

# Kalibrierung der Einström-Randbedingung des ANSYS® FLUENT®-Windtunnelmodells auf Basis der Windtunnelversuchsdaten

*eine kurze Beschreibung*

*letzte Veränderung 01.10.2018*

Alexander Prinz

*Die vorliegenden Seiten erheben keinerlei Anspruch auf Existenz als wissenschaftlich vollwertige Arbeit. Auf entsprechende Attribute einer solchen wie Zitate, Inhaltsverzeichnis, Literatur- und Bildverweise etc. wurde aus Zeitgründen verzichtet. Vielmehr dienen diese Seiten als eine Beschreibung meiner Arbeit um die Kalibrierung der Einströmrand-Randbedingung, bzgl. des Windtunnelmodells. Die gesamte Arbeit - beinhaltend PEST++-PYTHON-Model-Arbeitsfluss, Ergebnisvisualisierung und jene Dokumentation - wurde nicht während meiner Arbeitszeit als HIWI am ATB gemacht, sondern resultierte aus dem großen Interesse für das Projekt und dem Wunsch Dipl. Ing. Moustapha Doumbia sowie natürlich auch allen anderen Involvierten damit einen möglichen Gefallen zu tun.*

## Einleitung

Das bekannte numerische ANSYS-FLUENT-Modell basiert auf den Daten des bekannten realen Windtunnelmodellversuchs, welcher durch ein numerisches Modell via. numerischer Simulation abgebildet werden soll. Die im Windtunnelversuch laminar einströmende Luft bildet, durch am Boden installierte Metallwinkel, eine Grenzschicht aus, so wie sie natürlicher Weise durch etwaige Topografische Objekte (Berge, Vegetation, Bauwerke etc.) induziert wird (Rauhigkeit). Dies Grenzschicht zeichnet sich durch eine von der Höhe abhängige nicht konstante Strömungsgeschwindigkeit der Luft aus (Geschwindigkeitsprofil). Um dieses Geschwindigkeitsprofil im numerischen Model zu reproduzieren, wird am Einströmrand des Modellgebietes eine Dirichlet-Randbedingung in Form einer Geschwindigkeitsprofilfunktion definiert. Diese Funktion bildet in Abhängigkeit der Modellhöhe sowie einiger weiterer Parameter die Strömungsgeschwindigkeit der modellierten Luft im Modell ab. Die bisher genutzte Funktion, welche aus einer Wissenschaftlichen Arbeit zu Grenzschichtmodellierung im meteorologischen Kontext, entnommen wurde, bildet jedoch das gemessene Geschwindigkeitsprofil aus dem Windtunnelversuch mit einem größeren Fehler ab, als es mit der in diesem Skript vorgestellten Kalibrierungsstrategie möglich ist.

Die der Funktion zugehörigen Parameter wurden in einem ersten Schritt durch eine nichtlineare Regression mit dem Programm Microsoft EXCEL angepasst. Durch Erweiterung der Funktion um einen addierten funktionalen Term in Form einer gebrochen rationalen Polynomfunktion 6. Grades, konnte der Fehler der Geschwindigkeitsprofilfunktion zu den Versuchsdaten deutlich minimiert werden. Im Folgendem wird die Methode erläutert und die Ergebnisse präsentiert.

## Methode

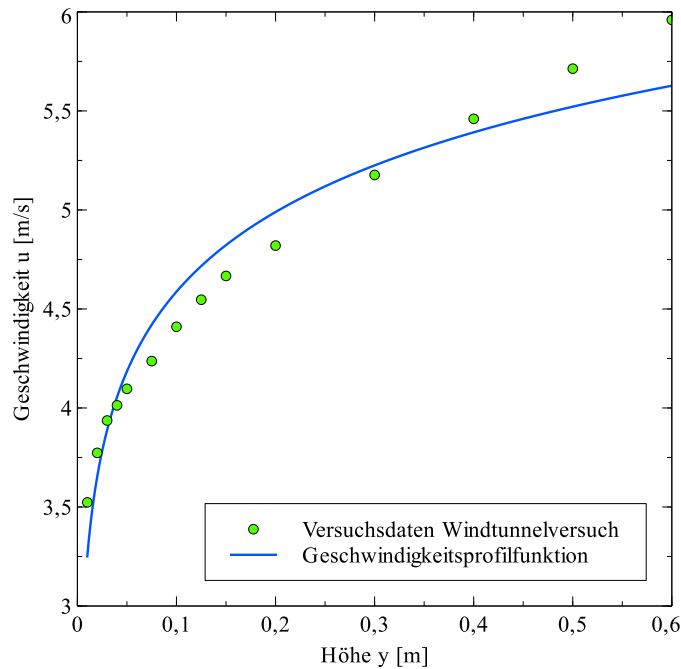


Abb. I:

Windtunnelversuchsdaten (grüne Punkte) gegen die ursprüngliche nicht kalibrierte Geschwindigkeitsprofilfunktion (blauer Graph).

$$u(y) = \frac{u_{ABL}^*}{\kappa} \ln\left(\frac{y}{y_0}\right) \quad (i)$$

$$u(y) = 0,5805 \cdot \ln(y) + 5,9238 \quad (ii)$$

Die Geschwindigkeitsprofilfunktion wurde zuvor mithilfe einer nichtlinearen Regression an die Versuchsdaten approximiert. Gleichung i stellt die Geschwindigkeitsprofilfunktion aus der erwähnten Publikation dar. Sie wurde entsprechend verändert und die Koeffizienten mittels Microsoft EXCEL geschätzt (siehe Gleichung ii). Diese Schritte wurde von Dipl. Ing. Moustapha Doumbia realisiert, und sind somit nicht direkter Bestandteil der vorgestellten Kalibrierungsmethode, bilden aber die Basis dafür. Wie in Abbildung I zu erkennen ist, bestehen zwischen den Versuchswerten und der Geschwindigkeitsprofilfunktion Differenzen, welche nicht konstant sind, sondern eine nichtlineare Abhängigkeit der Höhe annehmen lassen.

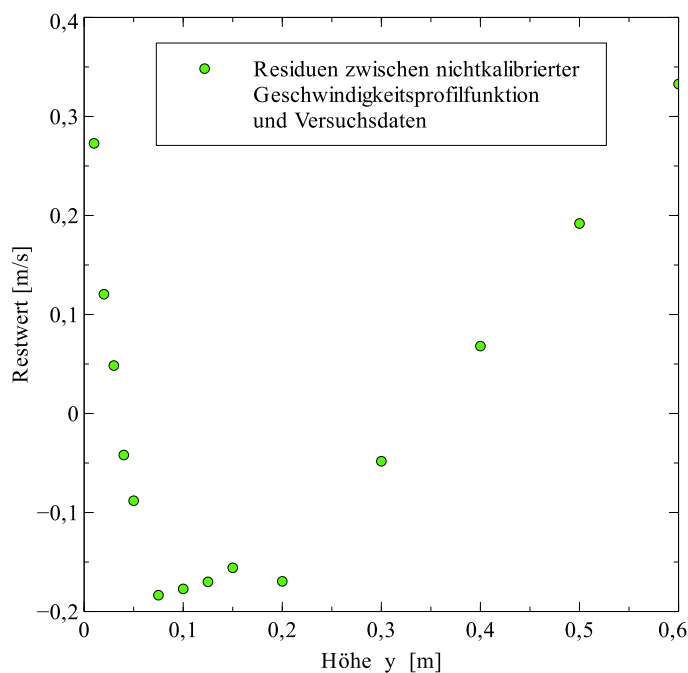


Abb. II:

Residuen zwischen den Versuchswerten des Geschwindigkeitsprofils und der Geschwindigkeitsprofilfunktion.

Das Kalibrierungsprinzip beruht nun darauf, die vorhandenen gemessenen Residuen sowie die dazwischen liegenden hypothetischen Residuen zwischen Windtunnelversuch und der Geschwindigkeitsprofilfunktion durch eine Residuenfunktion abzubilden und diese Residuenfunktion der Geschwindigkeitsprofilfunktion zu addieren. Abbildung II zeigt die Residuen zwischen den Windtunnelversuchsdaten und der Wandfunktion in Abhängigkeit der korrespondierenden Höhe.

$$r_i(y_i, m(y_i), u(y_i)) = u(y_i) - m(y_i) \quad (\text{iii})$$

$\Rightarrow$

$$\tilde{u}(y) = \tilde{r}(y, m(y_i), u(y_i)) + u(y) \quad (\text{iv})$$

mit:  $\tilde{u}$  ist ähnlich zu  $m$

$$\tilde{r}(y) = \frac{\sum_{i=0}^N a_i y^i}{\sum_{i=0}^N b_i y^i} \quad (\text{v})$$

Gleichung iii repräsentiert eine Abbildung, welche die diskreten Messhöhen  $y_i$ , die korrespondierenden Messwerte  $m$  des Windtunnelversuchs sowie die Geschwindigkeit gegeben durch die Geschwindigkeitsprofilfunktion  $u$  auf die Residuen  $r_i$  abbildet. Gleichung iv leitet sich aus Gleichung iii her, wobei die Funktion  $\tilde{r}$  eine approximierte Funktion der Residuen über alle Punkte ist, d.h. sowohl die Residuen innerhalb des Messhöhenintervalls abbilden soll (Interpolation) als auch jene Punkte, die außerhalb des Messhöhenintervalls liegen (Extrapolation). Die Funktion  $\tilde{u}$  stellt die kalibrierte Geschwindigkeitsprofilfunktion dar. Sie ergibt sich aus der funktionalen Summe der approximierten Residuenfunktion sowie der ursprünglichen Geschwindigkeitsprofilfunktion.

Um die Residuen in allen Punkten gut zu interpolieren und darüber hinaus auch zu extrapolieren, muss eine Funktion gefunden werden, welche die Residuen in allen Punkten möglichst mit geringem Fehler abbildet. Es hat sich nach einigen Fehlversuchen beispielsweise durch annähern mit rationalen Polynomfunktionen  $n$ 'ten Grades gezeigt, dass eine gebrochen rationale Polynomfunktion konkret 6. Grades als mögliche Approximation dienen kann. Gleichung v zeigt den allgemeinen Fall für eine gebrochen rationale Funktion. Setzt man für  $N$  (Polynomgrad) die Zahl 6 ein und schätzt die Koeffizienten  $a_i$  und  $b_i$  genügend genau, genügt diese Funktion als additiver funktionaler Term einer Kalibrierung der Geschwindigkeitsprofilfunktion.

Für die Schätzung der Koeffizienten wurde ein Arbeitsfluss in Form einer Kopplung zwischen dem Programm PEST++ und der gebrochen rationalen Polynomfunktion implementiert in ein PYTHON-Skript entwickelt. PEST++ (**Par**amter **Est**imation) ist ein Programm zur Lösung inverser Probleme und hat neben dem Vorteil, dass es der open source Anwendbarkeit unterliegt, mit diversen Modellen via ASCII-Textdateien mit jedem Modell koppelbar ist, sofern das Modell seine Lösungen als ASCII-Textdateien ausgeben kann.

*...Auf den genauen numerischen Prozess soll in diesem Skript nicht eingegangen werden. Auch nicht auf Entwickler und sonstige Attribute, welche sonst in wissenschaftlichen Arbeiten genannt werden. Bei Interesse an dem Programm und seinem Arbeitsfluss bin ich gerne bereit, entsprechende Erläuterungen zu geben. Andernfalls gibt es auch eine erstklassige Anleitung vom Entwickler auf der PEST-Internetseite. Die PEST++-Seite ist aus meiner Sicht hingegen weniger gut.*

Abbildung III zeigt eine grobe Beschreibung des Iterativen Prozesses bzgl. der Kopplung der Komponenten. Die Vorwärtskomponente stellt das Modell dar. Als PYTHON-Programm wurde das Modell so entwickelt, dass die zu verändernden Parameter  $a_i$  und  $b_i$  über eine externe ASCII-Textdatei (Parameterdatei) an das PYTHON-Programm übergeben werden können. PEST++ kann ebenfalls auf die Parameterdatei zugreifen und die Parametereinträge in ihr verändern. In jedem Iterationsschritt rechnet das PYTHON-Programm mit den neu angepassten Parametern und schreibt die Resultate bzw. Lösung in eine weitere ASCII-Textdatei welche ebenfalls von PEST++ eingelesen werden kann. Diese Lösungen werden dann von PEST++ evaluiert in dem PEST++ die Lösung mit den Versuchsdaten vergleicht. Hierbei können verschiedene Konvergenzkriterien bestimmt werden, wobei die „Best fit“-Option gewählt wurde, welche dem Prinzip der kleinsten Fehlerquadrate entspricht. Die Versuchsdaten erhält PEST++ durch die CONTROL-Datei. In dieser Datei werden auch diverse Parameter und Modi für den PEST++-Ablauf bestimmt. Wurde das Konvergenzkriterium im Iterativen Ablauf erfüllt, schreibt PEST++ die geschätzten Parameter sowie eine zugehörige Unsicherheitsanalyse in eine REC-Datei. Andernfalls wird ein weiterer Iterationsschritt mit neu angepassten Parametern initiiert. Ein Batch-Skript als ausführbare Datei, regelt die serielle Ansteuerung des PYTHON-Skriptes zu jedem Iterationsschritt in einer Kopplung mit PEST++. Die entsprechenden Anweisungen bzw. Pfade für die Skripte werden ebenfalls durch die CONTROL-Datei verwaltet.

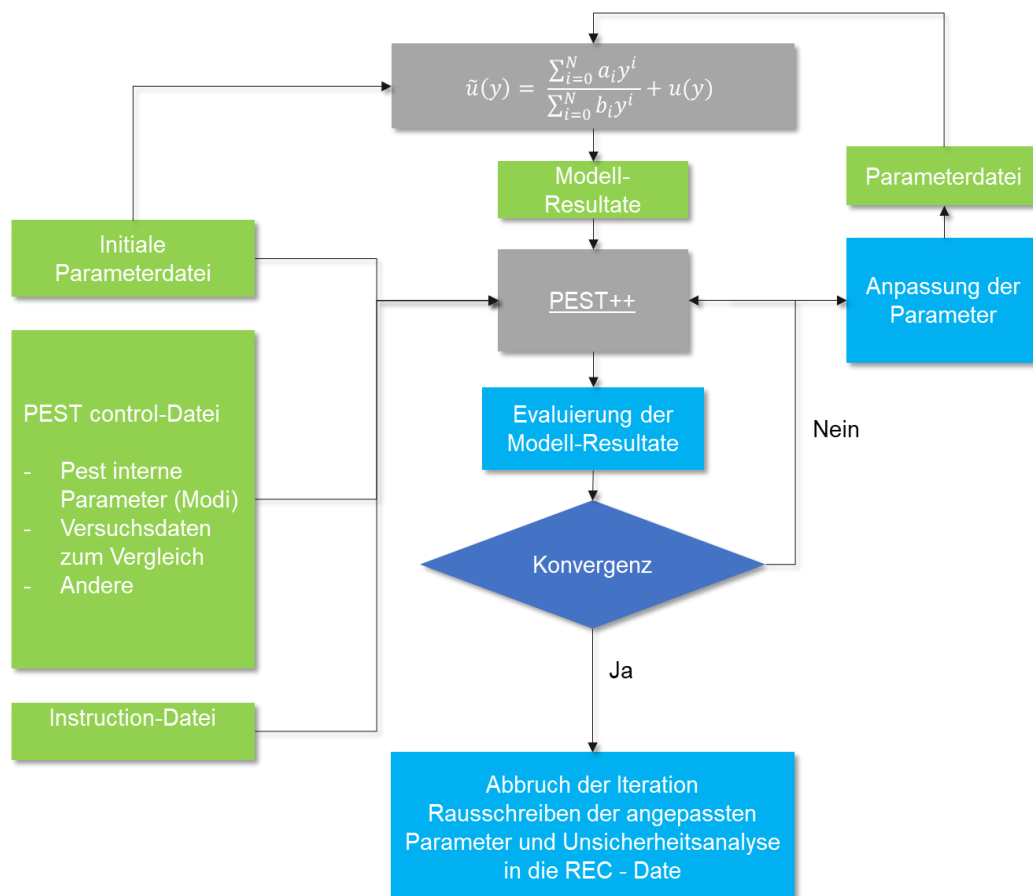
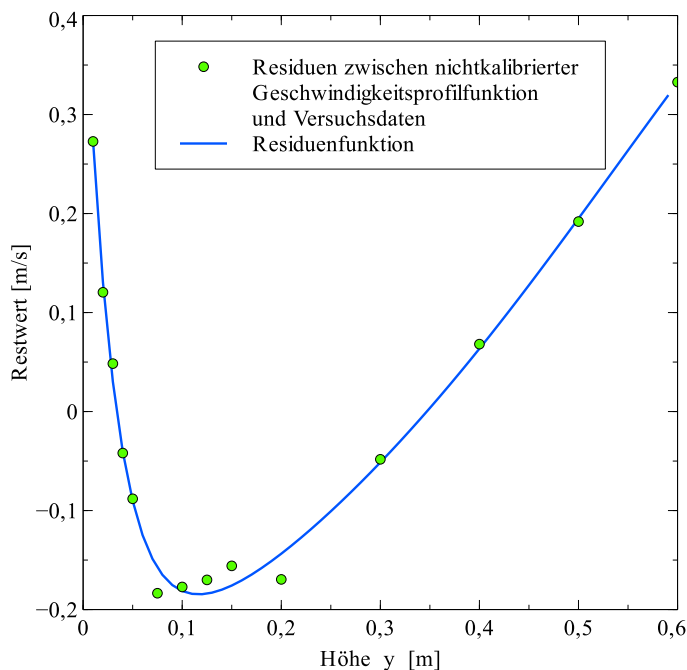


Abb. III:

PEST++ - Vorwärtsmodell – Workflow als einfaches Flussdiagramm. Es zeigt den iterativen Prozess der Kopplung beider Programme. Das Vorwärtsmodell ist in diesem konkreten Fall die um den funktionalen  $\tilde{r}$ -Term erweiterten  $u$ -Funktion (siehe Gleichungen iv und v). Der Polynomgrad wurde nicht als weiterer Parameter geschätzt, sondern „per Hand“ verändert. Es wurden also mehrere Szenarien mit jeweils unterschiedlichen aufsteigenden Polynomgrad von  $N = 3$  angefangen, gerechnet.

Nachdem die Koeffizienten sowie der optimale Polynomgrad ermittelt wurden, wurde der C-code, welcher die Geschwindigkeitsprofilfunktion als Randbedingung für das ANSYS-FLUENT-Modell repräsentiert, entsprechend um den funktionalen Term der Residuenfunktion ergänzt. Der so modifizierte C-code wurde dann im ANSYS-FLUENT-Modell implementiert und die Funktion durch einen Modelllauf validiert. Die entsprechenden Modellergebnisse wurden als CSV-Datei gespeichert und im Folgenden als Ergebnis präsentiert.

## Ergebnisse/Diskussion



**Abb. IV:**

Residuen zwischen den Versuchswerten des Geschwindigkeitsprofils und der Geschwindigkeitsprofilfunktion (grüne Punkte). Zudem wurde die Residuenfunktion geplottet (blauer Graph). Es lässt sich eine gute Approximation der Residuen zeigen. Lediglich im Messhöhenintervall [0,1m; 0,2m] zeigen sich größere Abweichungen.

Abbildung IV zeigt die approximierte Residuenfunktion gegen die Residuen zwischen Versuchsmesswerten und der Geschwindigkeitsprofilfunktion. Der Verlauf der Residuenfunktion folgt sehr gut den Residuen. Lediglich im y - Intervall [0,1;0,2] gibt es größere Abweichungen..

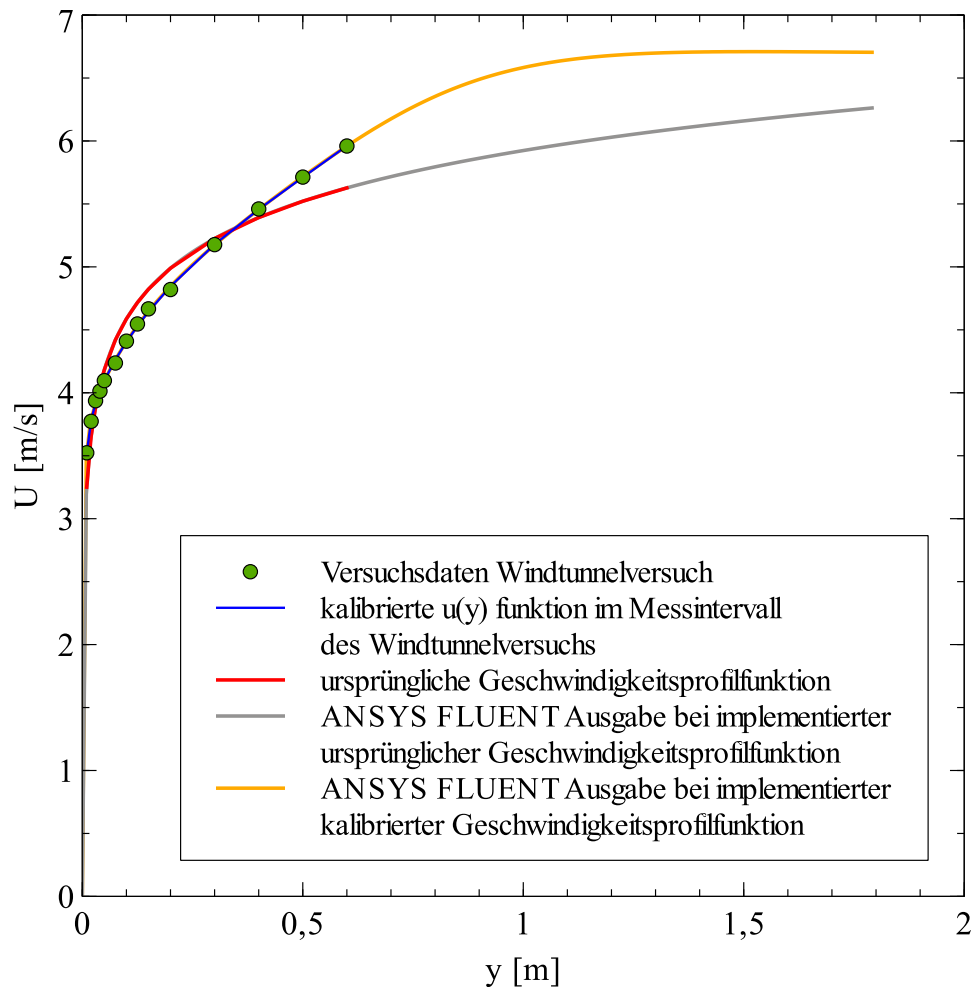


Abb. VI:

Die Abbildung zeigt die Ursprungsdaten (Versuchsdaten des Windtunnelversuchs, originale Geschwindigkeitsprofilfunktion) sowie die Resultate der Kalibrierung. Hierbei wurde nochmal Unterschieden zwischen den Ergebnissen der kalibrierten  $u(y)$ -Funktion ohne ANSYS-FLUENT Modell (blauer Graph) sowie der Ausgabe der Einströmkanalgeschwindigkeit in Abhängigkeit der gesamten Modellhöhe des ANSYS-FLUENT-Modells bei kalibrierter Geschwindigkeitsprofilfunktion (oranger Graph). Es ist ein deutlicher Unterschied zu erkennen, wobei die Versuchsdaten vom Windtunnelversuch wesentlich genauer angenähert werden.

In der Abbildung VI werden alle Ergebnisse in einem Plot dargestellt. Der rote Graph repräsentiert die ursprüngliche Geschwindigkeitsprofilfunktion vor der Kalibrierung. Der graue gepunktete Graph zeigt das Ergebnis bzw. die ANSYS-FLUENT Ausgabe des Geschwindigkeitsprofils an der Modelleinströmkannte bei implementierter ursprünglicher Geschwindigkeitsprofilfunktion. Die grünen Punkte repräsentieren die Messergebnisse aus dem Windtunnelversuch. Der Blaue Graph entspricht den Ergebnissen der kalibrierten  $u(y)$ -Funktion im Messintervall des Windtunnelversuchs. Der orange Graph zeigt das gesamte Geschwindigkeitsprofil an der Einströmkannte im ANSYS-FLUENT-Modell bei implementierter kalibrierter Geschwindigkeitsprofilfunktion. Er repräsentiert das Gesamtergebnis der Kalibrierungsmethode. Der Verlauf bildet die Messwerte des Windtunnelversuchs gut ab. Über das Messintervall hinaus, nimmt der Graph einen natürlichen transienten Verlauf in Richtung einer laminaren Strömung an.

## Fazit

Es konnte eine mögliche Eignung der Kalibrierungsmethode anhand passabler Ergebnisse gezeigt werden. Aus zeitlichen Gründen wurden bestimmte mathematische Verfeinerungen ausgelassen. So wäre etwa die genaue Untersuchung der Ergebnisqualität hinsichtlich der rein heuristisch bestimmten Polynomgrade der Residuenfunktion ein interessanter Punkt. Zudem wurde nicht untersucht ob weitere mögliche funktionale Terme existieren um eine noch genauere Fehlerreduzierung zu erreichen. Die in Abbildung II erkennbaren Anomalien im  $y$ -Intervall  $[0,1; 0,2]$  - möchte man sie denn so nennen - scheinen einen physikalischen Grund zu haben. Möglicherweise handelt es sich um Messanomalien bzw. Instationaritäten (sehr feine Turbulenzen etc.) hinsichtlich des Strömungsfeldes im Windtunnelversuch an den entsprechenden Messstellen.

Ferner sei zu erwähnen, dass die Residuenfunktion nur anhand der Messdaten im existierenden Messintervall approximiert werden konnte, die Modelldimensionierung jedoch über dieses Intervall hinausgeht. Inwieweit weitere Messpunkte Signifikant wären bzw. die Qualität der Residuenfunktion verbessern würden, bleibt ungeklärt.

Der „mathematisch regelmäßig“ anmutende Verlauf der Residuen (siehe Abbildungen II und IV) lässt auf einen systematischen Fehler/Grund des Systems Messversuch-Geschwindigkeitsprofilfunktion schließen. Es bleibt ungeklärt, ob dieser physikalischer Natur-, oder aber numerischer Natur entspringt.

### *Persönlicher Zusatz:*

*Es ist mir immer eine gewisse Freude, Wissen, welches ich mir in meinen Studienjahren, aneignen durfte, anwenden zu können. Inverse Probleme bzw. Kalibrierungsprobleme sind dabei, nicht zu guter Letzt auch durch meine Bachelorarbeit, zu einem gewissen Interesse geworden. Falls meine Arbeit hier einen kleinen positiven Anklang findet, würde mich das sehr freuen.*

*Kurznach dem man als noch unbedeutender Studierender seine erste wissenschaftliche Arbeit verfasst hat, verfällt man wohl allzu schnell bei entsprechenden Gelegenheiten (wie dieser) in einen Hang des übertrieben aufgeplusterten Schreibens, etwa so wie der Drang eines Kindes, neu Erlerntes – was auch immer das sein mag - in jeder nur erdenklichen Gelegenheit oft etwas übertrieben zu zeigen, nutzen, anzuwenden...ich hoffe das war hier nicht all zu sehr der Fall. Möglicher Weise, sehe ich mein „kleines Kalibrierprojekt“ hier auch euphorisch etwas zu ernst als es das tatsächlich ist. In diesem Fall bitte ich um Entschuldigung des möglicherweise anmaßenden Erscheinens.*

## Zusätzliches

Alle Ergebnisse, technischen Schritte (codes etc.) und sonstige dieser Arbeit zugehörigen Materialien, wurden gespeichert. Sollten diese für irgendwelche Zwecke als nützlich dienen und eine Aushändigung dieser Materialien gewünscht sein, so ist dies selbstverständlich kein Problem. Bei entsprechender Mitteilung, werde ich diese gern aushändigen.