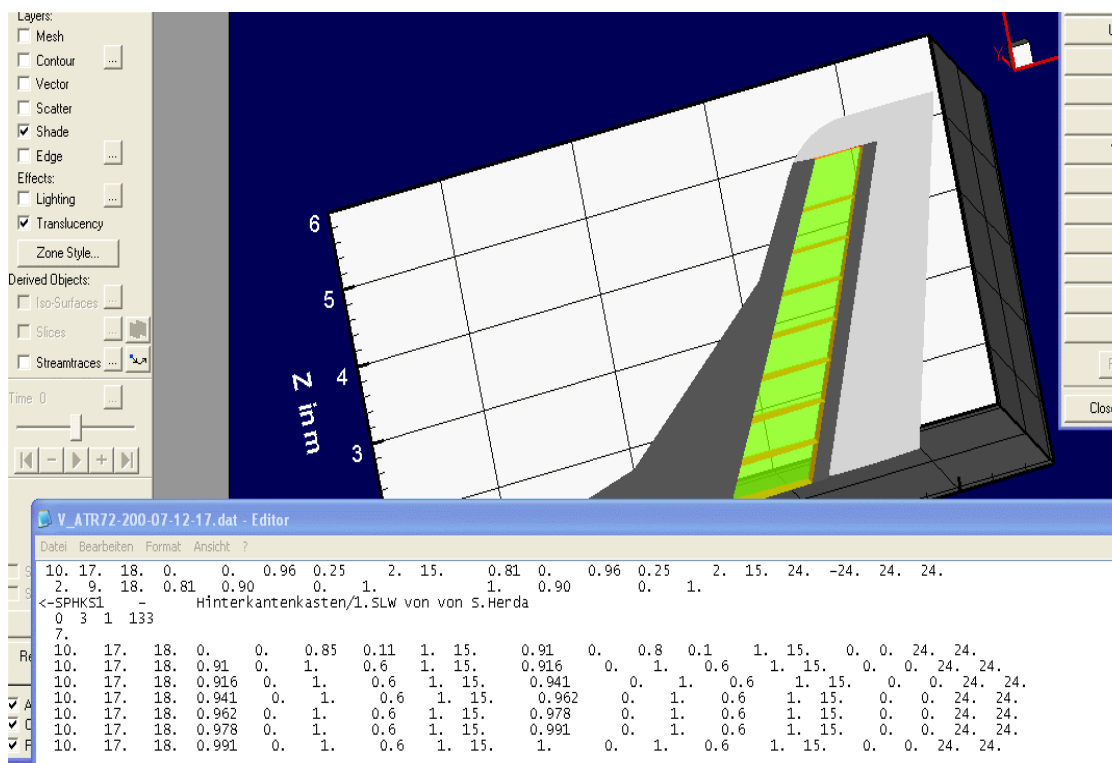


Bild 7.6 Das fertige SLW

7.3.2 Die Flugzeugsysteme

Die unter anderem in der Datenbank Nr. 10 Massen und Schwerpunktlagen benötigten Detailinformationen zu der Art und der Position der eingebauten Flugzeugsysteme,

konnten mit dem Buch „*Avions de Transport Regional*“ von Hans-Jürgen Becker und den Informationen aus der Broschüre „*ATR 42-300/320 and ATR 72-200/210*“ des Herstellers ATR (siehe Bild 7.7) in Bezug auf viele Angaben, die bereits in der Vorgabedatei enthalten waren, bestätigt werden. Allerdings ließen sich nicht alle benötigten Informationen ausfindig machen. An diesen Stellen wurden die noch von der Do 328 vorhandenen Daten einfach übernommen. Diese Daten waren in der Vorgabedatei enthalten, da Thorsten Kiesel die Vorgabedatei der Do 328 als „Gerüst“ für die ATR72 verwendet hatte.

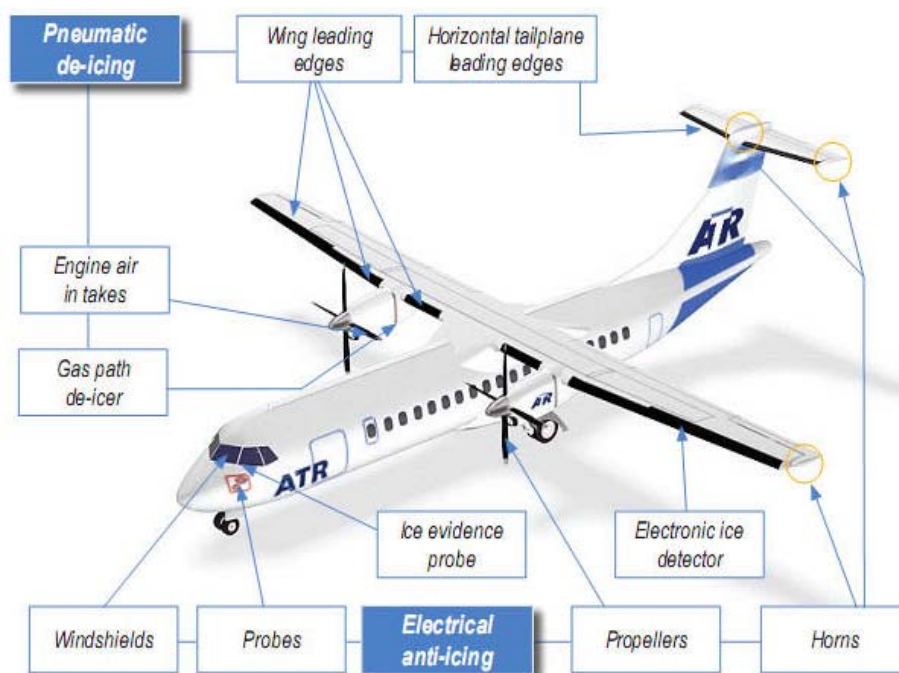


Bild 7.7 Flugzeugsysteme (ATR 2007)

7.3.3 Die Rumpfdati und die Cockpitfenster

In dem „Template“-Ordner befindet sich die Rumpfdati. Diese Datei wird durch eine Zeile in der Datenbank Nr.4 aufgerufen. In dieser Rumpfdati wurden folgende Veränderungen durchgeführt:

- Der Verlauf der Rumpfkonturen wurde durch das Verändern der Werte, besonders im Bereich des Fensters, verfeinert. In den beiden Bildern 7.8 und 7.9 ist die erreichte Verbesserung deutlich zu erkennen.

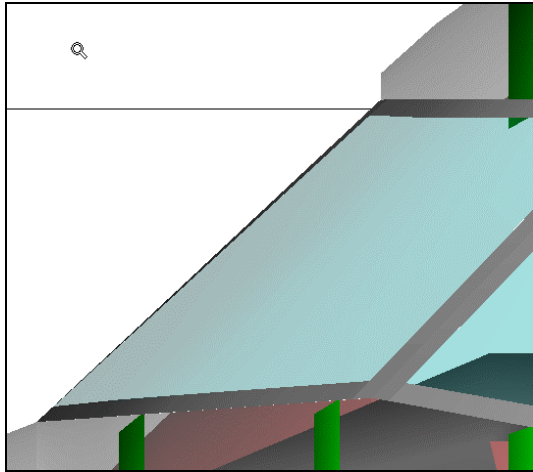


Bild 7.8 Vorher

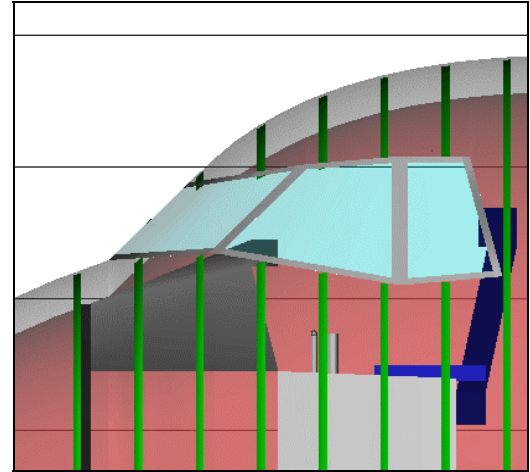


Bild 7.9 Nachher

- Die vorher vorhandene Cockpitfenstergeometrie (siehe Bild 7.11) wurde überprüft. Anschließend wurde der X-Fensterkoordinaten des vierten Punktes ein kleinerer Wert zugewiesen. Hierdurch konnten die Scheiben im Stirnbereich an die Realität angepasst werden (siehe Bild 7.10).
- Die Spantposition musste leicht nach hinten verschoben werden, um keine Verschneidungen der Spanten mit dem Cockpitfenster zu haben.(siehe Bild 7.12)

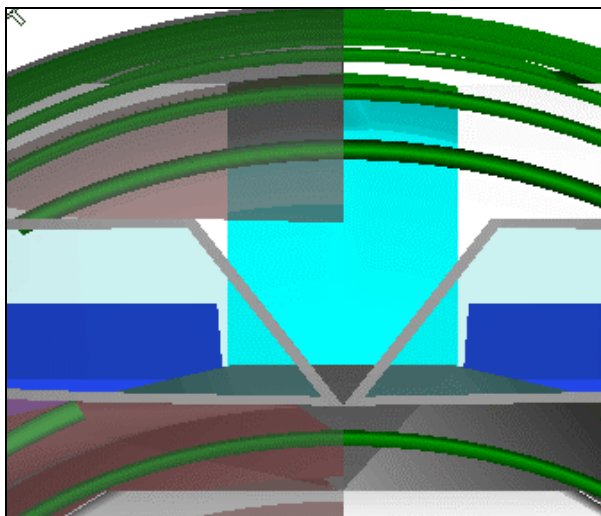


Bild 7.11 Fehler in der Fenstergeometrie

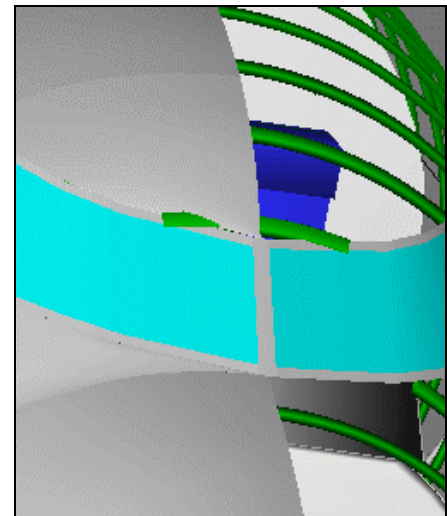


Bild 7.10 Fehlerhafte Rippenlage

Nach den Veränderungen sah die Geometrie wie in Bild 7.12 aus.

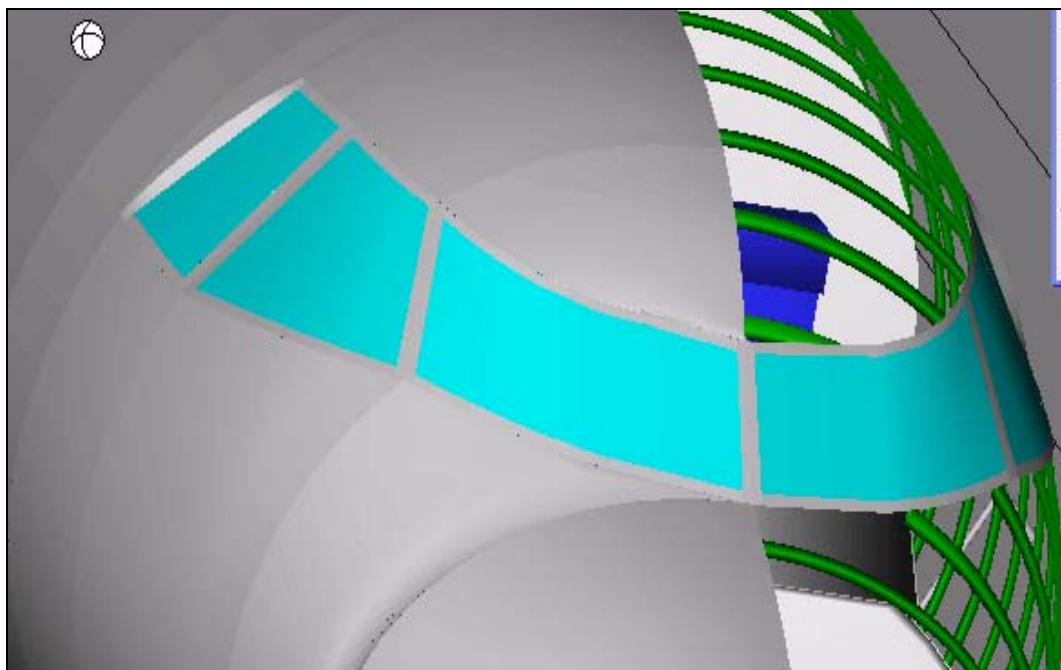


Bild 7.12 Das Ergebnis des Rumpfes

7.3.4 Die Profildatei

Für das SLW wurden zwei verschiedene Profile verwendet, damit der Teil der Rückenflosse einen gleichmäßigen Übergang zum restlichen SLW hat (siehe Bild 7.13 und 7.14). Andernfalls würde durch die gleiche relative Profildicke und die gleiche Dickenrücklage ein unförmiger Verlauf entstehen.

Die hierfür notwendigen relativen Abmessungen wurden mit dem in Kapitel 6.5.1 beschriebenen Verfahren ermittelt.

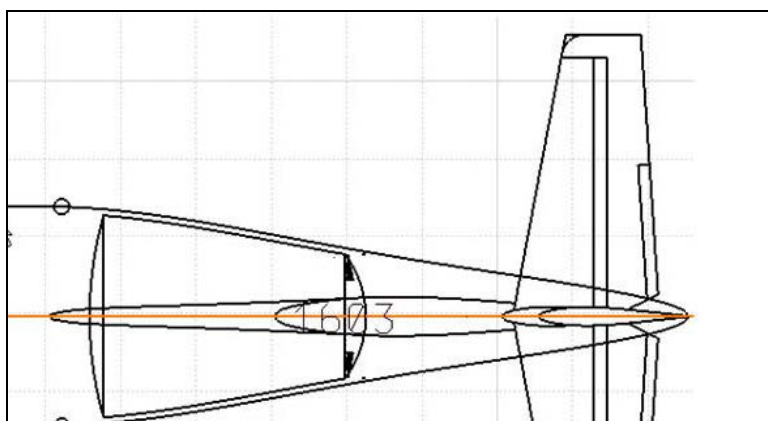


Bild 7.14 Draufsicht Profilgeometrien des SLW

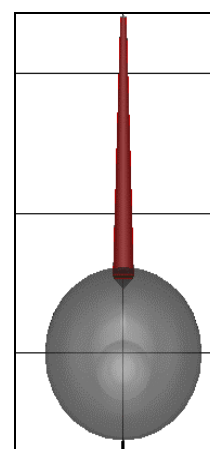


Bild 7.13 Front des SLW

Hinweis:

Bei der Profildatei ist darauf zu achten, dass der erste und der letzte Punkt, der zu beschreibenden Ober- bzw. Unterseite, dieselben Werte aufweisen. Sollte dies nicht der Fall sein, dann bricht der Programmablauf ohne jeglichen Hinweis ab.

7.3.5 Die Triebwerksdatei

Das Turbofantriebwerk *PW306*, das in der Arbeit von **Kiesel 2007** als Turbofansatz des original Triebwerkes *PW124B* eingesetzt wurde musste gegen ein stärkeres Triebwerk getauscht werden. Dies war nötig, da es für die Anforderungen zu schwach gewählt worden war und somit die Einzelanalyse von *PrADO* nicht erfolgreich durchlaufen werden konnte. Das neue, in die Vorgabedatei übernommene Ersatztriebwerk der ATR besteht aus dem *CF34-3B1* Turbofantriebwerk.

7.4 Die Ergebnisse des Nachentwurfes

Um eine Basis für noch genauere Berechnungen zu schaffen, soll an dieser Stelle versucht werden, die Hintergründe der Abweichungen zu beleuchten bzw. Erklärungen zu finden, warum Unterschiede in zukünftigen Berechnungen mit der erzeugten Vorgabedatei möglicherweise zu erwarten sind.

7.4.1 Ergebnisse der Machbarkeitsstudie

Die Ergebnisse der durchgeführten Überprüfung der Machbarkeit, also der Einzelanalyse, wurden von *PrADO* im Kurzprotokoll ausgegeben. Dieses, aus der behandelten ATR72-200F gewonnene, Kurzprotokoll ist im Anhang A hinterlegt.

7.4.2 Abweichungen zur realen ATR

Welche Abweichungen zu der realen ATR72-200F bestehen noch bzw. woraus können Unterschiede in den Ergebnissen resultieren:

- der Faserverbundanteil (siehe Bild 7.15) inklusive des oben bereits erwähnten Außenflügelkastens aus Kohlefaserwerkstoffen wurde noch nicht ausreichend berücksichtigt.
- die ATR72-200 hat anstelle der Turboprop-Triebwerke *PW124B* zwei Turbofantriebwerke vom Typ *CF34-3B1*.
- die Ausgleichshörner des HLWs, SLWs und der Tragfläche lassen sich noch nicht vollfunktions-fähig nachbilden bzw. ihre oben beschriebene Nachbildung muss in Bezug auf die fehlerfreie bzw. realistische Nachbildung innerhalb des Programmlaufs, erst noch validiert werden

Hierbei ist zu beachten, dass durch die eben aufgezählten Punkte neben geringen Abweichungen der entscheidenden Werte (z.B. Betriebsleermassen) viele im Verhältnis eher nebensächliche Werte (z.B. Lärmausbreitung), stark abweichen können und somit unbrauchbare Informationen darstellen.

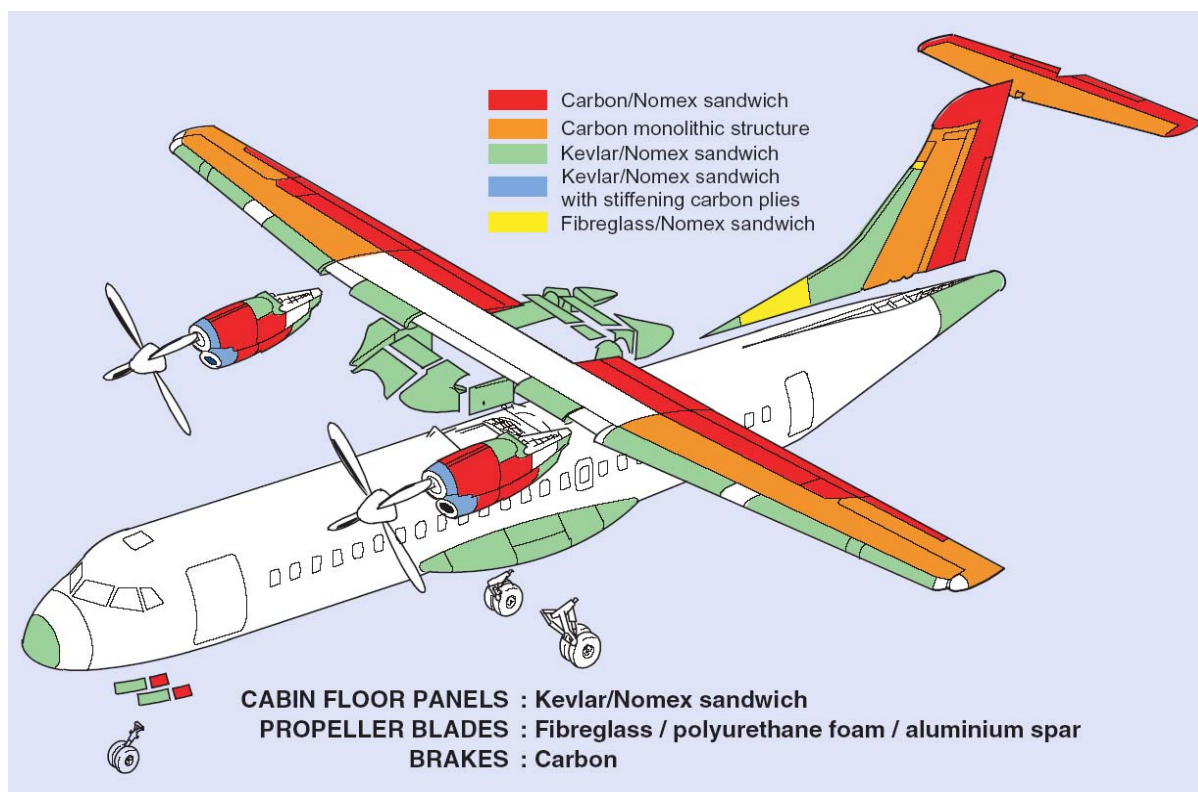


Bild 7.15

Faserverbundanteil der ATR72 (ATR 2007)

8 Entwicklungspotential von PrADO

Neben vielen, in PrADO hervorragend gelösten Details, wie beispielsweise

- dem sehr transparenter Aufbau
- den Zugriffsmöglichkeiten auf alle Unterprogramme
- dem sichtbaren Programmlauf, mit dessen Hilfe man durch das Nachvollziehen der einzelnen Schritte das Ergebnis bewerten kann. Desweiteren kann man bei Fehlern so sofort eingreifen, indem man den Programmlauf stoppt, den Fehler behebt und anschließend das Programm erneut startet.

gibt es ein paar Verbesserungs- und Ergänzungsmöglichkeiten. Diese Möglichkeiten sollen in diesem Kapitel beschrieben werden, um die Benutzerfreundlichkeit zukünftiger Programmversionen zu erhöhen.

8.1 Noch zu beseitigende Missstände

An dieser Stelle sollen die noch vorhandenen Fehler kurz aufgeführt werden.

8.1.1 In den FORTRAN-Programmtexten

Manche Ordner bzw. Bibliotheken, die in den FORTRAN-Programmen auftauchen, gibt es in den PrADO-Ordern nicht unter dem gleichen Namen (siehe Bild 8.1).

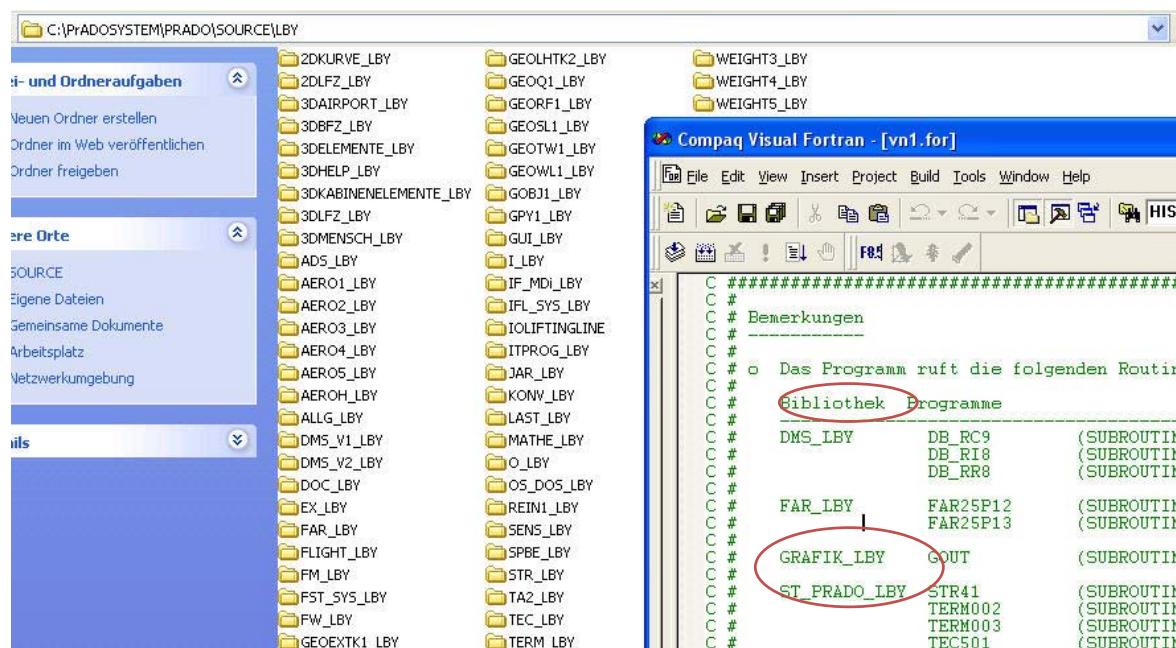


Bild 8.1 Fortran-Programme

8.1.2 In der Vorgabedatei

Die Beschreibung im SLW-Bereich, also der Datenbank Nr. 6, wurde teilweise aus der Tragflächenbeschreibung übernommen, wodurch z.B. die Längsrichtung nicht durch die relative Z-Koordinate sondern durch die Y-Koordinaten beschrieben wird.

8.1.3 In der GUI

In der GUI-Oberfläche für die Position des Flügels fehlt ein Eingabefeld für den „7. Parameter“ (siehe Bild 8.2).

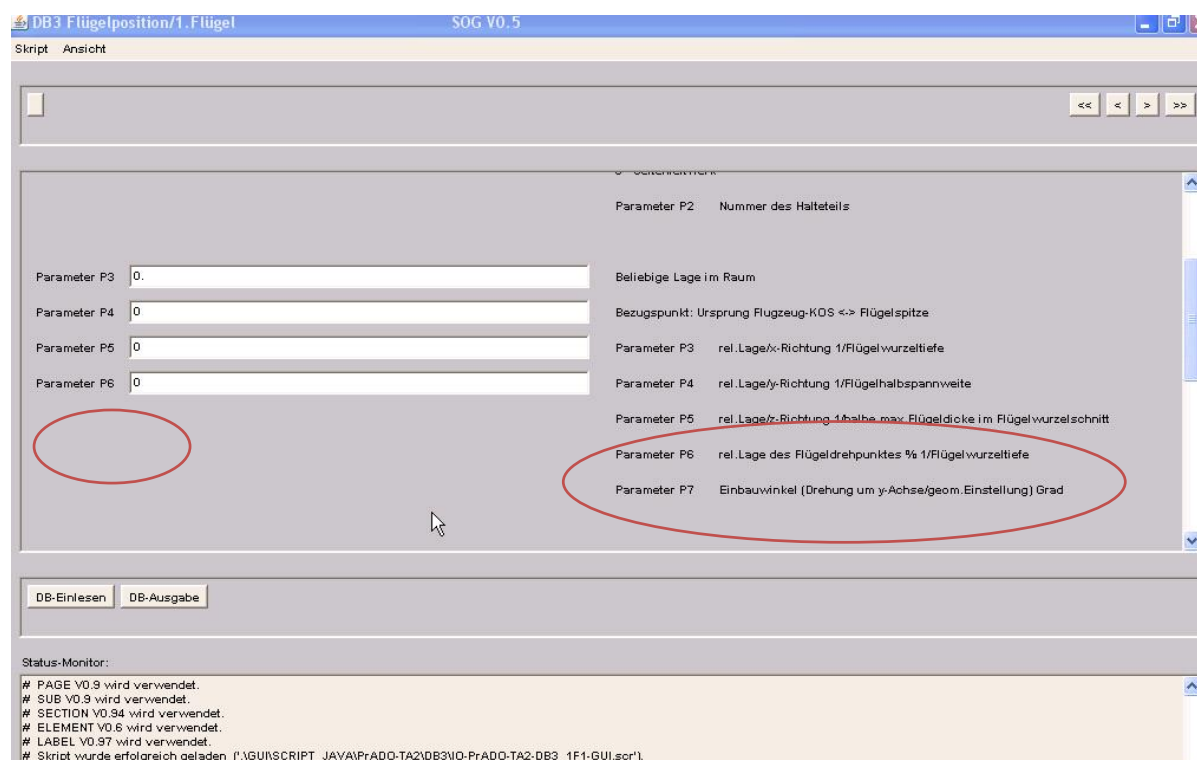
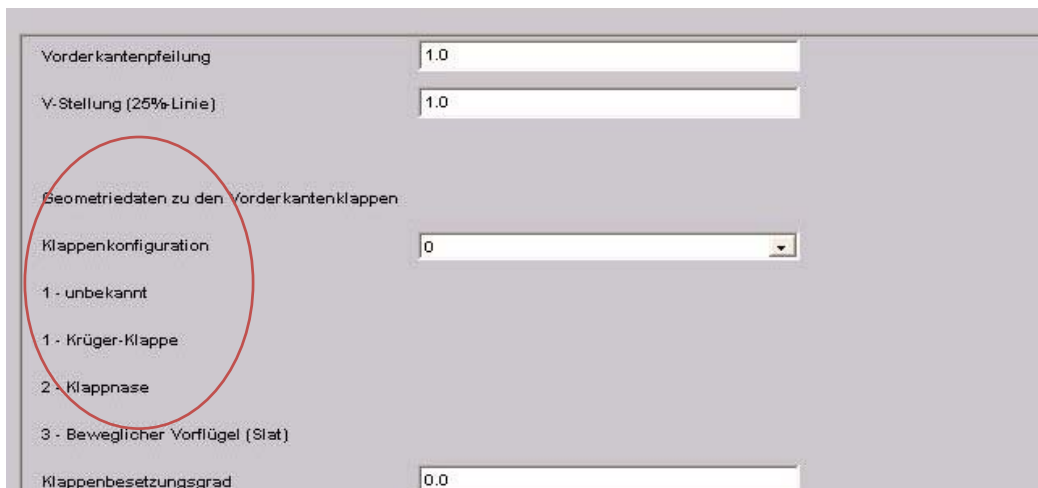


Bild 8.2 Position des Flügels

Zu dem oberen Bild 8.2 ist noch zu erwähnen, dass eine andere Anordnung und Unterteilung der Beschreibungen die Übersichtlichkeit erhöhen würde.

Desweiteren sind in den GUI-Beschreibungen noch ein paar kleine Fehler. So ist beispielsweise der einzugebende Wert in der „DB3“ bei der Wahl zur Klappenkonfiguration nicht stimmig (siehe Bild 8.3). Hier ist der einzugebende Wert für die Wahl der Klappenkonfiguration bei der „unbekannt“-Variante und der Krüger-Klappe mit dem Wert eins angegeben.



Vorderkantenpfeilung 1.0

V-Stellung (25%-Linie) 1.0

Geometriedaten zu den Vorderkantenklappen:

Klappenkonfiguration 0

1 - unbekannt

1 - Krüger-Klappe

2 - Klappnase

3 - Beweglicher Vorflügel (Stat)

Klappenbesetzungsgrad 0.0

Bild 8.3 Wahl zur Klappenkonfiguration

8.1.4 Im Programmlauf

In der benutzten Programm-Version treten bei den Fehlermeldungen gelegentlich falsche Variablen-Kürzel auf, die es nicht (oder nicht mehr) unter den aktuellen DMS-Variablen gibt. Hierdurch kann das Beseitigen des Fehlers viel Zeit in Anspruch nehmen.

8.2 Verbesserungsideen

8.2.1 Allgemein

6. Die Geometrieerzeugung über ein CAD-Programm wie beispielsweise Catia würde eine große Verbesserung darstellen
- Beim Beobachten der Prozesse, die Lifting Line einbeziehen, ist im Programmfenster und im Taskmanager auffällig, dass viel Zeit für das Starten und Beenden von Lifting Line verloren geht. Aus diesem Grund wäre es eine Verbesserung, Lifting Line so abzuändern bzw. das Zwischenspiel zwischen PrADO und Lifting Line so zu verändern, dass Lifting Line nicht ständig neu gestartet und beendet werden muss.
 - Für spätere Versionen würde ein Überarbeiten mancher „Befehlsnamen (<_...)“ in selbst erklärende Abkürzungen die Benutzerfreundlichkeit erhöhen.
 - Die Abhängigkeit von statistischen Werten sollte weiter verringert werden. Hierdurch würde man bei Konfigurationen, die besonders weit von der statistischen Datenbasis entfernt sind, genauere Ergebnisse erhalten.

- Eine Vereinfachung der Installation wäre zu begrüßen. Zurzeit besteht die lang andauernde Installation noch aus den folgenden Schritten:
 1. Programm auf den Rechner kopieren
 2. Systemvariablen erstellen
 3. Dateien für FORTRAN übersetzen und verlinken
- Für einzugebende aber unbekannt Werte könnte man z.B. durch das Setzen eines vorher definierten Zeichens, einen Standardwert erzeugen lassen. Dies ist zurzeit noch nicht möglich. Dieses Problem könnte ggf. auch einfach durch die Angabe eines ungefähren Wertebereichs in den Kommentarzeilen gelöst werden.
- Es sollten alle für die Geometrie notwendigen Eingaben einheitslos gestaltet werden. Zwar ist das größtenteils schon der Fall, allerdings gibt es auch Eingaben wie z.B. die Cockpitfenstergeometrie, die in Meter angegeben werden müssen.
- Verbessern bzw. Erweitern der Beschreibungen in den Vorgabedateien und vereinzelt in den *FORTRAN*-Texten
- Deaktivieren der automatischen Erzeugung der Ordner *Nastran*, *Zaero* und *SAM* während des Einlesens der Vorgabedatei. Diese Ordner sind nutzlos, da die Programme, die diese Ordner verwenden, nicht in der verwendeten *PrADO*-Version enthalten sind.
- Vereinfachung der Möglichkeit einzelne Punkte zu verändern, ohne dass dann in verschiedenen anderen Modulen Fehler auftreten. Ein Beispiel hierzu ist der Versuch, das Einrichten der Decks zu deaktivieren, um Rechenzeit zu sparen. Hierfür kann man nicht einfach, wie im folgenden Bild 8.4 zu sehen, den Haken in der GUI entfernen, sondern muss mehrere Variablen verändern oder deaktivieren. Durch das Entfernen des Hakens treten, bei sonst gleichen Einstellungen, mehrere Fehlermeldungen im Programmlauf auf, die in vorigen Programmläufen nicht vorkamen.



Bild 8.4 Haken in der GUI entfernen

8.2.2 In der GUI, dem Programmlauf und der Vorgabedatei

In der GUI:

- Mit dem DMS-Editor (siehe Bild 8.5) lassen sich die jeweiligen Werte und Beschreibungen aller Variablen durch kopieren und einfügen des Variablennamens in das Feld „gesuchte Variable“ darstellen. Diese Möglichkeit des Abrufens hat sich beim Arbeiten mit PrADO als sehr hilfreich erwiesen. Hier wäre ein Darstellen der Variablenliste als Schaltflächen, die die gewünschte Variable durch Anklicken aufrufen, eine einfache aber wirkungsvolle Steigerung des Komforts.

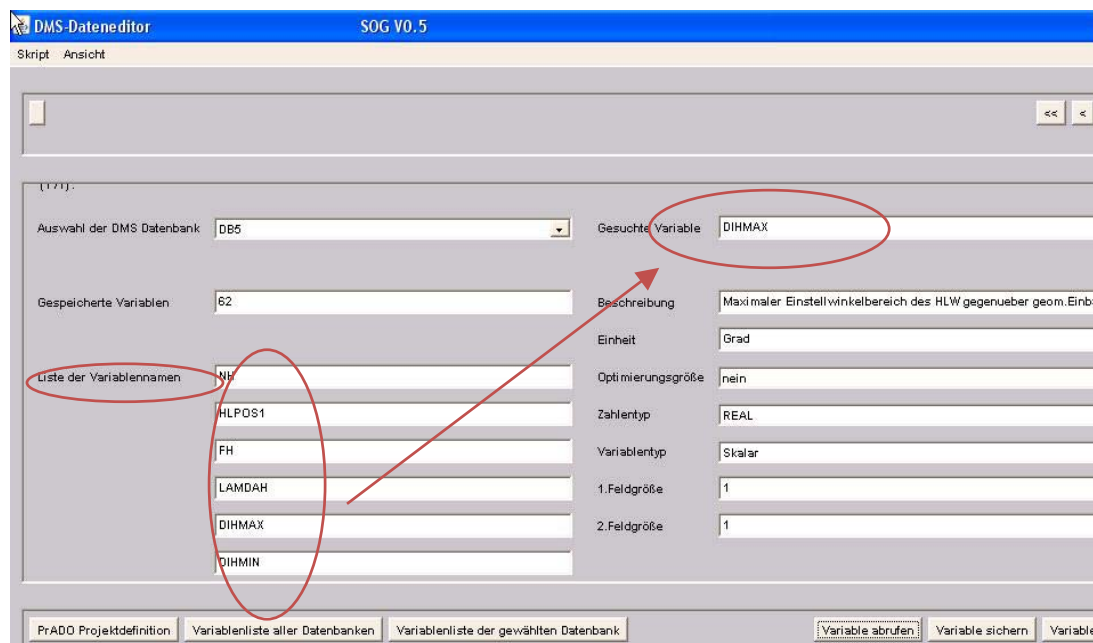


Bild 8.1 DMS-Editor

- Das „db-Einlesen“ sollte automatisch mit dem Öffnen der GUI-Oberflächen erfolgen. Da im Falle des Vergessens, durch das Ändern einzelner Werte und anschließendem „db-Auslesen“ die Werte in der Datenbasis mit den zufällig in der GUI stehenden Werten überschreibt. Dies würde die Datenbasis unbrauchbar machen.
- Es wäre hilfreich, z.B. hinter die Eingabefelder in der Java-Oberfläche, die Abkürzungen der Variablen hinzuzufügen, um diese schnell in der Vorgabedatei bzw. der DMS finden zu können. Dies ist sinnvoll, da zum Teil die ausgeschriebenen Bezeichnungen, die vor den Eingabefeldern stehen, nicht in der Vorgabedatei oder der DMS auftauchen.
- Eine zusätzliche GUI-Oberfläche, in der man:
 1. sehen kann, welcher Lastfall wie viel an der Dimensionierung beigetragen hat.
 2. die Lastfälle einfach skalieren kann, um daraus Schlüsse auf möglichen Nutzen von z. B. entlastend wirkenden Klappen etc. ziehen zu können.

- in der GUI-Oberfläche „3D-Flugzeug“ im *Main Window* unter „Darstellung/Tecplot“ wäre eine Möglichkeit zum automatischen markieren aller Felder in manchen Fällen sehr hilfreich.
- Eine Schaltfläche in der GUI-Oberfläche, die bei Betätigung den Projektordner öffnet, würde die Benutzerfreundlichkeit erhöhen.

Die nachfolgenden Punkte sollten optisch hervorgehoben werden.

- Bei der GUI-Oberfläche wäre ein optisches Abheben der Eingabefelder von den Ergebnisfeldern (siehe Bild 8.5) sinnvoll, um die Übersicht zu behalten. Das Fehlen dieser Kennzeichnung ist besonders verwirrend, da man in die Ergebnisfelder Werte eintragen und dann sogar auf „DB-Ausgabe“ drücken kann. Erfreulicherweise verändert sich die Datenbank in diesem Fall, durch das Drücken auf „DB-Ausgabe“, nicht.



Bild 8.2 Ergebnisfelder

- Wenn es in der GUI-Oberfläche mehrere Seiten zu einem Thema gibt, dann ist diese Information zu unauffällig, und lässt sich leicht übersehen (siehe Bild 8.6).



Bild 8.3 Mehrere Seiten

Die nachfolgenden Punkte sollten zeitweise, z.B. durch Ablenden oder durch Wechseln der Farbe, deaktiviert bzw. kenntlich gemacht werden.

- Ausgeschaltete Module lassen sich trotzdem noch starten (siehe Bild 8.7). Um die Übersicht zu erhöhen, sollten die Schaltflächen der in der Programmsteuerung ausgeschalteten Module gekennzeichnet oder ausgeblendet werden.

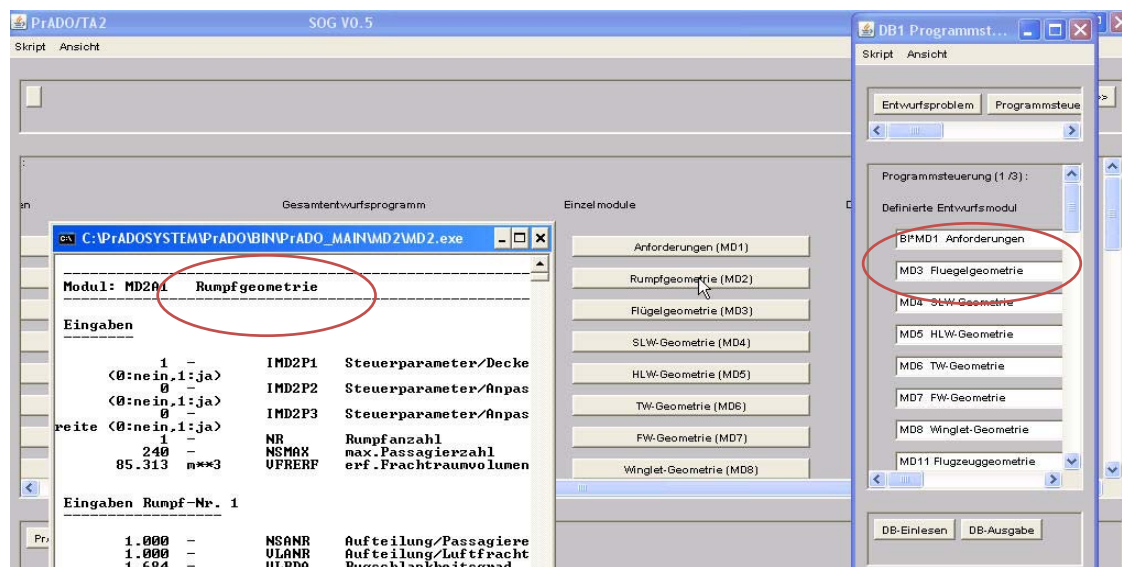


Bild 8.4 ausgeschalteten Module

- Innerhalb der GUI-Oberflächen „3D-Flugzeug“, im Bereich „Diagramme/Tecplot“, ist es nur möglich, sich etwas darstellen zu lassen, wenn die dafür benötigten Daten bereits erzeugt und anschließend im Ordner „Plot“ abgelegt worden sind. Dies geschieht durch das vorherige Aufrufen der jeweiligen Module. Hier wäre ein Ausblenden der Schaltflächen, der noch nicht durchlaufenen Module, sinnvoll.
- Für die benutzte PrADO-Version wäre ein Entfernen oder Ablenden der funktionslosen Schaltflächen „Flutteranalyse (MD23)“ in der allgemeinen „PrADO/TA2“ GUI-Oberfläche und der Schaltfläche „SAM/HISS“ aus der „Diagramme“-GUI-Oberfläche eine Verbesserung, da diese so nur verwirrend wirken. Desweiteren wäre ein Entfernen oder Ausblenden der Schaltfläche „SAM/ZAERO“ in der GUI-Datenkonvertierung, zu der man über das *Main Window* gelangt, sinnvoll, da die hierüber ansprechbaren Dateien nicht existieren.
- Ausblenden der Schaltflächen in der GUI „3D-Flugzeug“, wenn die dafür jeweils notwendigen Module *noch nicht* gestartet wurden

Im Programmlauf:

Oftmals wird eine Optimierung damit beendet, dass die ein oder andere Randbedingung verletzt wurde (siehe Bild 8.8). Hier wäre der Name der Randbedingung direkt in der Fehlermeldung und ggf. sogar eine Aufzählung aller bzw. einiger Möglichkeiten, die zum Erfüllen der Randbedingung beitragen könnten, eine geniale Ergänzung.

```

*** Iterationsabbruch ***
Summe/Variablendifferenzen   DDD =      0.00000E+00
Iterationsschritte           IZ1 =      7 < Maximum =    50

????????????????????????????????????????????????????????????????????????????????????
Meldung vom Unterprogramm - OPTIM

  Der Startentwurf verletzt die Randbedingungen
  LI = 8 RB( 8)
  Aendern sie die Eingaben fuer den startentwurf !!!

????????????????????????????????????????????????????????????????????????????????????

```

Bild 8.5 Fehlermeldung

In der Vorgabedatei:

In der Vorgabedatei würde, durch ein komplettes Inhaltsverzeichnis am Anfang der Datei, die Übersichtlichkeit erhöht werden.

8.3 Mögliche zukünftige Ergänzung

An dieser Stelle sollen einige Möglichkeiten zukünftiger Ergänzungen aufgezählt werden. Folgende Module bzw. einzuarbeitende Verfahren wären denkbar:

- ein Modul, das den notwendigen Ruderausgleich von Nurflüglern abbilden kann
- Integration einer Möglichkeit, verschiedene Flugphasen mit unterschiedlichen Treibstoffen und ggf. unterschiedlichen Triebwerken durchführen zu können
- Einarbeitung neuer Antriebsmöglichkeiten wie z.B.: Turboprop, Raketenantriebe, elektrische Propellertriebwerke usw.
- Einarbeiten eines Verfahrens, das die Lärmentwicklung in der Passagierkabine ermittelt

Hinweis:

Um zu erfahren, wie man in das „Getriebe“ von *PrADO* eingreifen kann, um neue Module hinzuzufügen, möchte ich hier auf die sehr übersichtliche Darstellung in der Studienarbeit: „*Structure and function of the aircraft design program PrADO*“ von Linoel Salavin hinweisen.

9 Zusammenfassung

Es wurde gezeigt, wie sich das Arbeiten mit Prado vereinfachen lässt. Desweiteren wurde zusammengefasst, welche Umstände die Benutzerfreundlichkeit zurzeit noch erschweren.

Trotz einiger Veränderungsvorschläge innerhalb dieser Arbeit, ist die Komplexität, die das Programm *PrADO* bewältigen kann und der daraus erreichbare große Nutzen, nicht zu unterschätzen.

Zwar kann *PrADO* für den erfahrenen Benutzer viele unterstützende Informationen liefern, ist aber für den Nutzer, der nur gelegentlich mit diesem Programm offene Fragen klären will, im Verhältnis zum notwendigen Einarbeitungsaufwand eher ungeeignet. Neben dem anzueignenden Wissen über *PrADO* sind, für das Ausschöpfen der Möglichkeiten, auch Kenntnisse in den Programmiersprachen *FORTRAN* und *Java* notwendig.

Im Zusammenhang mit den Zielen des Projektes „Grüner Frachter“ wird dieses Vorentwurfsprogramm die ein oder andere Entscheidung herbeiführen. Es darf allerdings bei der Suche nach der zum Erreichen der Ziele benötigten Flugzeugkonfiguration nicht überbewertet werden.

10 Literaturverzeichnis

- ATR 2005** AVIONS DE TRANSPORT REGIONAL (HRSG.): *ATR 42-300/320 and ATR 72-200/210 The Regional Way to Profitability*.
URL: <http://atracraft.com> (2007-12-07)
- Becker 1988** BECKER, Hans-Jürgen: *Avions de Transport Regional –ATR 42*.
NARA-Verlag, 1988
- Heinze 1994** HEINZE, Wolfgang: *Ein Beitrag zur quantitativen Analyse der technischen und wirtschaftlichen Auslegungsgrenzen verschiedener Flugzeugkonzepte für den Transport großer Nutzlasten*. Braunschweig: Institut für Flugzeugbau und Leichtbau, Dissertation, 1994
- Heinze 2005** HEINZE, Wolfgang: *Methodisches Entwerfen von Flugzeugen*.
Braunschweig: Institut für Flugzeugbau und Leichtbau, 2005
- Heinze 2007a** HEINZE, Wolfgang: *PrADO-TA2*.
Braunschweig: Institut für Flugzeugbau und Leichtbau, 2007
- Heinze 2007b** HEINZE, Wolfgang: *Hinweise zur Installation von PrADO*.
Braunschweig: Institut für Flugzeugbau und Leichtbau, 2007
- Heinze 2007c** HEINZE, Wolfgang: *Entwurfsuntersuchung zu umweltfreundlichen und kosteneffektiven Frachtflugzeugen mit unkonventioneller Konfiguration*.
Braunschweig: Institut für Flugzeugbau und Leichtbau, 2007

-
- Horstmann 2007** HORSTMANN, Karl Heinz: *Ein Mehrfach-Traglinienverfahren für Entwurf und Nachrechnung nichtplanarer Flügelanordnungen*. Braunschweig: Institut für Flugzeugbau und Leichtbau, 2004
- JANE'S 1996** FREDERICK, Thomas (Hrsg.): *Jane's all the World's Aircraft*. Couldsen: Jane's Information Group Limited, 1996
- Kiesel 2007** KIESEL, Thorsten: *Methodisches Entwerfen von Verkehrsflugzeugen mit PrADO*. Hamburg: HAW, Fachbereich Fahrzeugtechnik und Flugzeugbau
URL: <http://bibliothek.ProfScholz.de> (2008-05-20)
- Raymer 1989** RAYMER, D. P.: *Aircraft Design: A Conceptual Approach*. AIAA Education Series, Washington: AIAA, 1989
- Salavin 2008** SALAVIN, Lionel: *Structure and function of the aircraft design program PrADO*. Hamburg: HAW, Fachbereich Fahrzeugtechnik und Flugzeugbau, 2008
- Walde 2000** WALDE, David: *Flugzeugentwurf mit dem Programm RDS*. Hamburg: HAW Hamburg, Fachbereich Fahrzeugtechnik und Flugzeugbau
URL: <http://bibliothek.ProfScholz.de> (2008-04-13)
- Westphal 2004** WESTPHAL, Robert: *Untersuchungen zur Einbindung eines Lärm- ausbreitungsmodells in den integrierten Flugzeugvorentwurf*. Braunschweig: Institut für Flugzeugbau und Leichtbau, 2004
- Wunderlich 2007** WUNDERLICH, Tobias: *Polareninterpolationsmodul für die An- bindung an das Mehrfach-Traglinienverfahren LIFTING LINEV2.2.*. Braunschweig: DLR, 2007

Anhang A: Kurzprotokoll der ATR72-200F