

# Simulation des LDA-Meßprozesses in mehrdimensionalen Strömungen

## Kleiner Beleg

(korrigierte Abschrift)

cand. ing. Holger Nobach  
geb. am 22.07.69 in Rostock

# U n i v e r s i t ä t   R o s t o c k

Fachbereich Elektrotechnik, Albert-Einstein-Straße 2, O-2500 Rostock 6  
Fachgebiet Automatisierungsgeräte und -anlagen, Leiter Prof. Dr. sc. techn. P. Wildfang

---

Rostock, den 1. Nov. 1991

## Kleiner Beleg

für Herrn cand. ing. Holger N o b a c h

Thema:        Simulation des LDA-Meßprozesses in mehrdimensionalen  
                 Strömungen

Aufbauend auf die bestehende 1D-Simulation sind

- Grundlagen der Simulation mehrdimensionaler Strömungen mit autoregressiven Funktionen,
- Algorithmen für die Simulation des LDA-Meßprozesses in 2D/3D-Strömungen und
- eine parametrische Realisierung in Turbo-PASCAL

zu erarbeiten.

Plausibilität und Effektivität des erstellten Simulationsprogramms sind anhand ausgewählter Beispiele zu belegen.

Begin:            18.11.1991

Abgabetermin:   15.03.1992

Betreuer         Dr. Fuchs

Prof. Dr.-Ing. habil. Wildfang  
Fachgebietsleiter

Die Anwendung der Laser-Doppler-Anemometrie (LDA) als berührungsloses optisches Geschwindigkeitsmeßverfahren bedarf einer speziellen Anpassung an das konkrete Meßproblem.

Auf Grundlage der Beschreibung der Meßwertgewinnung als stochastische Abtastung ist die Informationsverdichtung des Meßwertensembles zu Zeitmittelwerten an a-priori Informationen des Strömungszustandes am Meßort und Geräteparameter gebunden.

Da hierfür eine Vielzahl von Eingangsparametern von Bedeutung sind, ist es durch Experimente nur unter erheblichem technischen Aufwand möglich, gezielt dem Einfluß einzelner Parameter zu untersuchen.

Hier bietet die Rechnersimulation eine kosten- und zeitgünstige Alternative. Durch die Rechnersimulation lassen sich Informationen über die Zuverlässigkeit der Ergebnisse einer realen Messung bei bekannten Eingangsparametern ableiten. Des weiteren können durch gezielte Veränderung einstellbarer Parameter durch die Simulation vorab Auswirkungen auf die Informationsgewinnung in der realen Messung festgestellt werden.

# Inhaltsverzeichnis

Symbolverzeichnis	v
<b>1 Einleitung</b>	<b>1</b>
<b>2 Konzept der Simulation</b>	<b>3</b>
<b>3 Algorithmen zur Simulation</b>	<b>10</b>
3.1 Grundserie eindimensional turbulenter Strömungen . . . . .	10
3.2 Teilchenserie eindimensional turbulenter Strömungen . . . . .	14
3.3 Prozessorserie eindimensional turbulenter Strömungen . . . . .	20
3.4 Wichtung eindimensional turbulenter Strömungen . . . . .	34
3.5 Grundserie dreidimensional turbulenter Strömungen . . . . .	37
3.6 Teilchenserie dreidimensional turbulenter Strömungen . . . . .	41
<b>4 Umsetzung in eine Programmiersprache</b>	<b>47</b>
4.1 Auswahl der Programmiersprache . . . . .	47
4.2 Programmoberfläche . . . . .	49
4.3 Einzugebende Werte . . . . .	51
<b>5 Ausgewählte Ergebnisse</b>	<b>54</b>
<b>6 Zusammenfassung und Ausblick</b>	<b>55</b>
Literatur	56
Erklärung	58
<b>A1 Beweis für die exponentielle Dichtefunktion der Teilchenabstände</b>	<b>59</b>
<b>A2 Testprogramme</b>	<b>65</b>
A2.1 VTEST unter MS-Fortran . . . . .	65
A2.2 VTEST unter Turbo-Pascal . . . . .	66
A2.3 AR unter MS-Fortran . . . . .	67
A2.4 AR unter Turbo-Pascal . . . . .	75

## Symbolverzeichnis

$a$	Halbachse des Meßvolumenellipsoids in $x$ -Richtung
$a_t$	zufälliger Anteil der Abweichung der Geschwindigkeit zum Zeitpunkt $t$ vom Mittelwert
$a'$	Halbachse der Projektionsellipse des Meßvolumens in Strömungsrichtung
$b$	Halbachse des Meßvolumenellipsoids in $y$ -Richtung
$b'$	Halbachse der Projektionsellipse des Meßvolumens in Strömungsrichtung
$c$	Halbachse des Meßvolumenellipsoids in $z$ -Richtung
$c_0$	konstanter Anteil der Teilchendichte
$c_H$	obere Teilchendichte
$c_L$	untere Teilchendichte
$c_V$	Teilchendichte
$d_m$	Durchmesser der Laserstrahlen
$f_D$	Dopplerfrequenz
$f_n$	Anzahl der Geschwindigkeitssamples pro integralem Zeitmaß
$f_r$	Reabtastrfrequenz
$f_s$	Shiftfrequenz
$i$	Laufvariable
$j$	Laufvariable
$m_c$	Teilchendichteanstieg
$m_i$	Anfangsmoment $i$ -ter Ordnung
$n$	Anzahl der Perioden
$n_{\text{erw}}$	zu erwartende Teilchenzahl während $T$
$n_{\text{res}}$	Periodenzahl zum Rücksetzen des Prozessors
$n_{\text{set}}$	Periodenzahl zum Setzen des Prozessors
$r$	gleichverteilte Zufallszahl
$r_\epsilon$	exponentialverteilte Zufallszahl
$r_n$	normalverteilte Zufallszahl
$r_t$	bezogener Weg
$\vec{r}$	Teilchenbahn
$\vec{r}_0$	Durchstoßpunkt der Teilchenbahn durch das $\xi$ - $\eta$ -Koordinatensystem
$s$	Weg
$t$	Zeit
$t_0$	Ankunftszeit des Teilchens

$t_{0\xi\eta}$	Durchstoßzeit des Teilchens durch die Normalenebene zur Strömungsrichtung
$t_a$	Anfangszeit des Burstes
$t'_a$	Differenz der Anfangszeit des Burstes zur Ankunftszeit des Teilchens
$t_d$	Differenz der Ankunftszeiten des letzten und des aktuellen Teilchens
$t_e$	Endzeit des Burstes
$t'_e$	Differenz der Endzeit des Burstes zur Ankunftszeit des Teilchens
$t_f$	Zeit der nächsten Prozessorfreigabe
$t_l$	Zeit, zu der der Zwischenspeicher leer ist
$t_p$	Zeit des Prozessorsignals
$t_{rdel}$	Rücksetzdelaytime
$t_r$	Reabtastzeit
$t_s$	Speicherzeit
$t_{üdel}$	Übertragungsdelaytime
$v_c$	Bezugswert für Momentenbestimmung
$v_s$	Geschwindigkeit, bei dem Teilchendichtesprung auftritt
$v_t$	Geschwindigkeit zum Zeitpunkt $t$
$v_x$	$x$ -Komponente der Geschwindigkeit
$v'$	Differenz von $v$ zum Mittelwert $\bar{v}$
ws	Speichertiefe
ws <sub>max</sub>	maximale Speichertiefe
$x$	Achse in Meßrichtung
$y$	Achse senkrecht zur Meßrichtung und zur optischen Achse
$z$	optische Achse
$z_t$	Abweichung der Geschwindigkeit zum Zeitpunkt $t$ vom Mittelwert
$A_{\perp x}$	Projektionsfläche des Meßvolumens in $x$ -Richtung
$A_{\perp}(\vec{v})$	Projektionsfläche des Meßvolumens in Richtung von $\vec{v}$
$D^2X$	Varianz der Strömungsgeschwindigkeit
$D^2X_{Ens}$	Varianz des Meßwertensembles
$D^2X_{GS}$	Varianz der Geschwindigkeitsgrundserie
$D^2X_{PS}$	Varianz der Prozessorserie
$D^2X_{TS}$	Varianz der Teilchenserie
EX	Erwartungswert der Strömungsgeschwindigkeit

$EX_{\text{Ens}}$	Erwartungswert des Meßwertensembles
$EX_{\text{GS}}$	Erwartungswert der Geschwindigkeitsgrundserie
$EX_{\text{PS}}$	Erwartungswert der Prozessorserie
$EX_{\text{TS}}$	Erwartungswert der Teilchenserie
$I$	Intensität des elektromagnetischen Feldes
$I_0$	Mittenintensität eines einfallenden Laserstrahl
$M_i(v_c)$	Moment $i$ -ter Ordnung bezüglich $v_c$
$N$	Anzahl der Samples
$\dot{N}_0, S_0$	Meßvolumenrate
$\dot{N}_1, S_1$	Triggerrate
$\dot{N}_2, S_2$	Meßrate
$\dot{N}_3, S_3$	Datenspeicherrate
$T$	Meßzeit
$V$	Volumen des Meßvolumens
$\beta_1$	Bias 1. Ordnung
$\beta_2$	Bias 2. Ordnung
$\beta_2^*(v_c)$	Bias 2. Ordnung bezüglich $v_c$
$\phi$	skalärer Koeffizient
$\phi_{ij}$	Komponente $i, j$ der Matrix $\Phi$
$\gamma_0$	Wichtungsfaktor
$\gamma_{ij}$	Kovarianz der Komponenten $i$ und $j$ des Vektors $\vec{v}$
$\eta$	Achse in der Normalenebene zur Anströmrichtung
$\psi_{ij}$	Komponente $i, j$ der Matrix $\Psi$
$\lambda$	Wellenlänge
$\mu_i$	Zentralmoment $i$ -ter Ordnung
$\nu$	Drehwinkel der Projektionsellipse des Meßvolumens in Strömungsrichtung gegenüber dem $\xi$ - $\eta$ -Koordinatensystem
$\rho_t$	normierter Wert der Autokorrelationsfunktion bei $\tau = t$
$\sigma_z$	Standardabweichung des Skalars $z$
$\sigma_z^2$	Varianz des Skalars $z$
$\tau$	Zeit
$\tau_m$	Aufenthaltszeit des Teilchens im Meßvolumen
$\varphi$	Breitenkoordinate im Kugelkoordinatensystem
$\varphi_z(\tau)$	Autokorrelationsfunktion der Größe $z$
$\xi$	Achse in der Normalenebene zur Anströmrichtung
$\vartheta$	skalares integrales Zeitmaß
$\zeta$	Längenkoordinate im Kugelkoordinatensystem

$\Delta cv$	Teilchendichtehub
$\Delta t$	Abstand der Samples
$\Delta x$	Interferenzstreifenabstand
$\Phi$	Koeffizientenmatrix
$\Gamma_z$	Kovarianzenmatrix des Vektors $\vec{z}$
$\Psi$	Koeffizientenmatrix
$\Lambda$	Parameter
$\Sigma_z$	Matrix der Standardabweichungen des Vektors $\vec{z}$
$\Theta$	integrale Zeitmaßmatrix
$\Theta_0$	Schnittwinkel der Laserstrahlen

allgemeine Symbole:

$\bar{g}$	Mittelwert der Größe $g$
$\tilde{g}$	zufällige Größe $g$
$\langle g \rangle$	durch Zufallsfunktion erzeugte Größe $g$



# 1 Einleitung

Die wesentlichen Eigenschaften der Laser-Doppler-Anemometrie als Strömungsgeschwindigkeitsmeßverfahren

- berührungsloses, indirektes optisches Verfahren,
- absoluter, linearer Zusammenhang zwischen der Strömungsgeschwindigkeit und der Signalfrequenz als elektrischer Meßgröße,
- geometrisch definierte Richtungsabhängigkeit im Raum,
- hohe räumliche und zeitliche Auflösung und
- Erweiterbarkeit zur simultanen Mehrkomponentenmessung

führten zu einem verbreiteten Einsatz in der Strömungsmechanik.

Schon bald nach der Erstanwendung zeigte sich, daß der Einsatz der Laser-Doppler-Anemometrie einer speziellen Anpassung an das konkrete Meßproblem bedarf.

Die Informationsverdichtung des Meßwertensembles im LDA-Burst-Betrieb zu Zeitmittelwerten ist an a-priori Informationen des Strömungszustandes am Meßort und Geräteparameter gebunden.

Zur Untersuchung des Einflusses einzelner Parameter wird der LDA-Meßprozeß durch mathematische Modelle beschrieben. Diese können direkt, mathematisch determiniert, verarbeitet werden oder Grundlage einer Rechnersimulation sein.

In der vorliegenden Arbeit sind mathematische Modelle Ausgangspunkt für die Simulation des LDA-Meßprozesses.

Hierzu wurden umfangreiche Untersuchungen zu Grundlagen der Simulation mehrdimensionaler Strömungen mit autoregressiven Funktionen durchgeführt.

Daraus wurden Algorithmen für die Simulation des LDA-Meßprozesses in dreidimensionalen Strömungen abgeleitet.

Die parametrische Realisierung durch eine Programmiersprache beschränkt sich bisher auf die eindimensional turbulenten Strömungen mit inhomogenen Teilchendichten.

Die erarbeiteten Algorithmen für die Simulation in mehrdimensionalen Strömungen bilden die Grundlage für eine Umsetzung in einem Programm.

Im ersten Abschnitt (Konzept der Simulation) wird der LDA-Meßprozeß untersucht, mathematische Modelle abgeleitet und das Konzept der Simulation dargestellt.

Im zweiten Abschnitt (Algorithmen zur Simulation) werden ausgehend vom LDA-Meßprozeß für jeden Schritt der Simulation die verwendeten Modelle in einen formelmäßigen Zusammenhang gebracht und daraus Algorithmen für die Simulation hergeleitet.

Im dritten Abschnitt (Umsetzung in eine Programmiersprache) wird die Realisierung der Algorithmen in einem Programm beschrieben.

## 2 Konzept der Simulation

Zur Bestimmung der Strömungsgeschwindigkeit von Gasen und Fluiden wird neben anderen Verfahren die Laser-Doppler-Anemometrie (LDA) verwendet. Der wesentliche Vorteil liegt hierbei in der berührungslosen Messung, auch über Entfernungen von mehreren Metern hinweg.

Die Laser-Doppler-Anemometrie beruht darauf, daß kleine Teilchen im Gas oder Fluid mit einem vernachlässigbaren Schlupf vom strömenden Medium mitgeführt werden und diese eingestrahlte elektromagnetische Wellen streuen.

Das LDA-Meßvolumen entsteht im Schittpunkt der sich unter dem Winkel  $\Theta_0$  kreuzenden Laserstrahlen (Abb. 2.1). Nimmt man an, daß es sich um zwei identische Laserstrahlen mit Gaußscher Intensitätsverteilung handelt und daß im Kreuzungspunkt ebene Wellenfronten vorliegen, dann entsteht in einem kartesischen Koordinatensystem  $x, y, z$  die räumliche Intensitätsverteilung [1]

$$I = I_0 \left\{ e^{\frac{-8}{d_m^2} \left[ \left( x \cos \frac{\Theta_0}{2} + z \sin \frac{\Theta_0}{2} \right)^2 + y^2 \right]} + e^{\frac{-8}{d_m^2} \left[ \left( x \cos \frac{\Theta_0}{2} - z \sin \frac{\Theta_0}{2} \right)^2 + y^2 \right]} + 2e^{\frac{-8}{d_m^2} \left( x^2 \cos^2 \frac{\Theta_0}{2} + z^2 \sin^2 \frac{\Theta_0}{2} + y^2 \right)} \cos \left( \frac{4\pi}{\lambda} x \sin \frac{\Theta_0}{2} \right) \right\} \quad (2.1)$$

mit  $d_m$  als Durchmesser zwischen dem  $e^{-2}$ -Abfall der Intensität der einfallenden Laserstrahlen. Die beiden ersten Summanden beinhalten den Gleichanteil der Intensität, der dritte Summand stellt die Intensität des Interferenzfeldes dar. Die dabei auftretenden Interferenzflächen liegen parallel zur  $y$ - $z$ -Ebene und haben den Abstand

$$\Delta x = \frac{\lambda}{2 \sin \frac{\Theta_0}{2}}, \quad (2.2)$$

d. h., Wechelsignale werden nur durch die  $x$ -Komponente der Teilchengeschwindigkeit gebildet.

Somit durchquert jedes Teilchen, das entlang der  $x$ -Achse das Meßvolumen passiert, den in Abb. 2.2 dargestellten Intensitätsverlauf.

Als Grenzen des Meßvolumens wird der Ort des  $e^{-2}$ -Abfalls der Maximalintensität des Wechselanteils angenommen. Die Meßvolumengrenzfläche ist dann ein Ellipsoid (Abb. 2.3) mit den Halbachsen

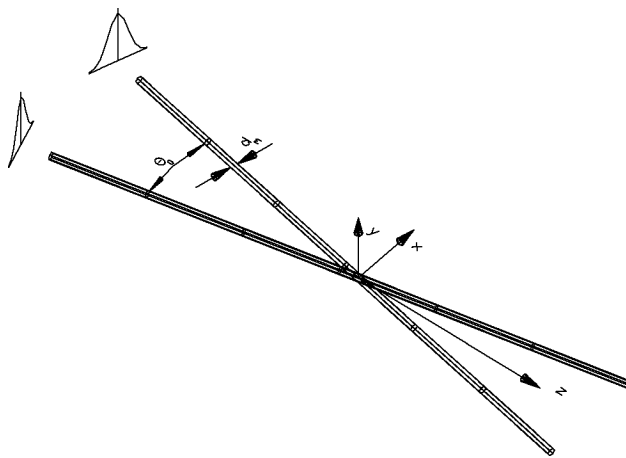


Abbildung 2.1: Definition des Koordinatensystems

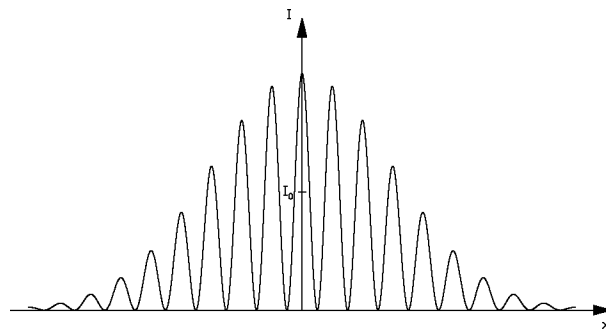


Abbildung 2.2: Intensitätsverlauf entlang der  $x$ -Achse

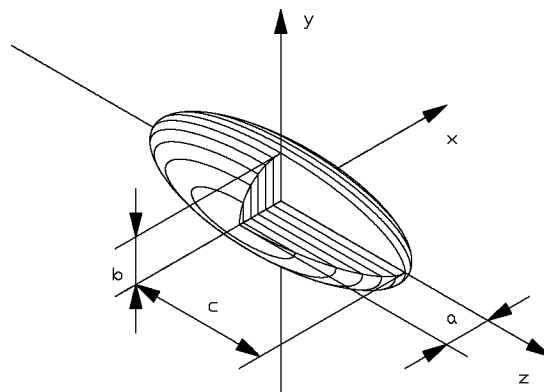


Abbildung 2.3: LDA-Meßvolumen

$$a = \frac{d_m}{2 \cos \frac{\Theta_0}{2}}, \quad b = \frac{d_m}{2}, \quad c = \frac{d_m}{2 \sin \frac{\Theta_0}{2}} \quad (2.3)$$

und dem Volumen

$$V = \frac{4\pi abc}{3}. \quad (2.4)$$

Passiert ein Streuteilchen das Meßvolumen, so streut es das einfallende Licht. Auf einem Fotoempfänger bildet sich, durch die Wechselwirkung des Teilchens mit dem räumlichen elektromagnetischen Feld, der räumliche Intensitätsverlauf zeitlich ab. Der Fotoempfänger wandelt den zeitlichen Intensitätsverlauf in ein elektrisches Meßsignal um.

Die Frequenz des Wechselanteils des elektrischen Meßsignals  $f_D$  und der Abstand der Interferenzstreifen im Meßvolumen  $\Delta x$  stehen durch

$$f_D = \frac{v_x}{\Delta x} \quad (2.5)$$

in Beziehung. Dabei ist  $v_x$  die  $x$ -Komponente der Teilchengeschwindigkeit. Damit kann durch Messung der Frequenz des Wechselanteils des elektrischen Signals auf die  $x$ -Komponente der Teilchengeschwindigkeit durch

$$v_x = \Delta x f_D \quad (2.6)$$

geschlossen werden.

Ausgehend von der Voraussetzung, daß die Teilchen der Strömung schlupfflos folgen, kann auf die  $x$ -Komponente der momentanen Strömungsgeschwindigkeit geschlossen werden, die dann gleich der Teilchengeschwindigkeit ist.

Somit ist die Informationsgewinnung über die momentane Strömungsgeschwindigkeit nur dann möglich, wenn ein Streuteilchen das Meßvolumen passiert. Die Geschwindigkeitsbestimmung unterliegt somit der statistischen Verteilung der Teilchen im Strömungsvolumen.

Ausgehend von im Fluidregellosen, integriert über das Strömungsvolumen zeitunabhängig mit der Konzentration  $c_V$  gleichverteilter Streuteilchen ist die momentane Anzahl von Teilchen  $N(t)$  in einem definierten Volumen statistisch mit der Poisson-Verteilung beschreibbar [2].

$$p(N(t) = n) = \frac{\overline{N(t)}^n}{n!} e^{-\overline{N(t)}} \quad (2.7)$$

Die mittlere Teilchenzahl  $\overline{N(t)}$  ergibt sich für das LDA-Meßvolumen aus der Teilchenkonzentration  $c_V$  aus

$$\overline{N(t)} = \frac{4c_V\pi abc}{3}. \quad (2.8)$$

Für  $\overline{N(t)} < 0,1$  ist  $p(N(t) > 1) = 4,7 \cdot 10^{-3}$  [3]. Damit stammt ein Meßsignal mit Sicherheit von genau einem Streuteilchen, die Geschwindigkeitsbestimmung ist aber wegen  $p(N(t) = 1) < 9\%$  nicht zu jedem Zeitpunkt möglich. Die Beschreibung der Häufigkeit des Auftretens der Meßsignale erfolgt durch die in [4] eingeführten Anzahlen pro Zeiteinheit (Raten). Ohne den Einfluß der Signalauswertetechnik zu berücksichtigen, ergibt sich aus der Teilchenzahl im pro Zeiteinheit vom LDA-Meßvolumen erfaßten Fluidvolumen die Meßvolumenrate

$$\dot{N}_0(\vec{v}(t)) = S_0(\vec{v}(t)) = c_V |\vec{v}(t)| A_{\perp}(\vec{v}(t)). \quad (2.9)$$

$A_{\perp}(\vec{v}(t))$ : Projektionsfläche des Meßvolumens in Strömungsrichtung  
 $\vec{v}(t)$ : Vektor der momentanen Strömungsgeschwindigkeit [3]

Nimmt man an, die Triggerschwelle der Auswerteelektronik befindet sich genau auf dem  $e^{-2}$ -Abfall der Maximalintensität, und die Teilchen haben konstante Streueigenschaften, so leisten alle Teilchen einen Beitrag zur Triggerrate  $\dot{N}_1(v(t)) = S_1(v(t))$  [4], deren Bahn das Meßvolumen wenigstens tangiert, und die Triggerrate ist gleich der Meßvolumenrate.

Da die Geschwindigkeitsbestimmung im LDA-Burst-Betrieb eine Abtastung darstellt, deren Meßhäufigkeit beschrieben durch die Signalrate sowohl vom Betrag der momentanen Strömungsgeschwindigkeit als auch von der Anströmrichtung abhängt, entstehen bei turbulenten Strömungen Differenzen zwischen Mittelwert und Varianz des Ensembles der Geschwindigkeitsmeßwerte und des kontinuierlichen zeitgemittelten Geschwindigkeitsverlaufes des strömenden Mediums (Bias).

Faßt man die Geschwindigkeit  $v_x$  als Zufallsgröße  $X$  auf, ist der Bias darstellbar als

$$\beta_1 = \frac{EX_{\text{Ens}} - EX}{EX} \quad (2.10)$$

$$\beta_2 = \frac{D^2X_{\text{Ens}} - D^2X}{D^2X} \quad (2.11)$$

Ist in (2.9) die Teilchendichte  $c_V$  selbst eine Funktion der momentanen Strömungsgeschwindigkeit (inhomogene Teilchendichte), so leistet dies ebenfalls einen Beitrag zum Bias.

Da durch den Meßprozeß nur das Meßwertensemble bereitgestellt wird, müssen die Meßwerte zur Bestimmung der zeitgemittelten Kenngrößen mit Hilfe von Wichtungsverfahren bewertet werden. Hierzu gehören folgende Methoden:

- Zeitdifferenzwichtung [5]/[6]:

$$\gamma_{0i} = \frac{1}{t_{0i} - t_{0i-1}}$$

- Geschwindigkeitswichtung [7]:

$$\gamma_{0i} = v_{xi}$$

- Aufenthaltszeitwichtung [8]:

$$\gamma_{0i} = \frac{1}{\tau_{mi}}$$

- Wichtung nach Nakoa [9]:

$$\gamma_{0i} = \left( 2 - \frac{v_{xi}}{\text{EX}_{\text{Ens}}(v_x)} \right)^{\pm \frac{3}{2}}$$

Für den gewichteten Mittelwert und die gewichtete Varianz gilt dann [3]

$$\overline{v_x} = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{v_{xi}}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (2.12)$$

$$\overline{v_x'^2} = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{(v_{xi} - \overline{v_x})^2}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (2.13)$$

Zur Untersuchung des Auftretens und der Größe des Bias sowie der Wirksamkeit dieser Wichtungsverfahren in Abhängigkeit vom Turbulenzgrad der Strömung, der Besähung des strömenden Mediums mit Streuteilchen und der Auswertelektronik, wird der LDA-Meßprozeß auf einem Rechner simuliert. Hierzu wird die Simulation des Gesamtprozesses wie folgt unterteilt:

1. Simulation der Strömung (durch Zufallsfunktion erzeugt)
2. Simulation der Streuteilchenbesähung (durch Zufallsfunktion erzeugt; unter Berücksichtigung von (2.9))
3. Simulation der Signalauswertung (Bewertung der Signale aus 2.)
4. Informationsverarbeitung des Meßwertensembles

Für die gezielte Untersuchung des Einflusses einzelner Parameter ist die Realisierung der einzelnen Teilschritte der Simulation durch getrennte Programmschritte nötig, wobei die Ausgabedaten eines Teilschrittes durch den nächsten verarbeitet werden. Dadurch ist einerseits eine realistische Nachbildung des LDA-Meßprozesses gewährleistet, andererseits die Untersuchung des Einflusses einzelner Parameter in jedem Teilschritt auf identische Eingangsdaten möglich.

Die Realisierung einer Simulation durch ein Computer-Programm basiert auf der numerischen Beschreibung des zu simulierenden Vorgangs. Hierbei können kontinuierliche Größen nur zeitdiskret in einem endlichen Untersuchungszeitraum und mit endlicher Genauigkeit nachgebildet werden.

Bei der Simulation der Strömung wird daher die kontinuierliche Größe Geschwindigkeit durch zeitdiskrete Samples beschrieben. Der kontinuierliche Verlauf geht dann durch Approximation aus der zeitdiskreten Funktion hervor.

Die einzelnen Teilschritte der Simulation des LDA-Meßprozesses erzeugen daher Zahlenreihen (Serien) mit folgenden Inhalten:

1. Die Simulation der Strömung beinhaltet die Erzeugung einer zeitdiskreten Geschwindigkeits-Zeit-Funktion mit konstanter Abtastfrequenz (Grundserie).
2. Die Simulation der Teilchenbesähung beinhaltet die Erzeugung einer Geschwindigkeits-Zeit-Funktion (Teilchenserie), wobei die Samples zu



den Zeitpunkten eines, sich aus der Simulation ergebenden, Teilchendurchtrittes durch das Meßvolumen auftreten. Es entsteht also eine Geschwindigkeits-Zeit-Funktion mit aperiodischer Abtastung. Hierbei wird zu jedem angenommenen Teilchendurchtritt in einem separaten Zufallsprozeß die Aufenthaltszeit des Teilchens im Meßvolumen berechnet.

3. Bei der Simulation der Auswertelektronik werden die Verfahren der Meßwertgewinnung aus dem eingegangenen Signal nachgebildet. Hierbei werden die Meßsignale aus 2. bewertet. Es entsteht wiederum eine aperiodisch abgetastete Geschwindigkeits-Zeit-Funktion (Prozessorserie).
4. Zur Beurteilung der Wirksamkeit der einzelnen Wichtungsverfahren werden die aus 3. erhaltenen Werte verarbeitet. Die Ergebnisse werden Mittelwert und Varianz der Grundserie und der Teilchenserie gegenübergestellt.

## 3 Algorithmen zur Simulation

### 3.1 Grundserie eindimensional turbulenter Strömungen

Zur Generierung der Grundserie eindimensional turbulenter Strömungsgeschwindigkeit wird angenommen, daß sich der Momentanwert der Strömungsgeschwindigkeit aus der Summe von einem Mittelwert und einer zufälligen normalverteilten Abweichung zusammensetzt.

Die zeitdiskreten Samples  $\tilde{v}_{xt}$  der Strömungsgeschwindigkeit ergeben sich dann aus

$$\tilde{v}_{xt} = \overline{v_x} + \tilde{z}_t, \quad (3.1.1)$$

wobei  $\overline{v_x}$  die vorgegebene mittlere Geschwindigkeit der zu simulierenden Strömung ist und  $\tilde{z}_t$  die zeitdiskreten zufälligen Abweichungen.

Zur Ermittlung des zufälligen Anteils  $\tilde{z}_t$  wird das autoregressive Modell verwendet, welches bereits von Tropea zur Untersuchung des Leistungsdichtespektrums erfolgreich eingesetzt wurde. [12]

Zur Berechnung der zeitdiskreten Samples eines autoregressiven Prozesses  $p$ -ter Ordnung wird die Rekursion

$$\tilde{z}_t = \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + \dots + \phi_p \tilde{z}_{t-p} + \tilde{a}_t \quad (3.1.2)$$

aus [10] verwendet mit vorgegebenen Konstanten  $\phi_i$ . Die Konstanten  $\phi_i$  geben an, wie stark die Abweichung zum Zeitpunkt  $t$  von den vorangegangenen Abweichungen abhängt, oder wie stark sie sich von einem Sample zum nächsten verändern kann.

Um die zeitlichen Abstände der Samples beliebig festzulegen, wird (3.1.2) als

$$\tilde{z}_t = \phi_1 \tilde{z}_{t-1\Delta t} + \phi_2 \tilde{z}_{t-2\Delta t} + \dots + \phi_p \tilde{z}_{t-p\Delta t} + \tilde{a}_t \quad (3.1.3)$$

geschrieben.

Zur Kennzeichnung der zeitlichen Abhängigkeit der zufälligen Abweichungen  $\tilde{z}_t$  zum Zeitpunkt  $t$  von den vorhergehenden wird das integrale Zeitmaß verwendet. Für sichere statistische Aussagen, ist es notwendig, die Anzahl der Samples pro integralem Zeitmaß  $f_n$  in  $\frac{1}{\Delta t}$  als Parameter zu wählen. Die zeitlichen Abstände der Samples betragen dann  $\Delta t = \frac{1}{f_n}$ . Das integrale Zeitmaß selbst ergibt sich dann aus

$$\vartheta = \frac{\sum_{i=0}^{\infty} \varphi_z\left(\frac{i}{f_n}\right)}{f_n \varphi_z(0)} = \frac{1}{f_n} \sum_{i=0}^{\infty} \frac{\varphi_z\left(\frac{i}{f_n}\right)}{\varphi_z(0)}. \quad (3.1.4)$$

wobei  $\varphi_z(\tau)$  die Autokorrelationsfunktion von  $z(t)$  ist.

Die im folgenden geschilderten Beziehungen zwischen den Zufallsgrößen sind Erwartungsgrößen, die jedoch von den konkreten, simulierten, Größen abweichen können, da diese nur einen zeitlichen Ausschnitt der Strömung widerspiegeln. Die aus den durch die Simulation erzeugten Werten errechneten Größen werden stets mit  $\langle \rangle$  gekennzeichnet. Für sehr lange Simulationsreihen ( $N \rightarrow \infty$ ) nähern sich die konkreten Größen dann den erwarteten.

Für die Autokorrelationsfunktion  $\varphi_z(\tau)$  gilt nach [10]:

$$\frac{\varphi_z(\tau)}{\varphi_z(0)} = \rho_\tau \quad (3.1.5)$$

wobei

$$\begin{aligned} \rho_\tau &= \rho_{\perp\tau} \\ \rho_0 &= 1 \\ \rho_\tau &= \phi_1 \rho_{\tau \perp \Delta t} + \phi_2 \rho_{\tau \perp 2\Delta t} + \dots + \phi_p \rho_{\tau \perp p\Delta t}. \end{aligned} \quad (3.1.6)$$

Das integrale Zeitmaß ergibt sich demnach aus

$$\vartheta = \frac{1}{f_n} (\rho_0 + \rho_{\Delta t} + \rho_{2\Delta t} + \dots). \quad (3.1.7)$$

Sind die voneinander unabhängigen zufälligen Abweichungen  $\tilde{a}_t$  wie in [12] normalverteilt, mit dem Erwartungswert 0, so sind auch die zufälligen Abweichungen der Strömungsgeschwindigkeit  $\tilde{z}_t$  vom vorgegebenen Mittelwert normalverteilt mit dem Erwartungswert 0, was dem betrachteten Modell entspricht.

Nach [10] besteht zwischen der Varianz  $\sigma_a^2$  der Zufallsgröße  $\tilde{a}_t$  und der Varianz  $\sigma_z^2$  der Abweichung  $\tilde{z}_t$  der Geschwindigkeitswerte von ihrem Mittelwert folgender Zusammenhang:

$$\sigma_z^2 = \frac{\sigma_a^2}{1 - \rho_{\Delta t} \phi_1 - \rho_{2\Delta t} \phi_2 - \dots - \rho_{p\Delta t} \phi_p} \quad (3.1.8)$$

Da die Geschwindigkeitswerte  $\tilde{v}_{xt}$  durch lineare Verschiebung aus  $\tilde{z}_t$  hervorgehen (vgl. (3.1.1)), sind die Varianzen beider Größen gleich.

Zur Berechnung der normalverteilten  $\tilde{a}_t$  mit der Varianz  $\sigma_a^2$  werden zunächst normalverteilte Zufallszahlen  $\tilde{r}_{n_t}$  mit dem Erwartungswert 0 und der Varianz 1 nach [11] durch

$$\tilde{r}_{n_t} = \sqrt{-2 \ln(1 - \tilde{r}_{1_t})} \cos(2\pi \tilde{r}_{2_t}) \quad (3.1.9)$$

erzeugt, wobei die  $\tilde{r}_{1_t}$  und  $\tilde{r}_{2_t}$  unabhängig voneinander und gleichverteilt aus  $[0, 1)$  sind, und somit die  $\tilde{r}_{n_t}$  normalverteilt und voneinander unabhängig sind.

Der zufällige Anteil der Abweichung der momentanen Strömungsgeschwindigkeit  $\tilde{a}_t$  in der Rekursion (3.1.2) errechnet sich dann aus

$$\tilde{a}_t = \sigma_a^2 \tilde{r}_{n_t}. \quad (3.1.10)$$

In dem vorliegenden Programm wurde der autoregressive Prozeß erster Ordnung realisiert. Demnach ergeben sich aus (3.1.1) und (1.1.3)

$$\langle v_{xt} \rangle = \overline{v_x} + \langle z_t \rangle \quad (3.1.11)$$

mit

$$\langle z_t \rangle = \phi_1 \langle z_{t-1} \rangle + \langle a_t \rangle. \quad (3.1.12)$$

Aus (3.1.5) ergeben sich

$$\rho_{\Delta t} = \phi_1. \quad (3.1.13)$$

Somit gilt für die erwartete Varianz der zufälligen Abweichungen der momentanen Strömungsgeschwindigkeit von ihrem Mittelwert

$$\sigma_z^2 = \frac{\sigma_a^2}{1 - \phi_1^2} \quad (3.1.14)$$

Für vorgegebene Varianz der zufälligen Abweichung  $\sigma_z^2$  der Strömungsgeschwindigkeit ergibt sich die erforderliche Varianz  $\sigma_a^2$  des zufälligen Anteil aus

$$\sigma_a^2 = \sigma_z^2 (1 - \phi_1^2). \quad (3.1.15)$$

Das integrale Zeitmaß ergibt sich als Summe der Reihe (3.1.7):

$$\vartheta = \frac{1}{f_n(1 - \phi_1)}. \quad (3.1.16)$$

Sind  $f_n$  und  $\vartheta$  gegeben, so ergibt sich daraus  $\phi_1$  nach

$$\phi_1 = 1 - \frac{1}{\vartheta f_n}. \quad (3.1.17)$$

Sind die vorgegebene mittlere Geschwindigkeit  $\overline{v_x}$ , deren Standardabweichung  $\sigma_{v_x} = \sigma_z$  sowie das integrale Zeitmaß  $\vartheta$  und die Anzahl der Samples pro integralem Zeitmaß  $f_n$  gegeben, so kann nach (3.1.17) der Koeffizient  $\phi_1$  und nach (3.1.15) die erforderliche Varianz  $\sigma_a^2$  des zufälligen Anteils berechnet werden.

Um Geschwindigkeitswerte zu erhalten, die unabhängig von ihren Startwerten sind, wird die Rekursion (3.1.12) für die zehnfache Zeit des integralen Zeitmaßes durchgeführt, ohne die Werte auszuwerten.

Anschließend werden die durch diese Rekursionen zusammen mit (3.1.11) berechneten Geschwindigkeitswerte gespeichert. Für den tatsächlichen Mittelwert und die Varianz der so generierten Samples der Strömungsgeschwindigkeit gelten

$$\overline{\langle v_x \rangle} = \frac{1}{N} \sum_{i=1}^N \langle v_x \rangle_i \quad (3.1.18)$$

$$\begin{aligned} (\langle \sigma_{v_x} \rangle)^2 &= \frac{1}{N-1} \sum_{i=1}^N \left( \langle v_x \rangle_i - \overline{\langle v_x \rangle} \right)^2 \\ &= \frac{1}{N-1} \sum_{i=1}^N \langle v_x \rangle_i^2 - \frac{1}{N(N-1)} \left( \sum_{i=1}^N \langle v_x \rangle_i \right)^2 \end{aligned} \quad (3.1.19)$$

## 3.2 Teilchenserie eindimensional turbulenter Strömungen

Die über eine vorgegebene Zeit zeitdiskret bestimmte Geschwindigkeitsgrundserie wird nach einer Zufallsfunktion mit Meßereignissen versehen. Es wird angenommen, daß die Teilchen im Gesamtvolumen mit der Dichte  $c_V$  gleichverteilt sind, d. h. die Koordinaten der Teilchen und somit insbesondere die  $x$ -Koordinaten (Strömungsrichtung) der Teilchen sind bei konstantem Querschnitt  $A_{\perp x}$  parallel zur  $y$ - $z$ -Ebene gleichverteilt mit der Dichte  $A_{\perp x}c_V$ . Da das Volumen, das während der Gesamtdauer für die Aufnahme eines Meßwertensembles durch das Meßvolumen fließt, sehr viel kleiner ist als das Gesamtvolumen, ist die Anzahl der Teilchen im untersuchten Strömungsvolumen poissonverteilt. Daraus folgt, daß die Abstände in Strömungsrichtung eines Teilchens zum nächsten exponentialverteilt sind (s. Anhang 1).

Für die momentane Meßvolumenrate  $S_0(t)$  gilt

$$S_0(t) = c_V A_{\perp x} |v(t)| \quad (3.2.1)$$

Hierbei ist  $c_V$  die räumliche Teilchendichte im strömenden Medium,  $v(t)$  der Momentanwert der Strömungsgeschwindigkeit und  $A_{\perp x}$  die Projektionsfläche des Meßvolumens in  $x$ -Richtung. Für die bisher untersuchten homogenen Teilchendichten wurde angenommen, daß  $c_V$  keine Funktion der Geschwindigkeit ist. Für die hier zu untersuchende inhomogene Teilchendichte erweitert sich (3.2.1) zu

$$S_0(t) = c_V(v(t)) A_{\perp x} |v(t)| \quad (3.2.2)$$

Da die Strömung eindimensional ist, ist die Projektionsfläche des Meßvolumens konstant mit

$$A_{\perp x} = \pi b c \quad (3.2.3)$$

wobei  $b$  und  $c$  die Halbachsen des Meßvolumenellipsoids in  $y$ - und  $z$ -Richtung sind.

Daher wird im vorliegenden Programm nicht mit der räumlichen Teilchendichte  $c_V(v(t))$  sondern mit der Liniendichte  $A_{\perp x}c_V(v(t))$  gerechnet.

Die momentane Signtrate hängt sowohl von der momentanen Geschwindigkeit als auch von der dazugehörigen momentanen Teilchendichte ab. Daher gilt für den zeitlichen Mittelwert  $\bar{S}_0$  der Signtrate i. allg.  $\bar{S}_0 \neq \overline{c_V(v) A_{\perp x} |v|} \neq c_V(\bar{v}) A_{\perp x} |\bar{v}|$ .

Laut Definition des Mittelwertes ergibt sich

$$\bar{S}_0 = \frac{1}{T} \int_0^T S_0(t) dt = \frac{A_{\perp x}}{T} \int_0^T c_V(v(t)) |v(t)| dt. \quad (3.2.4)$$

Für die Abhängigkeit der Teilchendichte werden im vorliegenden Programm folgende Funktionen realisiert:

1. konstante Teilchendichte:

$$c_V(v(t)) = c_0 \quad (3.2.5)$$

2. linear vom Betrag der Geschwindigkeit abhängige Teilchendichte:

$$c_V(v(t)) = c_0 + m_c |v(t)| \quad (3.2.6)$$

3. Teilchendichte ändert sich bei  $|v| = v_s$  sprunghaft:

$$c_V(v(t)) = \begin{cases} c_L & \text{für } |v(t)| < v_s \\ c_H & \text{für } |v(t)| > v_s \end{cases} \quad (3.2.7)$$

Da die konstante Teilchendichte ein Sonderfall der linearen Abhängigkeit vom Betrag der Geschwindigkeit ist, nämlich mit  $m_c = 0$ , werden nur die beiden letzten Funktionen untersucht:

Für den Fall der linearen Abhängigkeit der Teilchendichte vom Betrag der Geschwindigkeit ergibt sich die mittlere Meßvolumenrate nach (3.2.4) zu

$$\begin{aligned} \bar{S}_0 &= \frac{A_{\perp x}}{T} \int_0^T (c_0 + m_c |v(t)|) |v(t)| dt \\ &= \frac{A_{\perp x}}{T} \int_0^T c_0 |v(t)| dt + \frac{A_{\perp x}}{T} \int_0^T m_c |v(t)|^2 dt \\ &= \frac{A_{\perp x} c_0}{T} \int_0^T |v(t)| dt + \frac{A_{\perp x} m_c}{T} \int_0^T |v(t)|^2 dt \end{aligned} \quad (3.2.8)$$

Sind  $\bar{S}_0$  und  $m_c$  gegeben so folgt:

$$c_0 = \frac{\bar{S}_0 - \frac{A_{\perp x} m_c}{T} \int_0^T |v(t)|^2 dt}{\frac{A_{\perp x}}{T} \int_0^T |v(t)| dt} \quad (3.2.9)$$

Für den Fall des Sprunges der Teilchendichte bei einer bestimmten Geschwindigkeit werden zusätzlich die Geschwindigkeit  $v_s$  bei der der Sprung auftritt, und entweder die obere oder die untere Teilchendichte oder der Teilchendichtehub  $\Delta c_V$  als gegeben vorausgesetzt. Die mittlere Meßvolumenrate  $\bar{S}_0$  ergibt sich dann aus (3.2.3) zu

$$\begin{aligned}
\bar{S}_0 &= \frac{A_{\perp x}}{T} \int_0^T c_V(v(t)) |v(t)| dt \\
&= \frac{A_{\perp x}}{T} \int_0^T c_L |v(t)| dt \Big|_{|v| < v_s} + \frac{A_{\perp x}}{T} \int_0^T c_H |v(t)| dt \Big|_{|v| > v_s} \\
&= \frac{A_{\perp x} c_L}{T} \int_0^T |v(t)| dt \Big|_{|v| < v_s} + \frac{A_{\perp x} c_H}{T} \int_0^T |v(t)| dt \Big|_{|v| > v_s} \quad (3.2.10)
\end{aligned}$$

Sind die mittlere Meßvolumenrate  $\bar{S}_0$  und die untere Teilchendichte  $c_L$  gegeben, so ergibt sich die obere Teilchendichte  $c_H$  zu

$$c_H = \frac{\bar{S}_0 - \frac{A_{\perp x} c_L}{T} \int_0^T |v(t)| dt \Big|_{|v| < v_s}}{\frac{A_{\perp x}}{T} \int_0^T |v(t)| dt \Big|_{|v| > v_s}} \quad (3.2.11)$$

Sind die mittlere Meßvolumenrate  $\bar{S}_0$  und die obere Teilchendichte  $c_H$  gegeben, so ergibt sich die untere Teilchendichte  $c_L$  zu

$$c_L = \frac{\bar{S}_0 - \frac{A_{\perp x} c_H}{T} \int_0^T |v(t)| dt \Big|_{|v| > v_s}}{\frac{A_{\perp x}}{T} \int_0^T |v(t)| dt \Big|_{|v| < v_s}} \quad (3.2.12)$$

Ist der Teilchendichtehub  $\Delta c_V$  gegeben, so lassen sich die obere und untere Teilchendichte wie folgt berechnen:

$$c_L = \frac{\bar{S}_0 - \frac{A_{\perp x} \Delta c_V}{T} \int_0^T |v(t)| dt \Big|_{|v| > v_s}}{\frac{A_{\perp x}}{T} \int_0^T |v(t)| dt} \quad (3.2.13)$$



$$c_H = \frac{\bar{S}_0 + \frac{A_{\perp x} \Delta c_V}{T} \int_0^T |v(t)| dt \Big|_{|v| < v_s}}{\frac{A_{\perp x}}{T} \int_0^T |v(t)| dt} \quad (3.2.14)$$

Da die mittlere Meßvolumenrate  $\bar{S}_0$  auf das integrale Zeitmaß  $\vartheta$  bezogen wird, ergibt sich die zu erwartende Gesamtzahl der Teilchen im untersuchten Zeitraum  $T$  aus

$$n_{\text{erw}} = \bar{S}_0 \frac{T}{\vartheta} \quad (3.2.15)$$

und der mittlere Abstand der Teilchen aus

$$\overline{\Delta s} = \frac{\int_0^T |v(t)| dt}{n_{\text{erw}}} = \frac{\int_0^T |v(t)| dt}{\frac{A_{\perp x}}{\vartheta} \int_0^T c_V(v(t)) |v(t)| dt} \quad (3.2.16)$$

Der Abstand von einem Teilchen zum nächsten ergibt sich aus

$$\Delta s_i = \overline{\Delta s} \tilde{r}_{e_i} \quad (3.2.17)$$

wobei die  $\tilde{r}_{e_i}$  exponentiell verteilte Zufallszahlen mit dem Erwartungswert 1 sind.

Diese werden nach [2] durch

$$\tilde{r}_{e_i} = -\ln(1 - \tilde{r}_i) \quad (3.2.18)$$

erzeugt mit gleichverteilten Zufallszahlen  $\tilde{r}_i$  aus  $[0; 1)$ .

In der Zeit  $\Delta t$  von einem Geschwindigkeitssample der Geschwindigkeitsgrundserie zum nächsten wird das Strömungsvolumen um

$$\Delta s_{t_j} = \int_{t_j}^{t_j + \Delta t} |v(t)| dt = \Delta t |v(t_j)| \quad (3.2.19)$$

bewegt.

Die Abstände  $\Delta s_i$  und  $\Delta s_{t_j}$  werden bei der Generierung der Teilchenserie auf  $\overline{\Delta s}$  bezogen und miteinander verglichen.

Für die Berechnung des nächsten Teilchendurchtritts durch das Meßvolumen wird zunächst der bezogene Abstand  $\tilde{r}_{e_i}$  nach (3.2.18) berechnet. Dann wird der bezogene Weg des Strömungsvolumens bis zum nächsten Sample berechnet:

$$r_{t_j} = \frac{\Delta t |v(t_j)|}{\Delta s} \quad (3.2.20)$$

Diese werden miteinander verglichen:

Ist  $\tilde{r}_{e_i} > r_{t_j}$ , so ergibt sich im nächsten Schritt  $\tilde{r}_e$  aus  $\tilde{r}_{e_{i+1}} = \tilde{r}_{e_i} - r_{t_j}$  und  $r_t$  wird für den nächsten Geschwindigkeitssample der Grundserie berechnet.

Ist  $\tilde{r}_{e_i} < r_{t_j}$ , so wird im betrachteten Zeitintervall  $\Delta t$  ein Teilchen das Meßvolumen passieren. Die Ankunftszeit des Teilchens errechnet sich nach

$$t_0 = t_j + \frac{r_{t_j}}{\tilde{r}_{e_i}} \Delta t \quad (3.2.21)$$

Im nächsten Schritt wird dann  $\tilde{r}_e$  wieder nach (3.2.18) berechnet und  $r_t$  ergibt sich aus

$$\begin{aligned} r_{t_{j+1}} &= \frac{\Delta t \left(1 - \frac{r_{t_j}}{\tilde{r}_{e_i}}\right) |v(t_j)|}{\Delta s} \\ &= r_{t_j} - \frac{\Delta t \frac{r_{t_j}}{\tilde{r}_{e_i}} |v(t_j)|}{\Delta s} \end{aligned} \quad (3.2.22)$$

Zur Berechnung der Aufenthaltszeit wird angenommen, daß die Teilchen gleichverteilt über die Projektionsfläche  $A_{\perp x}$  des Meßvolumens in Strömungsrichtung durch dieses hindurchtreten. Hierzu müssen also über einer elliptischen Fläche gleichverteilte Zufallszahlen erzeugt werden.

In [2] wird ein Verfahren vorgeschlagen, bei dem gleichverteilte Punkte in einem Parallelogramm erzeugt werden, und zwar so oft, bis der Punkt in der Projektionsfläche des Meßvolumens enthalten ist.

Die Grenzen des Meßvolumens werden hierzu als Ort des  $e^{\pm 2}$ -Abfalls der Maximalintensität angenommen. Somit ergibt sich die Grenzfläche des Meßvolumens als das Ellipsoid

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \quad (3.2.23)$$

mit den Halbachsen  $a$ ,  $b$  und  $c$  (vgl. Konzept der Simulation). Als Strömungsrichtung wird  $x$  angenommen.

Somit ergeben sich die  $y$ - und die  $z$ -Koordinate eines durch das Meßvolumen hindurchtretenden Teilchens aus

$$\begin{aligned}\tilde{y} &= b\tilde{r}_y \\ \tilde{z} &= c\tilde{r}_z\end{aligned}\tag{3.2.24}$$

mit gleichverteilten Zufallszahlen  $\tilde{r}_y$  und  $\tilde{r}_z$  aus  $(-1; 1)$ .

Diese Berechnung wird so oft wiederholt, bis ein so erzeugter Punkt in der Projektionsfläche des Meßvolumens liegt. Dann sind die so erzeugten Punkte über der Projektionsfläche gleichverteilt.

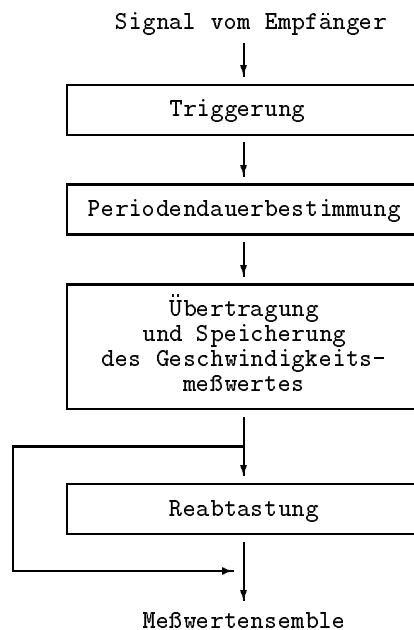
Die Aufenthaltszeit  $\tau_m$  des Teilchens im Meßvolumen ergibt sich dann aus

$$\begin{aligned}\tau_m &= \frac{2a\sqrt{1 - \frac{\tilde{y}^2}{b^2} - \frac{\tilde{z}^2}{c^2}}}{|v(t_j)|} \\ &= \frac{2a\sqrt{1 - \tilde{r}_y^2 - \tilde{r}_z^2}}{|v(t_j)|}\end{aligned}\tag{3.2.25}$$

Die Aufenthaltszeit ist bei angenommener eindimensionaler Strömung nicht von der Ausdehnung des Meßvolumens in  $y$ - und  $z$ -Richtung abhängig sondern nur in  $x$ -Richtung.

### 3.3 Prozessorserie eindimensional turbulenter Strömungen

Die von der Empfangselektronik kommenden Meßsignale werden zur Meßwertgewinnung und -speicherung, sowie zur Vorbereitung einer anschließenden Informationsverarbeitung durch die Auswertelektronik in vier Schritten verarbeitet.



#### 1. Trigger

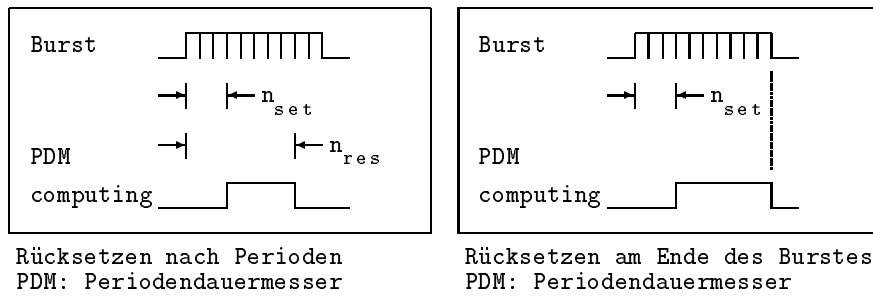
Zur Detektierung eines Teilchendurchtritts durch das Meßvolumen wird das von der Empfangselektronik kommende Meßsignal durch einen Amplitudentrigger bewertet. Überschreitet das Signal die Triggerschwelle, so handelt es sich notwendig, aber nicht hinreichend, um einen Teilchendurchtritt. Das Signal wird dann zu einer weiteren Brauchbarkeitsanalyse und zur Geschwindigkeitsmeßwertgewinnung an eine Auswerteinrichtung weitergeleitet.

#### 2. Periodendauerbestimmung

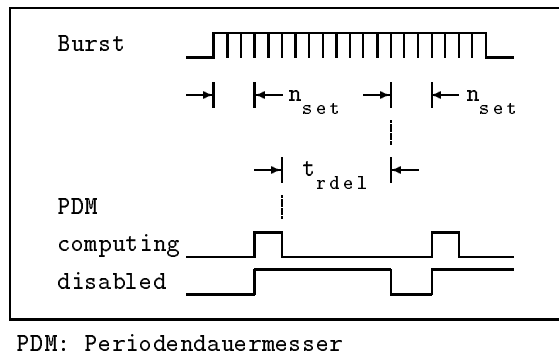
Der vom Trigger validierte Burst hat in der Frequenz seines Wechselsignals die Information über die Geschwindigkeit des detektierten Teilchens.

Diese Information kann im Frequenz- oder Zeitbereich gewonnen werden. Im Zeitbereich wird die Periodendauer durch Zählung eines Grundtaktes zwischen den Meßsignalnulldurchgängen bestimmt (Counterprinzip).

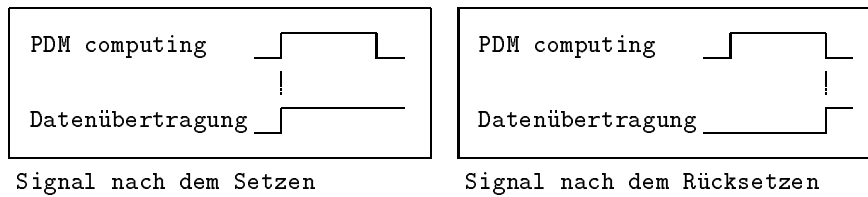
Der Counter zählt über eine bestimmte Anzahl  $n_{set}$  Signalperioden und bestimmt aus der Periodendauer die Geschwindigkeit des Teilchens. Der Counter wird dann entweder nach einer bestimmten Anzahl Perioden  $n_{res} > n_{set}$  oder am Ende des Bursts wieder zurückgesetzt.



Nach einer Totzeit  $t_{rdel}$  (Rücksetzdelaytime) ist der Counter dann für eine weitere Messung freigegeben.

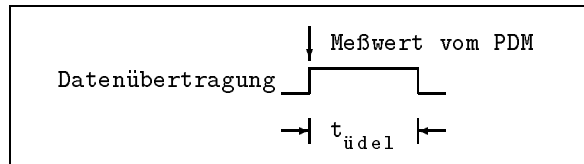


Wurde die Geschwindigkeit des detektierten Teilchens bestimmt, so wird die Datenübertragung an einen Rechner eingeleitet (entweder nach dem Setzen oder nach dem Zurücksetzen), der den Meßwert abspeichert.

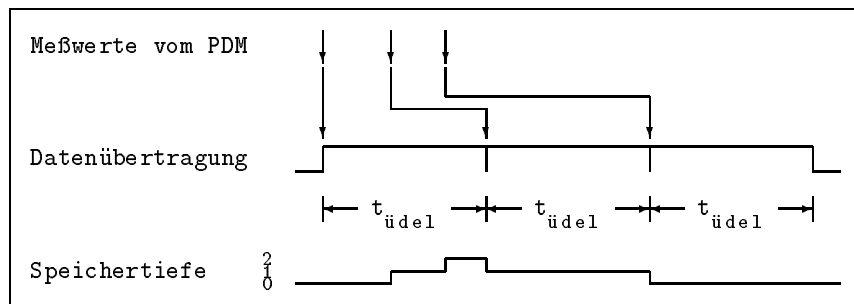


### 3. Datenübertragung/Speicherung

Die vom Counter bei vorliegendem Meßwert eingeleitete Datenübertragung und Speicherung im Rechner benötigt eine bestimmte Zeit  $t_{\text{üdel}}$  (Übertragungsdelatime), in der ein weiterer Meßwert nicht übertragen werden kann.

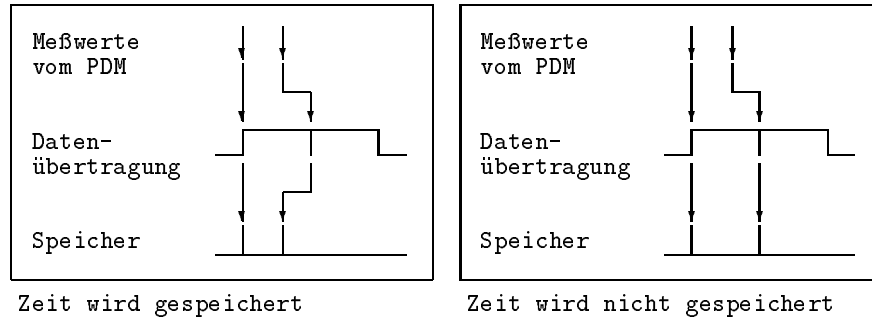


Werden die Meßwerte in einem Puffer zwischengespeichert, so kann nach dem Ende der Übertragung eines Meßwertes der nächste im Puffer zwischengespeicherte Wert übertragen werden. Dadurch können während der Übertragung eines Meßwertes weitere aufgenommen werden und nacheinander übertragen werden.



Da durch die Zwischenspeicherung der Zeitpunkt der Meßwertermittlung nicht mit dem des Auslösens der Datenübertragung übereinstimmt, kommt es, wenn der Zeitpunkt der Meßwertermittlung nicht an den Rechner weitergeleitet wird, zu einer Verschiebung der Speicherzeitpunkte gegenüber denen

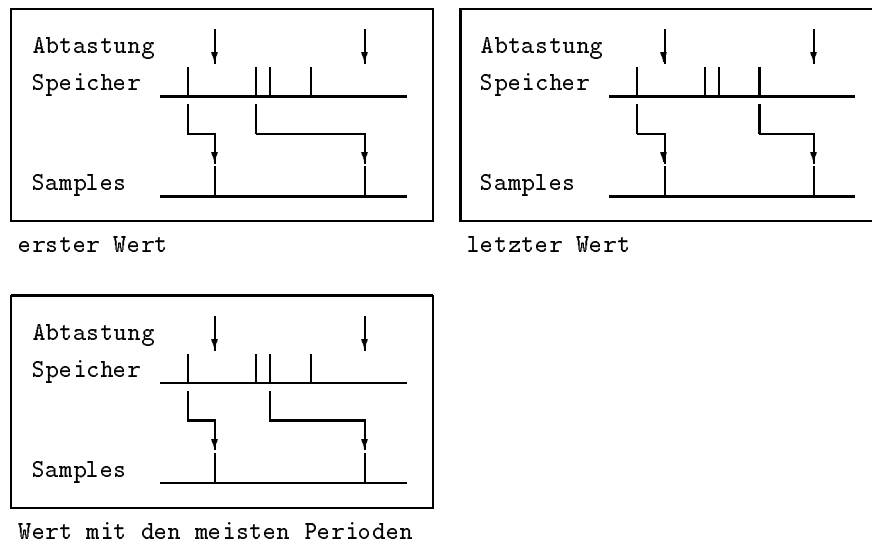
der Meßwertermittlung.



#### 4. Abtastung

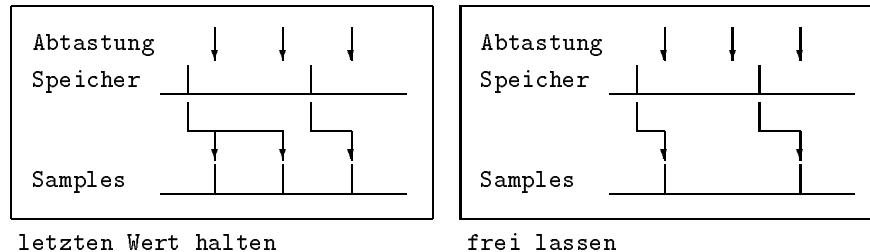
Das im Rechner abgelegte Meßwertensemble kann für die weitere Verarbeitung mit konstanter Frequenz reabgetastet werden.

Befinden sich innerhalb einer Abtastperiode mehr als ein Meßwert, so wird genau einer ausgewählt und als Abtastwert gespeichert. Welcher Burst ausgewählt wird, ist durch den Rechenablauf determiniert (erster oder letzter Burst oder Burst mit den meisten Perioden).



Befindet sich innerhalb einer Abtastperiode kein Meßwert, so wird entweder der letzte Abtastwert nochmals eingetragen, oder kein Meßwert übergeben,

was identisch mit der zusätzlichen Information, daß kein Meßwert vorliegt, ist.



Durch dieses Modell ist es möglich, die Arbeit von getakteten Prozessoren durch Einbeziehung des PDM und der Datenübertragung und -speicherung zu simulieren, evtl. mit späterer Abtastung.

Um einen getakteten Prozessor zu simulieren, bei dem die Datenübertragung zu definierten Zeiten erfolgt, und somit ein Datenverlust nicht durch die Totzeit der Übertragung und Speicherung, sondern durch die Reabtastung selbst erfolgt, ist hierzu das geschilderte Modell unter Umgehung der Totzeiten der Datenübertragung und -speicherung ebenfalls nutzbar.

Da in jedem zu bearbeitenden Schritt, mit Ausnahme der Triggerung, die nur eine Bewertung der eingehenden Signale vornimmt, Totzeiten einzubeziehen sind, bei der Simulation aber nicht die Zustände der einzelnen Verarbeitungsschritte kontinuierlich über der Zeit parallel vorliegen, sondern stets die Zeiten von Ereignissen (Auslösung, Ende u. s. w.), werden für jeden Verarbeitungsschritt die Zeit des nächsten Ereignisses und alle damit verbundenen Werte gespeichert. Diese Werte werden in jedem Verarbeitungsschritt für die Einschätzung seines eigenen Zustandes als interne Werte abgespeichert.

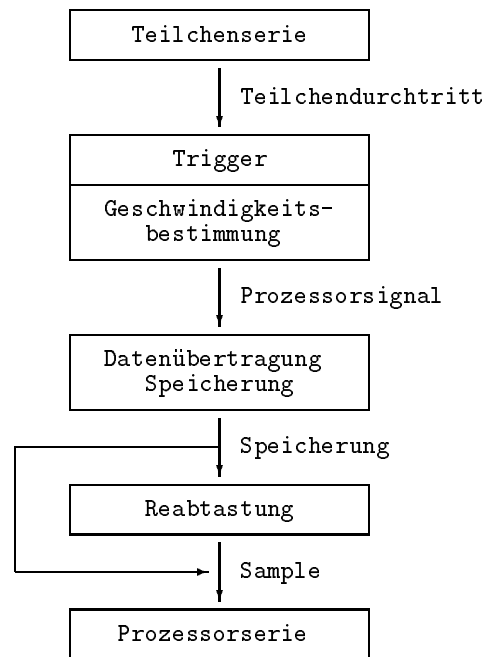
Dadurch entstehen drei interne Zeitebenen

1. Geschwindigkeitsbestimmung
2. Datenübertragung/Speicherung
3. Reabtastung

die parallel bearbeitet werden.

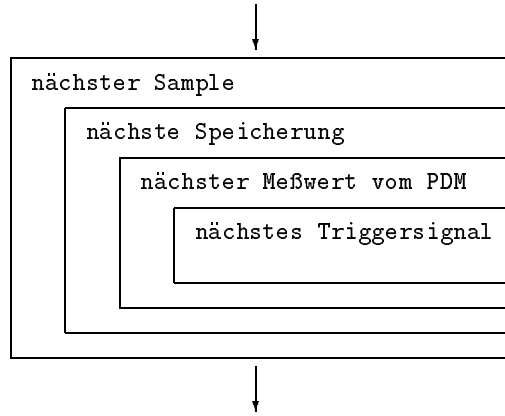
Darüber hinaus wird nach dem Ende eines Verarbeitungsschrittes der nächste ausgelöst. Daher werden Ereigniszeiten und die damit verbundenen Werte stets von einem Verarbeitungsschritt an den nächsten übergeben.





Durch den Vergleich von einlaufenden Ereigniszeiten mit internen Zeiten wird die Reaktion des Verarbeitungsschrittes auf das Ereignis bestimmt. Es wird stets das nächstfolgende Ereignis an den nächsten Verarbeitungsschritt weitergegeben. Dieser reagiert darauf nur, wenn sein eigenes nächstes Ereignis zu einem späteren Zeitpunkt erfolgt.

Dadurch ergibt sich folgende Programmstruktur für die Ausgabe des nächsten, nach der Verarbeitung durch den Prozessor und der Reabtastung im Speicher abgelegten Wertes:



### 1. nächstes Triggersignal

Da als Grenzen des Meßvolumens der Ort des  $e^{\pm 2}$ -Abfalls der Maximalintensität angenommen wird, und zur Simulation die Triggerschwelle der Auswertelektronik auf genau diesem Wert angenommen wird, sind die Signale nach der Triggerung identisch mit den durch die Teilchenserie bereitgestellten Daten.

Um andere Triggerschwellen zu simulieren, ist eine Neufestlegung der Meßvolumenabmessungen vorzunehmen.

Die Ankunftszeit  $t_{0i}$  eines Teilchens im Meßvolumen, die durch die Teilchenserie bereitgestellt wird, beinhaltet die statistische Verteilung der Teilchen im Strömungsvolumen. Die Aufenthaltszeit  $\tau_{mi}$  wird in einem separaten Zufallsprozeß bestimmt. Dadurch errechnen sich die Anfangszeit  $t_a$  und Endzeit  $t_e$  des Burstes nach

$$t_a = t_{0i} - \frac{\tau_{mi}}{2} \quad (3.3.1)$$

und

$$t_e = t_{0i} + \frac{\tau_{mi}}{2} \quad (3.3.2)$$

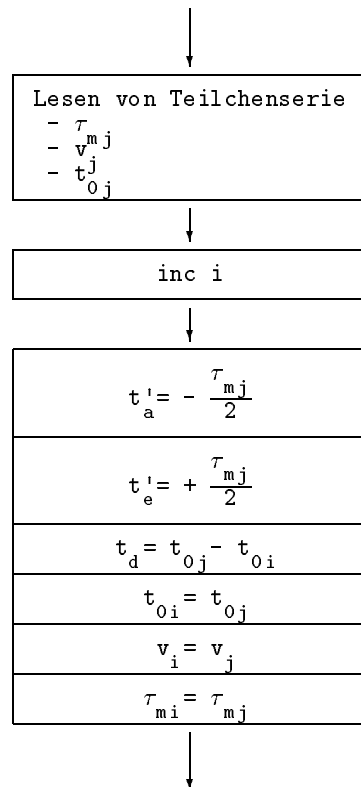
Da die Zeiten zwischen den Teilchendurchritten durch das Meßvolumen um mehrere Größenordnungen von den Zeitdifferenzen bei Mehrfachmessungen innerhalb eines Burstes abweichen können, werden, um die Genauigkeit der Anfangs- und Endzeit des Bursts, der Zeit der nächsten Prozessorfreigabe

$t_f$  (s. nächster Meßwert vom PDM) und der Zeit  $t_l$ , zu der der Zwischenspeicher leer ist (s. nächster Speicherwert), zu erhöhen, diese Werte stets als Differenzen zur Ankunftszeit des aktuellen Teilchens  $t_{0i}$  angeben (\*\*\*)  
Daraus ergeben sich für die Anfangs- und Endzeit des Bursts:

$$t'_a = t_a - t_{0i} = -\frac{\tau_{mi}}{2} \quad (3.3.3)$$

und

$$t'_e = t_e - t_{0i} = +\frac{\tau_{mi}}{2}. \quad (3.3.4)$$



Parameter:

- keine

interne Werte:

- Ankunftszeit des letzten Teilchens  $t_{0i}$
- Geschwindigkeit des letzten Teilchens  $v_i$
- Aufenthaltszeit des letzten Teilchens  $\tau_{mi}$
- Teilchenzähler  $i$

Ausgaben:

- Differenz der Ankunftszeiten des letzten und des aktuellen Teilchens  $t_d$
- Anfangszeit des Burstes des aktuellen Teilchens  $t'_a$
- Endzeit des Burstes des aktuellen Teilchens  $t'_e$
- Ankunftszeit des aktuellen Teilchens  $t_{0i}$
- Geschwindigkeit des aktuellen Teilchens  $v_i$
- Aufenthaltszeit des aktuellen Teilchens  $\tau_{mi}$

## 2. nächster Meßwert vom PDM

Ohne Berücksichtigung der Phasenlage des Burstes ergibt sich die Anzahl der Perioden innerhalb des Burstes aus

$$n = \tau_{mi} \left| \frac{v_i}{\Delta x} + f_s \right|, \quad (3.3.5)$$

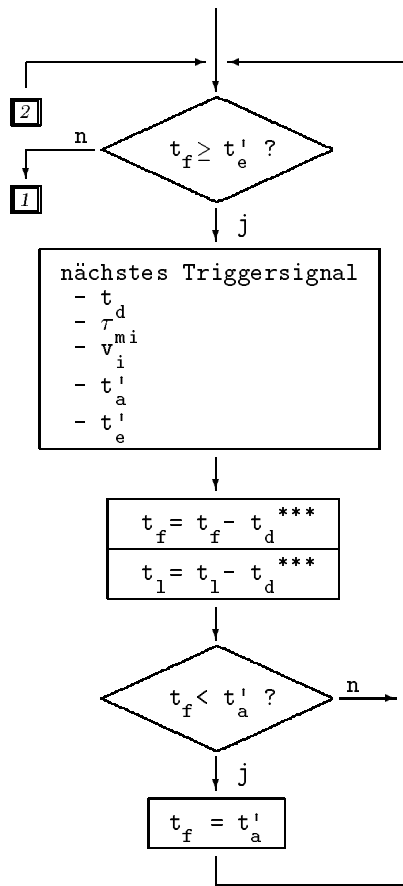
wobei  $\tau_{mi}$  die Aufenthaltszeit des Teilchens im Meßvolumen,  $\Delta x$  der Abstand der Interferenzstreifen,  $v_i$  die Geschwindigkeit des Teilchens und  $f_s$  die Shiftfrequenz ist.

Bei Mehrfachmessung in einem Burst ergibt sich die verbleibende Anzahl der Perioden innerhalb der verbleibenden Zeit  $\Delta t$  aus

$$n(\Delta t, v_i) = \Delta t \left| \frac{v_i}{\Delta x} + f_s \right| \quad (*) \quad (3.3.6)$$

und die Dauer von  $n$  Perioden aus

$$t(n, v_i) = \frac{n}{\left| \frac{v_i}{\Delta x} + f_s \right|}. \quad (**) \quad (3.3.7)$$



Parameter:

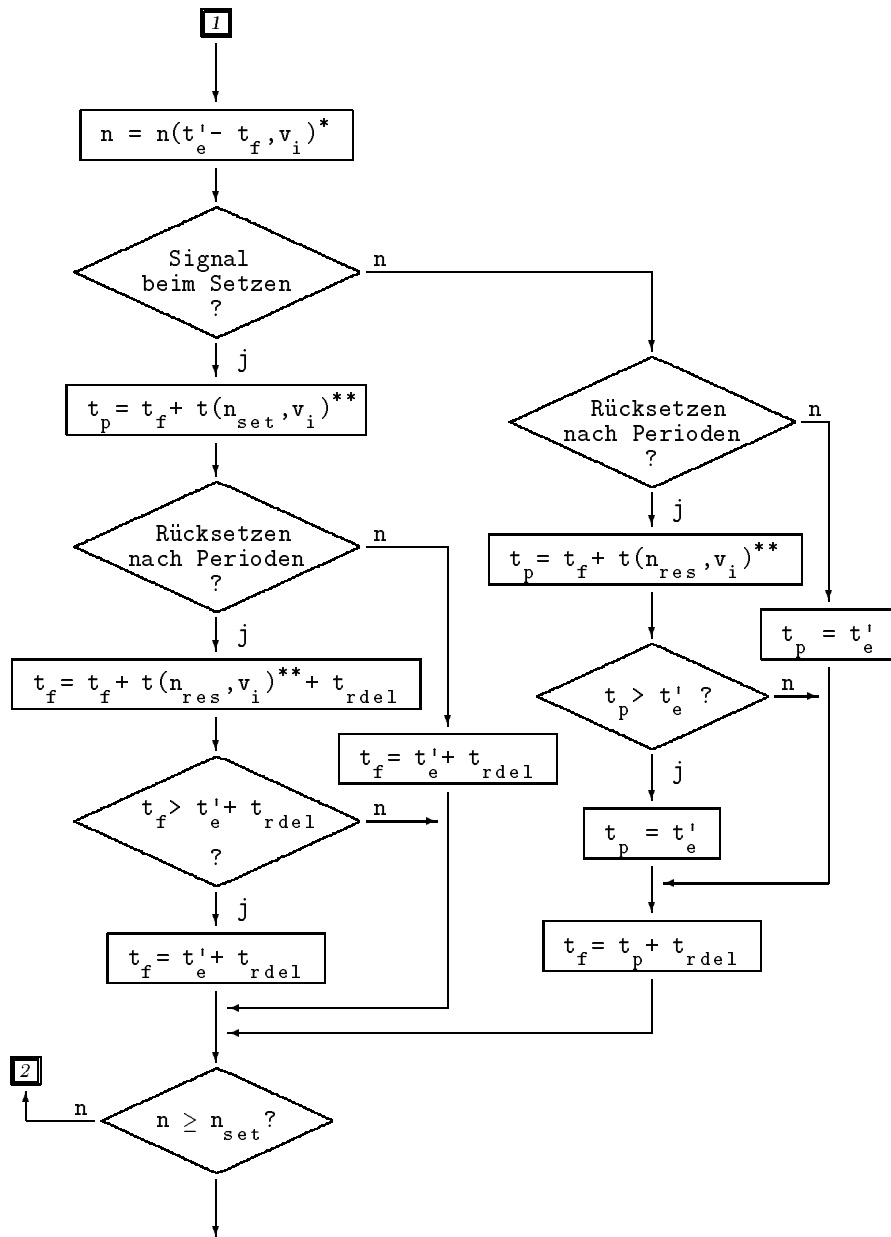
- Shiftfrequenz  $f_s$
- Interferenzstreifenabstand  $\Delta x$
- Datenübertragung nach dem Setzen oder nach dem Rücksetzen des Prozessors
- Rücksetzen am Ende des Burstes oder nach Perioden
- Rücksetzdelaytime  $t_{rdel}$
- Periodenzahl zum Setzen des Prozessors  $n_{set}$
- Periodenzahl zum Rücksetzen des Prozessors  $n_{res}$

interne Werte:

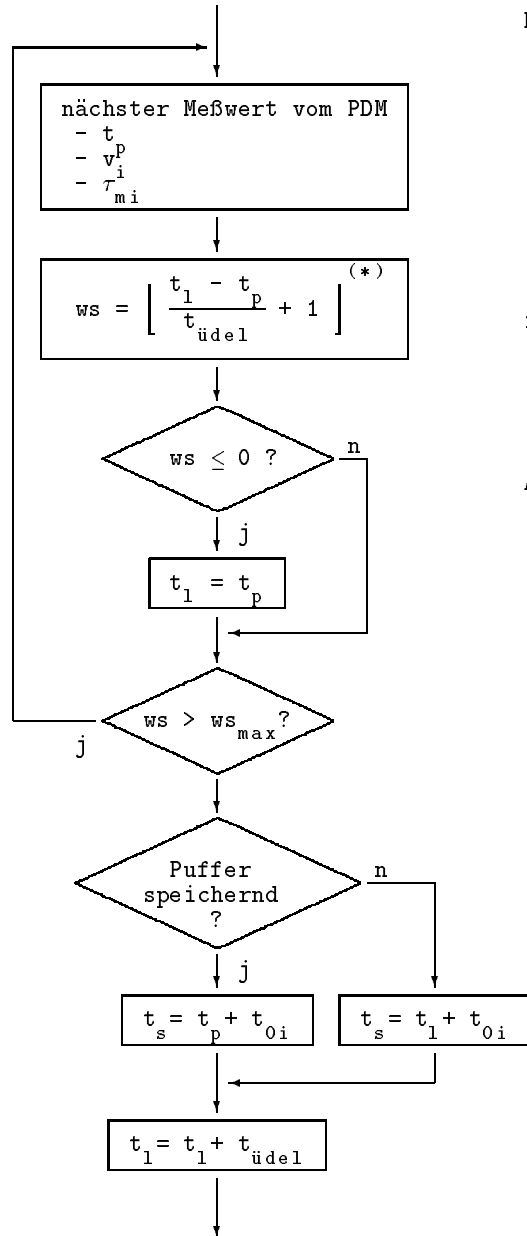
- Zeit der nächsten Prozessorfreigabe  $t_f$
- Ende des Burstes  $t_e'$
- Geschwindigkeit des letzten Teilchens  $v_i$

Ausgaben:

- Zeit des Prozessorsignals  $t_p$
- Geschwindigkeit des letzten Teilchens  $v_i$
- Aufenthaltszeit des letzten Teilchens  $\tau_{mi}$



### 3. nächste Speicherung



Parameter:

- Übertragungsdelaytime  $t_{\text{üdel}}$
- Länge des Puffers  $ws_{\text{max}}$
- Übertragung der Zeit des Prozessorsignals durch den Puffer (speichernd) oder Zeit der Übernahme durch den Rechner

interne Werte:

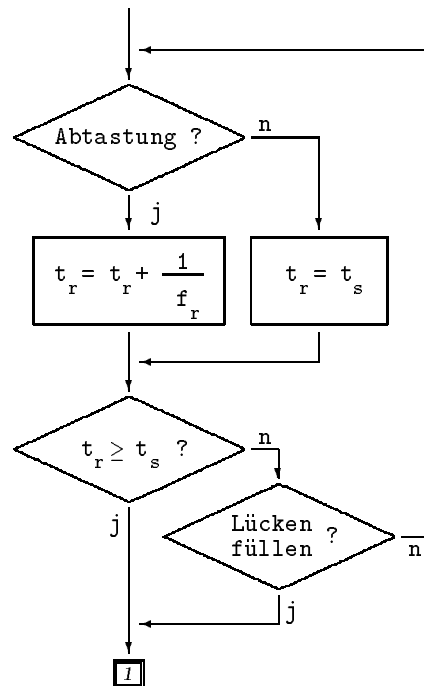
- Anzahl der Werte im Puffer  $ws$
- Zeit, bei der der Puffer leer ist  $t_l$

Ausgaben:

- Zeit der nächsten Speicherung  $t_s$
- Geschwindigkeit des letzten Teilchens  $v_i$
- Aufenthaltszeit des letzten Teilchens  $\tau_{mi}$

(\*)  $\lfloor x \rfloor$  ist die kleinste ganze Zahl von  $x$ .  
Es gilt  $x-1 < \lfloor x \rfloor \leq x$  mit  $\lfloor x \rfloor \in \mathbb{Z}$ .

#### 4. nächster Sample



Parameter:

- Abtastung oder frei laufend
- Abtastfrequenz  $f_r$
- Reaktion auf Intervalle ohne Meßwert (frei lassen oder letzten Wert halten)
- weiterzuleitender Burst im Abtastintervall (erster, letzter, max. Periodenzahl)

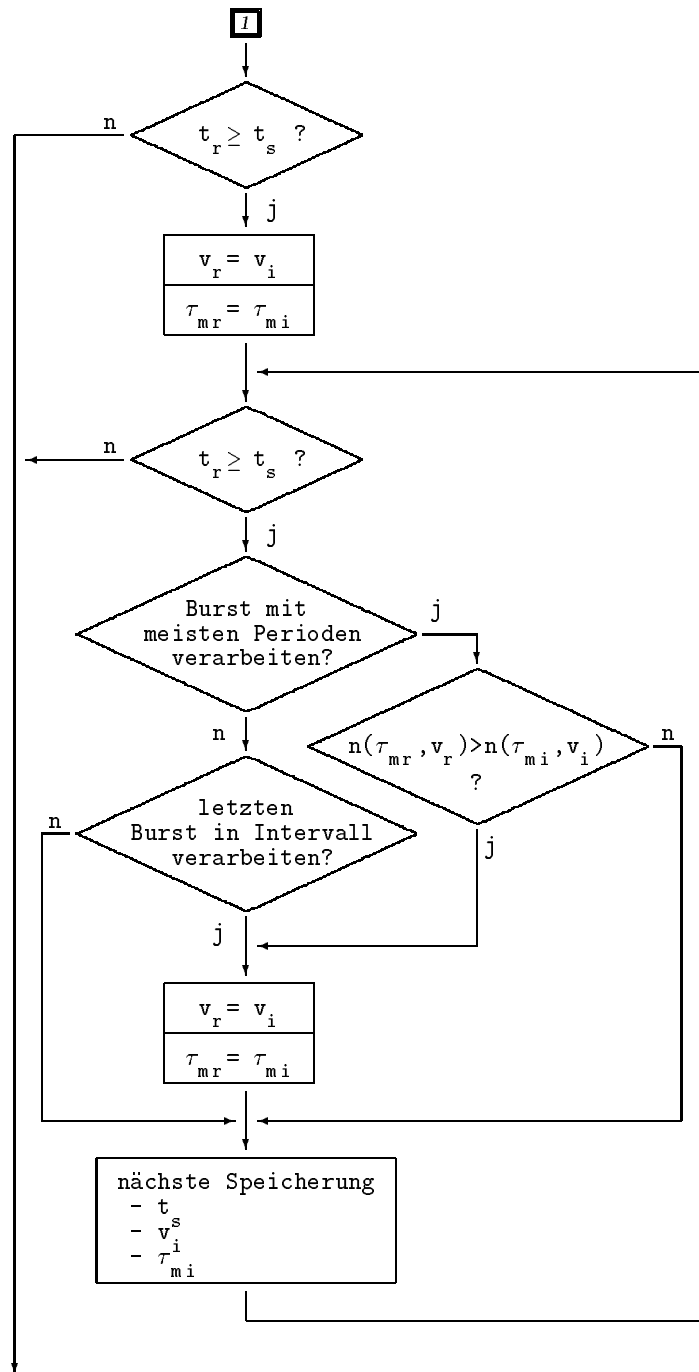
interne Werte:

- Zeit des letzten Samples  $t_r$
- Geschwindigkeit des Teilchens bei der letzten Abtastung  $v_r$
- Aufenthaltszeit des Teilchens bei der letzten Abtastung  $\tau_{mr}$

Ausgaben:

- Zeit des nächsten Samples  $t_r$
- Geschwindigkeit des Teilchens bei der nächsten Abtastung  $v_r$
- Aufenthaltszeit des Teilchens bei der nächsten Abtastung  $\tau_{mr}$





### 3.4 Wichtung eindimensional turbulenter Strömungen

Zur Bestimmung der Zeitmittelwerte aus dem LDA-Meßwertensemble werden gewichtete Mittelungsverfahren eingesetzt.

Um den Einfluß des Prozessors auf die Wirksamkeit der Wichtung zu untersuchen, werden die Wichtungsalgorithmen sowohl auf die Teilchenserie als auch auf die Prozessorserie angewandt.

Die gewonnenen Informationen über die Momente werden mit denen der Grundserie verglichen:

$$\beta_1 = \frac{m_{1\text{TS/PS}} - m_{1\text{GS}}}{m_{1\text{GS}}} \quad (3.4.1)$$

$$\beta_2 = \frac{\mu_{2\text{TS/PS}} - \mu_{2\text{GS}}}{\mu_{2\text{GS}}} \quad (3.4.2)$$

$$\beta_2^*(v_c) = \frac{M_{2\text{TS/PS}}(v_c) - \mu_{2\text{GS}}}{\mu_{2\text{GS}}} \quad (3.4.3)$$

wobei  $m_i$  die Anfangsmomente,  $\mu_i$  die Zentralmomente und  $M_i(v_c)$  die Momente bezüglich  $v_c$   $i$ -ter Ordnung sind (vgl. (2.10) und (2.11)).

Die Momente der gewichteten Geschwindigkeitssamples sind wie folgt definiert (vgl. [3]):

- Anfangsmoment 1. Ordnung:

$$m_1 = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{v_{xi}}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (3.4.4)$$

- Anfangsmoment 2. Ordnung:

$$m_2 = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{v_{xi}^2}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (3.4.5)$$

- Zentralmoment 2. Ordnung:

$$\mu_2 = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{(v_{xi} - m_1)^2}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (3.4.6)$$

- Moment 2. Ordnung bzgl.  $v_c$ :

$$M_2(v_c) = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{(v_{xi} - v_c)^2}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (3.4.7)$$

- Anfangsmoment 3. Ordnung:

$$m_3 = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{v_{xi}^3}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (3.4.8)$$

- Zentralmoment 3. Ordnung:

$$\mu_3 = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{(v_{xi} - m_1)^3}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (3.4.9)$$

- Moment 3. Ordnung bzgl.  $v_c$ :

$$M_3(v_c) = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^N \frac{(v_{xi} - v_c)^3}{\gamma_{0i}}}{\sum_{i=1}^N \frac{1}{\gamma_{0i}}} \quad (3.4.10)$$

Nach dem Verschiebungssatz gilt:

$$\mu_2 = m_2 - m_1^2 \quad (3.4.11)$$

$$M_2(v_c) = m_2 - 2v_c m_1 + v_c^2 \quad (3.4.12)$$

$$\mu_3 = m_3 - 3m_1 m_2 + 2m_1^3 \quad (3.4.13)$$

$$M_3(v_c) = m_3 - 3v_c m_2 + 3v_c^2 m_1 - v_c^3 \quad (3.4.14)$$

Für  $\gamma_{0i}$  werden die durch die einzelnen Wichtungsverfahren definierten Werte eingesetzt:

- Zeitdifferenzwichtung [5]/[6]:

$$\gamma_{0i} = \frac{1}{t_{0i} - t_{0i\perp 1}} \quad (3.4.15)$$

- Geschwindigkeitswichtung [7]:

$$\gamma_{0i} = v_{xi} \quad (3.4.16)$$

- Aufenthaltszeitwichtung [8]:

$$\gamma_{0i} = \frac{1}{\tau_{mi}} \quad (3.4.17)$$

- Wichtung nach Nakoa [9]:

$$\gamma_{0i} = \left( 2 - \frac{v_{xi}}{\text{EX}_{\text{Ens}}(v_x)} \right)^{\perp \frac{3}{2}} \quad (3.4.18)$$

Soll keine Wichtung durchgeführt werden so ist  $\gamma_{0i} = 1$  zu setzen.

## 3.5 Grundserie dreidimensional turbulenter Strömungen

### 1. Konzept

Bei der Simulation dreidimensional turbulenter Strömungen wird die zur Simulation eindimensional turbulenter Strömungen verwendete Rekursion

$$z_t = \phi z_{t-1} + a_t \quad (3.5.1)$$

erweitert.

Es werden drei unabhängige eindimensionale Zeitreihen jeweils durch (3.5.1) erzeugt. Hierdurch entsteht ein Vektor

$$\vec{z}_t = \begin{pmatrix} z_{x_t} \\ z_{y_t} \\ z_{z_t} \end{pmatrix} = \begin{pmatrix} \phi_x z_{x_{t-1}} + a_{x_t} \\ \phi_y z_{y_{t-1}} + a_{y_t} \\ \phi_z z_{z_{t-1}} + a_{z_t} \end{pmatrix} \quad (3.5.2)$$

mit

$$a_{i_t} = \sqrt{1 - \phi_i^2} x_{i_t} \quad (3.5.3)$$

wobei die  $x_i$  normalverteilt sind mit dem Erwartungswert 0 und der Varianz 1.

Der simulierte Strömungsvektor  $\vec{v}_t$  zum Zeitpunkt  $t$  ergibt sich aus

$$\vec{v}_t = C \vec{z}_t + \vec{v} \quad (3.5.4)$$

mit

$$\vec{v} = \frac{1}{T} \int_0^T \vec{v} dt = \frac{1}{T} \begin{pmatrix} \int_0^T v_x dt \\ \int_0^T v_y dt \\ \int_0^T v_z dt \end{pmatrix} = \begin{pmatrix} \overline{v_x} \\ \overline{v_y} \\ \overline{v_z} \end{pmatrix} \quad (3.5.5)$$

und der symmetrischen Matrix

$$C = \begin{pmatrix} c_{xx} & c_{xy} & c_{xz} \\ c_{yx} & c_{yy} & c_{yz} \\ c_{zx} & c_{zy} & c_{zz} \end{pmatrix} \quad (3.5.6)$$

mit

$$\begin{aligned}
c_{xy} &= c_{yx} \\
c_{xz} &= c_{zx} \\
c_{yz} &= c_{zy}
\end{aligned}$$

## 2. Eigenschaften

Für die Varianzen der Zufallsgrößen  $z$  und  $a$  in (3.5.1) gilt:

$$\sigma_z^2 = \frac{\sigma_a^2}{1 - \phi^2} \quad (3.5.7)$$

Für das integrale Zeitmaß der durch (3.5.1) erzeugten Zeitreihe  $z$  gilt:

$$\vartheta = \frac{1}{1 - \phi} \quad (3.5.8)$$

Nach (3.5.3) gilt für die Varianz der Zufallsgröße  $a_i$ :

$$\sigma_{a_i}^2 = 1 - \phi_i^2 \quad (3.5.9)$$

Somit gilt für die Varianz der Zufallsgröße  $z_i$  nach (3.5.7):

$$\sigma_{z_i}^2 = 1 \quad (3.5.10)$$

und nach (3.5.8) für die integralen Zeitmaße der durch (3.5.2) erzeugte Zeitreihen  $z_i$ :

$$\vartheta_{z_i} = \frac{1}{1 - \phi_i} \quad (3.5.11)$$

Für die Kovarianzmatrix der Zufallsgröße  $\vec{v}$  in (3.5.4) gilt dann:

$$\Gamma_v = \begin{pmatrix} \gamma_{xx} & \gamma_{xy} & \gamma_{xz} \\ \gamma_{yx} & \gamma_{yy} & \gamma_{yz} \\ \gamma_{zx} & \gamma_{zy} & \gamma_{zz} \end{pmatrix} \quad (3.5.12)$$

mit

$$\gamma_{xx} = c_{xx}^2 + c_{xy}^2 + c_{xz}^2 \quad (3.5.12.1)$$

$$\gamma_{yy} = c_{yx}^2 + c_{yy}^2 + c_{yz}^2 \quad (3.5.12.2)$$

$$\gamma_{zz} = c_{zx}^2 + c_{zy}^2 + c_{zz}^2 \quad (3.5.12.3)$$

$$\gamma_{xy} = \gamma_{yx} = c_{xx}c_{yx} + c_{xy}c_{yy} + c_{xz}c_{yz} \quad (3.5.12.4)$$

$$\gamma_{xz} = \gamma_{zx} = c_{xx}c_{zx} + c_{xy}c_{zy} + c_{xz}c_{zz} \quad (3.5.12.5)$$

$$\gamma_{yz} = \gamma_{zy} = c_{yx}c_{zx} + c_{yy}c_{zy} + c_{yz}c_{zz} \quad (3.5.12.6)$$

Für das integrale Zeitmaß  $\Theta_v$  der durch (3.5.4) erzeugten Zeitreihe  $\vec{v}$  gilt:

$$\Theta_v = \begin{pmatrix} \vartheta_{xx} & \vartheta_{xy} & \vartheta_{xz} \\ \vartheta_{yx} & \vartheta_{yy} & \vartheta_{yz} \\ \vartheta_{zx} & \vartheta_{zy} & \vartheta_{zz} \end{pmatrix}^* \quad (3.5.13)$$

mit

$$\vartheta_{xx} = \frac{c_{xx}^2 \frac{1}{1-\phi_x} + c_{xy}^2 \frac{1}{1-\phi_y} + c_{xz}^2 \frac{1}{1-\phi_z}}{c_{xx}^2 + c_{xy}^2 + c_{xz}^2} \quad (3.5.13.1)$$

$$\vartheta_{yy} = \frac{c_{yx}^2 \frac{1}{1-\phi_x} + c_{yy}^2 \frac{1}{1-\phi_y} + c_{yz}^2 \frac{1}{1-\phi_z}}{c_{yx}^2 + c_{yy}^2 + c_{yz}^2} \quad (3.5.13.2)$$

$$\vartheta_{zz} = \frac{c_{zx}^2 \frac{1}{1-\phi_x} + c_{zy}^2 \frac{1}{1-\phi_y} + c_{zz}^2 \frac{1}{1-\phi_z}}{c_{zx}^2 + c_{zy}^2 + c_{zz}^2} \quad (3.5.13.3)$$

$$\vartheta_{xy} = \vartheta_{yx} = \frac{c_{xx}c_{yx} \frac{1}{1-\phi_x} + c_{xy}c_{yy} \frac{1}{1-\phi_y} + c_{xz}c_{yz} \frac{1}{1-\phi_z}}{c_{xx}c_{yx} + c_{xy}c_{yy} + c_{xz}c_{yz}} \quad (3.5.13.4)$$

\*Die Komponente  $\vartheta_{ij}$  der Matrix  $\Theta_v$  ist definiert als

$$\vartheta_{ij} = \int_0^\infty \frac{\varphi_{ij}(t)}{\varphi_{ij}(0)} dt$$

wobei  $\varphi_{ij}(t)$  die Kreuzkorrelationsfunktion der Komponenten  $i$  und  $j$  von  $\vec{v}$  ist.

$$\vartheta_{xz} = \vartheta_{zx} = \frac{c_{xx}c_{zx}\frac{1}{1-\phi_x} + c_{xy}c_{zy}\frac{1}{1-\phi_y} + c_{xz}c_{zz}\frac{1}{1-\phi_z}}{c_{xx}c_{zx} + c_{xy}c_{zy} + c_{xz}c_{zz}} \quad (3.5.13.5)$$

$$\vartheta_{yz} = \vartheta_{zy} = \frac{c_{yx}c_{zx}\frac{1}{1-\phi_x} + c_{yy}c_{zy}\frac{1}{1-\phi_y} + c_{yz}c_{zz}\frac{1}{1-\phi_z}}{c_{yx}c_{zx} + c_{yy}c_{zy} + c_{yz}c_{zz}} \quad (3.5.13.6)$$

### 3. Vorgehen bei der Simulation

Bei der Simulation dreidimensional turbulenter Strömungen werden bei Verwendung des Reynolds'schen Ansatzes die Mittelwerte der einzelnen Geschwindigkeitskomponenten  $\overline{v_x}$ ,  $\overline{v_y}$ ,  $\overline{v_z}$  und deren Varianzen  $\overline{v_x'^2}$ ,  $\overline{v_y'^2}$ ,  $\overline{v_z'^2}$  sowie deren Kreuzkovarianzen  $\overline{v_x'v_y'}$ ,  $\overline{v_x'v_z'}$ ,  $\overline{v_y'v_z'}$  und die integralen Zeitmaße  $\vartheta_{xx}$ ,  $\vartheta_{yy}$ ,  $\vartheta_{zz}$  der Komponenten als Eingabeparameter vorausgesetzt.

(3.5.12) ergibt dann ein Gleichungssystem für das symmetrische  $C$ . Dieses wird durch Iteration gelöst. Die Wurzel für  $C$  wird in die Gleichungen (3.5.13.1) bis (3.5.13.3) eingesetzt. Hierdurch entsteht ein lineares Gleichungssystem für  $\frac{1}{1\pm\phi_x}$  bis  $\frac{1}{1\pm\phi_z}$  dessen Wurzeln nach den  $\phi_x$  bis  $\phi_z$  umgestellt werden.

Durch Einsetzen der Wurzeln in (3.5.13.4) bis (3.5.13.6) ergeben sich die verbleibenden Größen von  $\Theta_v$  und durch Einsetzen der erhaltenen  $\phi_i$  in (3.5.2) bis (3.5.4) ergibt sich die Zeitreihe  $\vec{v}$  der simulierten dreidimensional turbulenten Strömung.



### 3.6 Teilchenserie dreidimensional turbulenter Strömungen

Da die Größe der Projektionsfläche  $A_{\perp}$  des Meßvolumens in Richtung der Strömung von dieser abhängig ist, ergibt sich die momentane Meßvolumenrate  $S_0(t)$  aus (vgl. (3.2.2))

$$S_0(t) = c_V(\vec{v}(t))A_{\perp}(\vec{v}(t))|\vec{v}(t)|. \quad (3.6.1)$$

Die Meßvolumenrate kann bei der dreidimensionalen Simulation jedoch nicht auf das integrale Zeitmaß bezogen werden, da es, wie bei der Generierung der dreidimensionalen Grundserie gezeigt, eine Matrix ist. Daher werden alle nötigen Zeitangaben auf die Komponente  $\vartheta_{ii}$  des integralen Zeitmaßes  $\Theta$  der gemessenen Strömungsrichtung  $i$  bezogen.

#### 1. Berechnung der Projektionsfläche $A_{\perp}(\vec{v}(t))$ des Meßvolumens in Strömungsrichtung

Die Richtung eines gegebenen Strömungsvektors  $\vec{v}(t)$  läßt sich in einem Kugelkoordinatensystem [14] (s. Abb. 3.6.1) durch die Winkel

$$\varphi = \arctan \frac{v_y}{v_x} \quad (3.6.2)$$

und

$$\zeta = \arctan \frac{\sqrt{v_x^2 + v_y^2}}{v_z} \quad (3.6.3)$$

mit

$$\vec{v} = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$$

beschreiben.

Für die Komponenten des Geschwindigkeitsvektors gilt dann:

$$\begin{aligned} v_x &= v \cos \varphi \sin \zeta \\ v_y &= v \sin \varphi \sin \zeta \\ v_z &= v \cos \zeta \end{aligned} \quad (3.6.4)$$

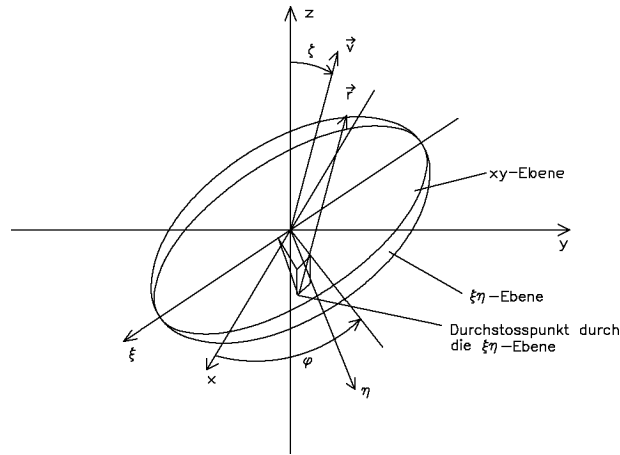


Abbildung 3.6.1: Vektorielle Darstellung der Teilchenbahn [14]

Die vektorielle Darstellung der Teilchenbahn  $\vec{r}$  ergibt sich aus der Summe eines Punktes  $\vec{r}_0$  und dem parametrisierten Geschwindigkeitsrichtungsvektor.

$$\vec{r} = \vec{r}_0 + \Lambda \vec{v} \quad (3.6.5)$$

Der Parameter  $\Lambda$  ist demzufolge die Zeit, die ein Teilchen in der Strömung von einem Punkt  $\vec{r}$  zu einem anderen bei festem  $\vec{r}_0$  benötigt.

Ohne Einschränkung der Allgemeinheit liege  $\vec{r}_0$  in der Normalenebene von  $\vec{v}$ . In dieser Ebene befinde sich ein rechtwinkliges Koordinatensystem  $\xi, \eta$ , wobei der Ursprung dieses Koordinatensystems im Ursprung des Koordinatensystems  $x, y, z$  liege und die  $\xi$ -Achse die Schnittgerade der Normalenebene von  $\vec{v}$  und der  $x$ - $y$ -Ebene sei (s. Abb. 3.6.1).

Dann kann die Bahn des Teilchens  $\vec{r}$  mit  $\Lambda$  als Parameter wie folgt dargestellt werden.

$$\vec{r} = \begin{pmatrix} r_x \\ r_y \\ r_z \end{pmatrix} = \Lambda v \begin{pmatrix} \cos \varphi \sin \zeta \\ \sin \varphi \sin \zeta \\ \cos \zeta \end{pmatrix} + \begin{pmatrix} +\xi \sin \varphi + \eta \cos \varphi \cos \zeta \\ -\xi \cos \varphi + \eta \sin \varphi \cos \zeta \\ -\eta \sin \zeta \end{pmatrix} \quad (3.6.6)$$

Durch Einsetzen in die Ellipsoidgleichung der Meßvolumengrenze

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \quad (3.6.7)$$

ergibt sich für den Parameter  $\Lambda$  die quadratische Gleichung

$$A\Lambda^2 + B\Lambda + C = 0 \quad (3.6.8)$$

mit

$$\begin{aligned}
A &= \frac{v^2}{a^2} \cos^2 \varphi \sin^2 \zeta + \frac{v^2}{b^2} \sin^2 \varphi \sin^2 \zeta + \frac{v^2}{c^2} \cos^2 \zeta \\
&= \frac{v_x^2}{a^2} + \frac{v_y^2}{b^2} + \frac{v_z^2}{c^2} \\
B &= \frac{v}{a^2} (2\xi \sin \varphi \cos \varphi \sin \zeta + 2\eta \cos^2 \varphi \sin \zeta \cos \zeta) \\
&\quad + \frac{v}{b^2} (-2\xi \sin \varphi \cos \varphi \sin \zeta + 2\eta \sin^2 \varphi \sin \zeta \cos \zeta) \\
&\quad + \frac{v}{c^2} (-2\eta \sin \zeta \cos \zeta) \\
&= \frac{2v_x}{a^2} (\xi \sin \varphi + \eta \cos \varphi \cos \zeta) + \frac{2v_y}{b^2} (\xi \cos \varphi + \eta \sin \varphi \cos \zeta) \\
&\quad + \frac{2v_z}{c^2} \eta \sin \zeta \\
C &= \frac{1}{a^2} (\xi^2 \sin^2 \varphi + \eta^2 \cos^2 \varphi \cos^2 \zeta + 2\xi\eta \sin \varphi \cos \varphi \cos \zeta) \\
&\quad + \frac{1}{b^2} (\xi^2 \cos^2 \varphi + \eta^2 \sin^2 \varphi \cos^2 \zeta - 2\xi\eta \sin \varphi \cos \varphi \cos \zeta) \\
&\quad + \frac{1}{c^2} \eta^2 \sin^2 \zeta - 1 \\
&= \frac{1}{a^2} (\xi \sin \varphi + \eta \cos \varphi \cos \zeta)^2 + \frac{1}{b^2} (\xi \cos \varphi - \eta \sin \varphi \cos \zeta)^2 \\
&\quad + \frac{1}{c^2} (\eta \sin \zeta)^2 - 1
\end{aligned}$$

Für die Wurzeln der Gleichung gilt dann

$$\Lambda_{1,2} = -\frac{B}{2A} \pm \frac{1}{2A} \sqrt{B^2 - 4AC}. \quad (3.6.9)$$

Die Projektion des Ellipsoids in die Normalenebene zum Geschwindigkeitsvektor  $\vec{v}$  ergibt sich dann aus der Lösung des Grenzfalles  $\Lambda_1 = \Lambda_2$ . Für diesen gilt

$$B^2 - 4AC = 0. \quad (3.6.10)$$

Von dieser Bedingung ausgehend, ergibt sich eine Gleichung für  $\xi$  und  $\eta$  der Form

$$K\xi^2 + L\eta^2 + M\xi\eta - N = 0 \quad (3.6.11)$$

mit

$$\begin{aligned} K &= \frac{\sin^2 \zeta}{a^2 b^2} + \frac{\sin^2 \varphi \cos^2 \zeta}{a^2 c^2} + \frac{\cos^2 \varphi \cos^2 \zeta}{b^2 c^2} \\ L &= \frac{\cos^2 \varphi}{a^2 c^2} + \frac{\sin^2 \varphi}{b^2 c^2} \\ M &= 2 \sin \varphi \cos \varphi \cos \zeta \left( \frac{1}{a^2 c^2} - \frac{1}{b^2 c^2} \right) \\ N &= \frac{1}{a^2} \cos^2 \varphi \sin^2 \zeta + \frac{1}{b^2} \sin^2 \varphi \sin^2 \zeta + \frac{1}{c^2} \cos^2 \zeta. \end{aligned}$$

Die Projektion des Ellipsoids in die Normalenebene von  $\vec{v}$  ist also eine Ellipse im Koordinatensystem  $\xi, \eta$  in Ursprungslage um den Ursprung gedreht. Durch die Drehung des Koordinatensystems  $\xi, \eta$  um den Winkel  $\nu$ , wobei

$$\tan 2\nu = \frac{M}{K - L}, \quad (3.6.12)$$

fällt das Mischglied  $M\xi\eta$  heraus und man erhält die Halbachsen der Ellipse aus

$$\begin{aligned} a' &= \sqrt{\frac{N(1 + \tan^2 \nu)}{K + M \tan \nu + L \tan^2 \nu}} \\ b' &= \sqrt{\frac{N(1 + \tan^2 \nu)}{L - M \tan \nu + K \tan^2 \nu}} \end{aligned} \quad (3.6.13)$$

Die Lage der Halbachsen im  $\xi$ - $\eta$ -Koordinatensystem ergibt sich aus

$$\begin{aligned} \vec{a}' &= a' \begin{pmatrix} \cos \nu \\ \sin \nu \end{pmatrix} \\ \vec{b}' &= b' \begin{pmatrix} -\sin \nu \\ \cos \nu \end{pmatrix} \end{aligned} \quad (3.6.14)$$

Der Flächeninhalt der Projektionsfläche ergibt sich dann aus  $A_{\perp}(\vec{v}(t)) = \pi a' b'$  zu

$$A_{\perp}(\vec{v}(t)) = \pi \sqrt{\frac{4N^2}{4LK - M^2}} \quad (3.6.15)$$

$$A_{\perp}(\vec{v}(t)) |\vec{v}(t)| = \pi abc \sqrt{A}. \quad (3.6.16)$$

## 2. Berechnung der Aufenthaltszeit

Zur Berechnung der Aufenthaltszeit wird angenommen, daß die Teilchen gleichverteilt über die Projektionsfläche  $A_{\perp}$  des Meßvolumens in Strömungsrichtung durch dieses hindurchtreten. Hierzu müssen über einer elliptischen Fläche im Raum mit den Halbachsen (3.6.14) gleichverteilte Zufallszahlen erzeugt werden.

Hierzu wird das bei der Generierung der eindimensionalen Teilchenserie verwendete Verfahren eingesetzt.

Das die Projektionsfläche umschließende Rechteck wird so gelegt, daß seine Kanten parallel zu den Halbachsen liegen. Die zufälligen Abstände  $\tilde{a}'$  und  $\tilde{b}'$  des Durchstoßpunktes der Teilchenbahn durch die Normalebene von  $\vec{v}$  von den Halbachsen ergeben sich aus

$$\begin{aligned} \tilde{a}' &= a' \tilde{r}_{a'} \\ \tilde{b}' &= b' \tilde{r}_{b'} \end{aligned} \quad (3.6.17)$$

mit gleichverteilten Zufallszahlen  $\tilde{r}_{a'}$  und  $\tilde{r}_{b'}$  aus  $(-1; 1)$ .

Somit gilt für den Durchstoßpunkt  $\vec{r}_0$  der Teilchenbahn durch die Normalebene zu  $\vec{v}$

$$\vec{r}_0 = \tilde{a}' \vec{e}_1 + \tilde{b}' \vec{e}_2 = \tilde{a}' \begin{pmatrix} \cos \nu \\ \sin \nu \end{pmatrix} + \tilde{b}' \begin{pmatrix} -\sin \nu \\ \cos \nu \end{pmatrix} \quad (3.6.18)$$

Somit gilt für den Durchstoßpunkt

$$\begin{aligned} \xi_0 &= \tilde{a}' \cos \nu - \tilde{b}' \sin \nu \\ \eta_0 &= \tilde{a}' \sin \nu + \tilde{b}' \cos \nu \end{aligned} \quad (3.6.19)$$

Gleichung (3.6.9) liefert zwei Wurzeln  $\Lambda_1$  und  $\Lambda_2$  deren Differenz gleich der Aufenthaltszeit  $\tau_m$  ist.

$$\tau_m = \Lambda_2 - \Lambda_1 = \frac{1}{A} \sqrt{B^2 - 4AC} \quad (3.6.20)$$

### 3. Berechnung der Ankunftszeit

Da die Strömungsgeschwindigkeit zwischen den Samples der Grundserie durch eine Funktion 0. Ordnung approximiert wird, ist für die Zeit  $\Delta t$  zwischen den Samples die Strömungsrichtung konstant.

Damit ist auch der Querschnitt des Strömungsvolumens, das Teilchen enthält, die das Meßvolumen in  $\Delta t$  passieren, konstant. Sind die Teilchen in diesem gleichverteilt, so sind auch die Komponenten ihrer Abstände von einer festen Normalenebene zur Strömungsrichtung gleichverteilt.

Somit sind die Komponenten der Abstände eines Teilchens zum nächsten in Strömungsrichtung exponentialverteilt (s. Anhang 1).

Durch dieses Modell beschrieben, werden nach der bei der Teilchenserie eindimensional turbulenter Strömungen genannten Methode die Durchstoßzeitpunkte  $t_{0\xi\eta}$  der Teilchen durch die Normalenebene bestimmt.

Wird die Ankunftszeit des Teilchens als die Mitte zwischen Eintritt des Teilchens in das Meßvolumen und dessen Austritt definiert nach

$$t_0 = \frac{\Lambda_1 + \Lambda_2}{2} + t_{0\xi\eta}, \quad (3.6.21)$$

so gelten für die Prozessorserie die bei der Prozessorserie eindimensional turbulenter Strömungen eingeführten Formeln.

## 4 Umsetzung in eine Programmiersprache

### 4.1 Auswahl der Programmiersprache

Vor der Realisierung der betrachteten Algorithmen durch ein Programm mußte die Frage der Auswahl geeigneter Programmiersprachen entschieden werden. Die Erfahrungen während des Praktikums im Sommer 1991 am Lehrstuhl für Strömungsmechanik der FAU in Erlangen ließen die Entscheidung der Frage der Programmierung des Programmkerns in Fortran oder Turbo-Pascal offen [15].

Daher wurde ein Testprogramm (s. Anhang 2.1 und 2.2) entwickelt, das mit der Generierung von Zahlen und einer statistischen Auswertung dieser eine typische Aufgabe des fertigen Simulationsprogrammes realisiert. Dieses Testprogramm wurde sowohl unter Fortran als auch unter Turbo-Pascal realisiert und für verschiedene Rechner compiliert.

Die Testdurchläufe ergaben in Abhängigkeit vom Rechnertyp und dessen Ausstattung folgende Bearbeitungszeiten:

	Anzahl der Durchgänge	Bearbeitungszeit unter	
		MS Fortran	Turbo-Pascal
386DX mit N-Co.	100	1:45	2:29
386DX ohne N-Co.	10	5:27	0:57
286 ohne N-Co.	1	1:10	0:13

N-Co.: Numeric-Co-Prozessor

Die hohe Geschwindigkeit des Fortran-Programms bei Verwendung eines Arithmetik-Co-Prozessors verdeutlicht, wie maschinennahe das Programm arbeitet. Während das in Turbo-Pascal geschriebene Programm auf anderen Rechnerkonfigurationen eindeutig schneller arbeitet als das unter Fortran realisierte, ist unter Verwendung eines Numeric-Co-Prozessors das unter Fortran realisierte Programm schneller. Da jedoch in diesem Testprogramm nur eine typische Aufgabe der Seriengenerierung bearbeitet wird, wurde ein weiteres Testprogramm (s. Anhang 2.3 und 2.4) entwickelt. Das unter Fortran geschriebene Programm „AR“ ist identisch mit dem Teilprogramm „AR“, welches während des Praktikums in Erlangen entwickelt wurde. Die gleiche Aufgabe realisiert das unter Turbo-Pascal geschriebene Programm „AR“. Dieses Programm hat sowohl die gleichen Ein- und Ausgaben als auch die

gleiche Struktur wie das Fortran-Programm. Die Testdurchläufe ergaben folgende Bearbeitungszeiten:

eingestellte Parameter	Bearbeitungszeit unter	
	MS Fortran	Turbo-Pascal
mittl. Geschwindigkeit: 100	4:49	4:46
Turbulenzgrad in %: 1		
Koeffizient c1: 0.9		
Meßzeit (/1000): 100		
Verteilungsfunktion: normal		

Da bei der konkreten Realisierung der Generierungsroutinen nicht nur numerische Operationen auftreten, verschwindet der Vorteil der schnelleren Bearbeitung der numerischen Aufgaben unter Fortran. Darüber hinaus ist die Funktion des kompilierten Programmes nicht für jeden Rechner gewährleistet. So funktioniert das Fortran-Programm auf einem Rechner ohne Co-Prozessor gar nicht. Die Software muß also auf den Rechner abgestimmt werden. Dies geschieht i. allg. durch Neucompilieren des Programmes. Das setzt das Vorhandensein geeigneter Compiler voraus. Da aber die Compilierung zumeist nicht auf dem Rechner erfolgt, auf dem die Software dann laufen soll, müßte eine Variante des Programmes entwickelt werden, die möglichst maschinenunabhängig arbeitet, was unweigerlich Geschwindigkeitseinbußen zur Folge hätte.

Ein Turbo-Pascal-Programm kann so kompiliert werden, daß es, ohne große Geschwindigkeitsverluste, selbständig das Vorhandensein eines Co-Prozessors erkennt und diesen in den Programmablauf einbindet. Dadurch ist gewährleistet, daß das Programm auf allen Rechnern funktioniert und dabei auf gut ausgestatteten Rechnern automatisch entsprechend schneller arbeitet.

Durch eine zeitoptimalere Struktur des weiterentwickelten Programms konnte inzwischen die im zweiten Testprogramm bearbeitete Aufgabe der Generierung von Samples der Geschwindigkeitsgrundserie stark verkürzt werden. Für die zum Vergleich herangezogene Generierung von 100 000 Samples benötigt das überarbeitete Programm nur noch 45s.

Die Entscheidung, die Programmierung ausschließlich unter Turbo-Pascal durchzuführen, hat außerdem den Vorteil der Einheitlichkeit der Benutzeroberfläche und des Programmkerns.



## 4.2 Programmoberfläche

Das Programm wird mit „Entry“ aufgerufen. Das Programmlogo wird sichtbar. Durch Betätigung der Taste „Enter“ oder Anklicken von „Ok“ mit der Maus schließt es sich. Das Menüfeld ist dann aktiviert. Das Menü kann dann grundsätzlich mit einer Maus oder durch die Tastatur aufgerufen werden.

### 1. Arbeiten mit einer Maus

Durch Anklicken der Menüfelder werden, falls vorhanden, Untermenüs geöffnet. Diese können ebenfalls durch Anklicken der Menüfelder ausgewählt werden. Letztendlich werden Dialogfenster geöffnet.

Es gibt Dialogfenster, die die Eingabe von Parametern oder Angabe von Auswahldaten ermöglichen. Die Angaben werden durch „Ok“ bestätigt. Diese Dialoge können durch „Cancel“ abgebrochen werden.

Daneben gibt es Dialogfenster, die Mitteilungen enthalten. Sie können durch „Ok“ geschlossen werden.

In den Parameterdialogen kann durch Anklicken der Eingabezeile oder des links davon stehenden Textes zwischen den Eingabezeilen gesprungen werden. Die Eingabe erfolgt dann mittels der Tastatur.

Die Ansichtfenster enthalten links oben das „Schließen-Kästchen“ und rechts oben den „Zoom-Pfeil“. Durch Anklicken des Kästchens wird das gerade aktive Fenster geschlossen. Durch Anklicken des Pfeils wird das Fenster auf den gesamten Bildschirm ausgedehnt bzw. wieder auf Originalgröße gebracht. Durch Anklicken und Ziehen des oberen Randes kann das Fenster verschoben werden. Durch Anklicken und Ziehen der unteren rechten Ecke kann die Größe des Fensters verändert werden. Daneben gibt es rechts und unten Rollbalken. Durch Anklicken oder Ziehen dieser kann schnell durch den dargestellten Text geblättert werden. Durch Anklicken eines anderen Fensters kann direkt in dieses gesprungen werden.

### 2. Arbeiten mit der Tastatur

Durch „F10“ wird in das Menü gesprungen. Mit den Kursortasten (links, rechts) kann zwischen den Menüpunkten gewählt werden. Der Menüpunkt wird durch „Enter“ bestätigt. In den dadurch geöffneten Untermenüs kann dann mit den Kursortasten (auf, ab) zwischen den Untermenüpunkten gewählt werden. Durch „Enter“ wird der Untermenüpunkt bestätigt.

Die Menüpunkte können außerdem durch „Alt“+*roter Buchstabe* direkt ausgewählt werden. Die Untermenüpunkte können dann durch Betätigung des hervorgehobenen Buchstabens direkt ausgewählt werden.

Die Untermenüs öffnen Dialogfenster.

Die Mitteilungsdialoge werden durch „Enter“ oder „Esc“ geschlossen.

Die Parameterdialoge werden durch „Enter“ bestätigt bzw. durch „Esc“ abgebrochen.

In den Dialogfenstern kann durch „Tab“ zwischen den Eingabezeilen und sonstigen Einträgen vorwärts und durch „Shift,“+“Tab“ rückwärts gesprungen werden.

In Auswahlfeldern mit runden Klammern kann mit Hilfe der Kursortasten zwischen den Einträgen gewählt werden.

In Auswahlfeldern mit eckigen Klammern wird der Cursor mit Hilfe der Kursortasten auf die zu setzende oder zu löschende Position bewegt. Diese Position wird dann durch „Space“ gesetzt oder zurückgesetzt.

Zwischen den Ansichtfenstern kann neben den Untermenüpunkten „Ansicht-Nächstes“ und „Ansicht-Vorheriges“ durch die Tastenkombination „Alt“+*Fensternummer* direkt gesprungen werden. Die Größe und Position der Ansichtfenster kann durch „Ansicht-Größe/Position“ verändert werden. Hierbei wird durch die Cursortasten die Position und durch „Shift“+*Cursortasten* die Größe des Ansichtfensters verändert.

### 4.3 Einzugebende Werte

Zum Generieren der einzelnen Serien sind Angaben von Parametern nötig. Diese können in den Dialogen zum Generieren der Serien eingegeben werden. Die Bedeutung der Kurzbeschreibungen der Eingabewerte in den Dialogfenstern sowie deren Einheiten sind für die einzelnen Serien aufgeschlüsselt.

#### 1. Grundserie 1D

Varianz:	Varianz der Strömungsgeschwindigkeit in $\frac{m^2}{s^2}$
integrales Zeitmaß:	integrales Zeitmaß der Strömungsgeschwindigkeit in $s$
Punkte je int. Zeitmaß:	Anzahl der Samples pro integralem Zeitmaß der Strömungsgeschwindigkeit in $\frac{1}{\text{int. Zeitmaß}}$
Meßzeit (in int. Zeitmaßen):	Meßzeit als Vielfaches vom integralen Zeitmaß (einheitenlos)
Mittl. Geschwindigkeit:	mittlere Strömungsgeschwindigkeit in $\frac{m}{s}$

#### 2. Grundserie 3D

Kovarianzen:	(Matrix) Kovarianzen der Komponenten der Strömungsgeschwindigkeit in $\frac{m^2}{s^2}$
integrales Zeitmaß:	(Matrix; nur Hauptdiagonale) integrale Zeitmaße der Komponenten der Strömungsgeschwindigkeit in $s$
Punkte je int. Zeitmaß:	(Matrix; nur Wert für $xx$ ) Anzahl der Samples pro integralem Zeitmaß der $x$ -Komponente der Strömungsgeschwindigkeit in $\frac{1}{\text{int. Zeitmaß}}$
Meßzeit (in int. Zeitmaßen):	(Matrix; nur Wert für $xx$ ) Meßzeit als Vielfaches vom integralem Zeitmaß der $x$ -Komponente (einheitenlos)
Mittl. Geschwindigkeit:	(Vektor) mittlere Werte der Komponenten der Strömungsgeschwindigkeit in $\frac{m}{s}$

#### 3. Teilchenserie 1D/3D

Ausdehnung des Meßvolumens

$x$ (Meßrichtung):	Meßvolumenausdehnung in Meßrichtung (Vollachse) in $m$
$y$	(nur 3D) Meßvolumenausdehnung senkrecht zur Meßrichtung und senkrecht zur optischen Achse (Vollachse) in $m$
$z$ (optische Achse):	(nur 3D) Meßvolumenausdehnung in Richtung der optischen Achse (Vollachse) in $m$
mittl. Teilchendichte:	mittlere Teilchendichte in $\frac{1}{\text{int. Zeitmaß}}$
Abhängigkeit der Teilchend.:	(nur 3D) Die Teilchendichte wird als Funktion von einem gewichteten Betrag der Strömungsgeschwindigkeit nach $ \vec{v}_{\text{gew}}(t)  = \sqrt{k_x^2 v_x^2 + k_y^2 v_y^2 + k_z^2 v_z^2}$ berechnet.
$x$ -Richtung:	Abhängigkeit der Teilchendichte von der $x$ -Komponente der Strömungsgeschwindigkeit (einheitenlos)
$y$ -Richtung:	dto. $y$ -Komponente
$z$ -Richtung:	dto. $z$ -Komponente
Anstieg (Betrag):	Anstieg der Teilchendichtefunktion in $\frac{1/\text{int. Zeitmaß}}{m/s}$
Geschwindigkeit beim Sprung:	Geschwindigkeit bei der ein Teilchensprung auftritt in $\frac{m}{s}$
min. Teilchendichte:	minimale Teilchendichte in $\frac{1}{\text{int. Zeitmaß}}$
max. Teilchendichte:	maximale Teilchendichte in $\frac{1}{\text{int. Zeitmaß}}$
Teilchendichtehub:	Änderung der Teilchendichte beim Sprung in $\frac{1}{\text{int. Zeitmaß}}$
Grundserie	
Teilbereich	
von:	Anfangszeit in Vielfachen des integralen Zeitmaßes (einheitenlos)
bis:	Endzeit in Vielfachen des integralen Zeitmaßes (einheitenlos)

#### 4. Prozessorserie 1D/3D

Minimum der Perioden:	Anzahl der zur Meßwertgewinnung nötigen Perioden (einheitenlos)
Maximum der Perioden:	Anzahl der Perioden, nach der der Prozessor zurückgesetzt wird (einheitenlos)
Rücksetzdelaytime:	Totzeit beim Rücksetzen des Prozessors in $s$
Abstand der Interferenzstreifen:	in $m$
Shiftfrequenz:	in $Hz$
Übertragungsdelaytime:	Totzeit zum Übertragen eines Meßwertes an einen Rechner in $s$
Pufferlänge:	Anzahl der zwischenzuspeichernden Meßwerte (einheitenlos)
Abtastfrequenz:	Frequenz mit der ein getakteter Prozessor abgefragt wird oder mit der ein Datensatz reabgetastet wird in $Hz$

## 5 Ausgewählte Ergebnisse

Die durch die Anwendung des Computerprogramms „ENTRY“, welches die in Abschnitt 3 geschilderten Algorithmen realisiert, gewonnenen Erkenntnisse, wurden in zwei Veröffentlichungen

- Fuchs W., Nobach H.: *Auswirkungen der Turbulenz und inhomogener Streuteilchenkonzentration auf das Meßwertensemble eines Laser-Doppler-Anemometers*, Proc. 7. Symposium Maritime Elektronik, Rostock, 1992.
- Fuchs W., Albrecht H., Nobach H., Tropea C., Graham L.J.W.: *Simulation and Experimental Verification of Statistical Bias in Laser Doppler Anemometry including Non-Homogeneous Particle Density*, Proc. 6th Int. Symp. of Appl. of Laser Techn. to Fluid Mechanics, LADOAN, Lisbon, 1992.

zusammengefaßt.

## 6 Zusammenfassung und Ausblick

Anwendungsvoraussetzung für den konkreten Meßprozeß sind a-priori Informationen über den Strömungszustand am Meßort und Detailkenntnisse über die Meßwertgewinnung des Laser-Doppler-Anemometers.

Ausgehend von diesen kann der LDA-Meßprozeß auf einem Rechner simuliert werden.

Dadurch ist es möglich, durch Änderung einzelner Parameter deren Einfluß auf die Meßergebnisse zu untersuchen, und es lassen sich für konkrete Meßanwendungen Voraussagen über die Auswirkungen der Randbedingungen treffen.

Die Rechnersimulation basiert auf mathematischen Modellen. Diese heben bestimmte Eigenschaften hervor und vernachlässigen andere.

Somit wird durch die Simulation stets nur ein Teil der Randbedingungen berücksichtigt.

Die Simulation des LDA-Meßprozesses ist darum durch Verwendung umfassenderer Modelle erweiterbar.

Komplexere Modelle, die den LDA-Meßprozeß präziser beschreiben, führen aber unweigerlich zu höheren Anforderungen an die Leistungsfähigkeit der Rechentechnik. Daher wird die Einbeziehung weiterer Randbedingungen nur schrittweise erfolgen.

## Literatur

- [1] Durst F., Melling A., Whitelaw J. H.: *Principles and practice of laser-Doppler anemometry*. Academic Press, London/New York/San Francisco, 1976.
- [2] Papoulis A.: *Probability, random variables and stochastic processes*. McGraw-Hill Book Comp., New York, 1965.
- [3] Fuchs W.: *Problembezogene Meßtechnologie des Laser-Doppler-Anemometers*. Diss. B, Universität Rostock, 1991.
- [4] Edwards R. V.: *Report of the special panel on statistical particle bias problems in laser anemometry*. Trans. ASME. Journal of Fluids Engineering **109** (1987), 89–93.
- [5] Barnet D. O., Bentley H. T.: *Statistical bias of individual realization laser velocimetry*. Proc. 2nd Int. Workshop on Laser Velocimetry, Purdue, 1974, 428–444.
- [6] Dimotakis P. E.: *Single scattering particle laser Doppler measurements of turbulence*. Proc. AGARD-Conf. No. 193 Symp. on Non-intrusive Instrumentation in Fluid Flow Research, Saint Louis, 1976, paper 10.
- [7] McLaughlin D. K., Tiederman W. G.: *Biasing correction for individual realization of laser anemometer measurements in turbulent flows*. Phys. of Fluids **16** (1973), 2082–2088.
- [8] Hoesel W., Rodi W.: *New biasing elimination method for laser-Doppler velocimeter counter processing*. Rev. Sci. Instrum. **48** (1977), 910–919.
- [9] Nakoa S., Terao Y., Hirata K.: *New method for eliminating the statistical bias in highly turbulent flow measurements*. AIAA Journal **25** (1987), 443–447.
- [10] Box G. E. P., Jenkins G. M.: *Time series analysis: forecasting and control*.
- [11] Abramovitz M., Stegun I. A.: *Handbook of mathematical function*. Dover Publications Inc., New York.



- [12] Tropea C.: *Turbulence-Induced Spectral Bias in Laser Anemometry*. AIAA Journal **25** 2, 306–309.
- [13] Buchhave P., von Benzon H. H., Rasmussen C. N.: *LDA Bias: Comparison of measurement errors from simulated and measured data*. Proc. 5th International Symposium on Application of Laser Anemometry to Fluid Mechanics, Lisbon, 1990.
- [14] Buchhave P.: *Biasing errors in individual particle measurements with the LDA-counter signal processor*. Proc. LDA-Symposium, Copenhagen, 1975.
- [15] Nobach, H.: *Programm zur Simulation von LDA-Signalen*. Praktikumsarbeit, FAU Erlangen, 1991.

## **Erklärung**

Ich erkläre, die Arbeit selbständig und nur unter Verwendung angegebener Literatur angefertigt zu haben.

Rostock-Warnemünde, 14.04.1992

## A1 Beweis für die exponentielle Dichtefunktion der Teilchenabstände

Auf einem Weg  $S_N$  seien  $N$  Teilchen mit gleichverteilten zufälligen Abständen  $s_i$  ( $i = 1 \dots N$ ) vom Ursprung angeordnet. Die Wahrscheinlichkeitsdichtefunktion des Abstandes eines Teilchens vom Ursprung ist dann

$$f_N(s) = \begin{cases} \frac{1}{S_N} & \text{für } 0 \leq s < S_N \\ 0 & \text{sonst} \end{cases} . \quad (\text{A1.1})$$

Die Wahrscheinlichkeit, daß die Anzahl  $k$  der Teilchen auf einer Teilstrecke  $S \leq S_N$  gerade  $K$  beträgt, ist dann

$$P(k = K) = \binom{N}{K} p^K (1-p)^{N-K} \quad (\text{A1.2})$$

mit

$$p = \frac{S}{S_N} .$$

Die Anzahl der Teilchen auf der Teilstrecke  $S$  ist also binomialverteilt. Ist  $N$  groß und  $p$  klein, d. h.  $S \ll S_N$ , so gilt

$$P(k = K) = \frac{(np)^K}{K!} e^{-np} \quad (\text{A1.3})$$

Somit ist die Anzahl von Teilchen in einem kleinen Teilbereich von  $S_N$  poissonverteilt.

Nachzuweisen ist nun, daß durch eine Exponentialverteilung der Abstände von einem Teilchen zum nächsten

1. die Anzahl der Teilchen auf einer begrenzten Strecke  $S$  poissonverteilt ist, und
2. die Abstände  $s_i$  ( $i = 1 \dots K$ ) von  $K$  Teilchen vom Ursprung gleichverteilt sind, wobei die Anzahl  $K$  von Teilchen in  $S$  poissonverteilt zufällig aber bestimmt ist.

1.)

Es seien  $X_1, X_2, \dots$  unabhängige Zufallsvariable mit exponentialverteilten  $X_i$ , d. h.

$$P(X_i < X) = \begin{cases} 1 - e^{-\lambda x} & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} . \quad (\text{A1.4})$$

Die Dichte beträgt dann

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} . \quad (\text{A1.5})$$

Des weiteren sei

$$S_K = \sum_{i=1}^K X_i \quad (\text{A1.6})$$

und  $f_K$  die Dichtefunktion von  $S_K$ . Für  $f_K(x)$  gilt dann

$$f_K(x) = \begin{cases} \lambda^K \frac{x^{K-1}}{(K-1)!} e^{-\lambda x} & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} . \quad (\text{A1.7})$$

Der Beweis erfolgt durch vollständige Induktion:

a)

Laut Definition (A1.6) gilt

$$S_1 = X_1. \quad (\text{A1.8})$$

Da (A1.5) auch die Dichtefunktion von  $X_1$  ist, gilt auch

$$f_1(x) = \begin{cases} \lambda e^{-\lambda x} & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} . \quad (\text{A1.9})$$

(A1.9) ist aber für  $K = 1$  identisch mit (A1.7).

b)

Da  $S_{k+1} = S_k + X_{k+1}$  mit unabhängigen Zufallsvariablen  $S_k$  und  $X_{k+1}$ , so gilt für die Verteilungsfunktion von  $S_{k+1}$

$$f_{k+1}(x) = f_k(x) * f(x) = \int_{-\infty}^{+\infty} f_k(\tau) f(x - \tau) d\tau. \quad (\text{A1.10})$$

Nun sei

$$f_k(x) = \begin{cases} \lambda^k \frac{x^{k-1}}{(k-1)!} e^{-\lambda x} & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases}. \quad (\text{A1.11})$$

Dann ist  $f_k(\tau)f(x - \tau)$  wegen (A1.11) in  $(-\infty; 0)$  und wegen (A1.5) in  $(x; +\infty)$  gleich null.

Daraus folgt

$$\begin{aligned} f_{k+1}(x) &= \int_0^x f_k(\tau) f(x - \tau) d\tau \\ &= \begin{cases} \int_0^x \lambda^k \frac{\tau^{k-1}}{(k-1)!} e^{-\lambda \tau} \lambda e^{-\lambda(x-\tau)} d\tau & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} \\ &= \begin{cases} \frac{\lambda^{k+1}}{k!} e^{-\lambda x} x^k & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} \end{aligned} \quad (\text{A1.12})$$

(A1.12) ist aber identisch mit (A1.7) für  $k + 1 = K$ .

c)

Somit sind die Voraussetzungen für den Schluß nach vollständiger Induktion erfüllt.

(A1.7) gilt somit für beliebiges ganzes  $K \geq 1$  und nach dem Schluß auf eine Allaussage für alle ganzen  $K \geq 1$ .

q. e. d.

Die Wahrscheinlichkeit dafür, daß sich genau  $K$  Teilchen im Intervall  $[0; S]$  befinden, beträgt folglich

$$\begin{aligned} P(k = K) &= P(S_k \leq S < S_{k+1}) \\ &= P(S_k \leq S) - P(S_{k+1} \leq S) \\ &= \int_0^S f_k(x) dx - \int_0^S f_{k+1}(x) dx \\ &= \int_0^S \lambda^k \frac{x^{k-1}}{(k-1)!} e^{-\lambda x} dx - \int_0^S \lambda^{k+1} \frac{x^k}{k!} e^{-\lambda x} dx \end{aligned}$$

$$\begin{aligned}
&= \frac{\lambda^k}{(k-1)!} \int_0^S x^{k+1} e^{-\lambda x} dx - \frac{\lambda^{k+1}}{k!} \int_0^S x^k e^{-\lambda x} dx \\
&= \frac{\lambda^k}{(k-1)!} \int_0^S x^{k+1} e^{-\lambda x} dx + \frac{\lambda^k}{k!} \left[ S^k e^{-\lambda S} - k \int_0^S x^{k+1} e^{-\lambda x} dx \right] \\
&= \frac{\lambda^k}{k!} S^k e^{-\lambda S}. \tag{A1.13}
\end{aligned}$$

Die Anzahl der Teilchen im untersuchten Abschnitt  $S$  ist also poissonverteilt.  
q. e. d.

2.)

Es wird nun die bedingte Wahrscheinlichkeit

$$P(S_1 < R_1, \dots, S_K < R_K | k = K) = \frac{P(\{S_1 < R_1, \dots, S_K < R_K\} \cap \{k = K\})}{P(k = K)} \tag{A1.14}$$

mit

$$0 < R_1 < \dots < R_K < S$$

untersucht.

Es ist

$$\begin{aligned}
I &= P(\{S_1 < R_1, \dots, S_K < R_K\} \cap \{k = K\}) \\
&= P(S_1 < R_1, \dots, S_K < R_K, S_{k+1} \geq S) \\
&= P(X_1 < R_1, \dots, X_1 + \dots + X_K < R_K, X_1 + \dots + X_{K+1} > S)
\end{aligned}$$

Da die Dichte von  $X_1 \dots X_K$  gerade (A1.5) ist, gilt:

$$I = \int \dots \int_B \lambda e^{-\lambda x_1} \dots \lambda e^{-\lambda x_{K+1}} dx_1 \dots dx_{K+1}$$

mit

$$B = \{x_1 \dots x_{K+1} : 0 \leq x_1 < R_1, \dots, 0 \leq x_1 + \dots + x_K < R_K, 0 \leq x_1 \dots x_{K+1} > S\}$$

Läßt man in  $I$  zunächst  $x_1 \dots x_K$  fest und integriert über  $x_{K+1}$  mit

$$S - (x_1 + \dots + x_K) < x_{K+1} < \infty,$$

so erhält man, da

$$\int_{S \perp (x_1 + \dots + x_K)}^{\infty} \lambda e^{\perp \lambda x_{K+1}} dx_{K+1} = e^{\perp \lambda (S \perp (x_1 + \dots + x_K))}$$

für  $I$ :

$$\begin{aligned} I &= \int_{0 \leq x_1 < R_1} \dots \int_{0 \leq x_1 + \dots + x_K < R_K} \lambda e^{\perp \lambda x_1} \dots \lambda e^{\perp \lambda x_K} e^{\perp \lambda (S \perp (x_1 + \dots + x_K))} dx_1 \dots dx_K \\ &= \lambda^K e^{\perp \lambda S} \int_{0 \leq x_1 < R_1} \dots \int_{0 \leq x_1 + \dots + x_K < R_K} dx_1 \dots dx_K \end{aligned} \quad (\text{A1.15})$$

Somit ist nach (A1.14)

$$\begin{aligned} P(S_1 < R_1, \dots, S_K < R_K | k = K) &= \frac{1}{\frac{(\lambda S)^K}{K!} e^{\perp \lambda S}} \lambda^K e^{\perp \lambda S} \int_{0 \leq x_1 < R_1} \dots \int_{0 \leq x_1 + \dots + x_K < R_K} dx_1 \dots dx_K \\ &= \left( \frac{1}{K!} S^K \right)^{\perp 1} \int_{0 \leq x_1 < R_1} \dots \int_{0 \leq x_1 + \dots + x_K < R_K} dx_1 \dots dx_K. \end{aligned} \quad (\text{A1.16})$$

Es seien nun  $r_1 \dots r_k$  unabhängig, und jedes  $r_i$  sei gleichverteilt in  $[0, S]$ , d. h. es besitze die Dichte

$$g_i(r_i) = \begin{cases} \frac{1}{S} & \text{für } 0 \leq r_i < S \\ 0 & \text{sonst} \end{cases}. \quad (\text{A1.17})$$

Wegen der Unabhängigkeit der  $r_1 \dots r_k$  gilt dann für die Dichte

$$g(r_1, \dots, r_k) = f_1(r_1) \dots f_k(r_k). \quad (\text{A1.18})$$

Jetzt seien  $r_{[1]} \dots r_{[k]}$  die geordneten Zufallsvariablen, d. h.  $r_{[1]} = \min(r_1, \dots, r_k)$ ,  $\dots, r_{[k]} = \max(r_1, \dots, r_k)$ . Dadurch zerfällt der Raum  $[0, S]^k$  gerade in  $n!$  gleichgroße Gebiete, wo die Werte der Koordinaten geordnet sind.

Somit gilt für die Dichte

$$h(r_{[1]}, \dots, r_{[k]}) = \left(\frac{1}{k!} S^k\right)^{\perp 1} \quad (\text{A1.19})$$

Jetzt ist

$$P(r_{[1]} < R_1, \dots, r_{[k]} < R_k) = \int_C \dots \int h(r_{[1]}, \dots, r_{[k]}) dr_{[1]} \dots dr_{[k]} \quad (\text{A1.20})$$

wobei

$$C = \left\{ (r_{[1]}, \dots, r_{[k]}) : r_{[i]} < R_i \right\}.$$

Somit gilt

$$P(r_{[1]} < R_1, \dots, r_{[k]} < R_k) = \left(\frac{1}{k!} S^k\right)^{\perp 1} \int_{\substack{0 \leq R_1 \leq \dots \leq R_k \leq S \\ r_{[1]} \leq R_1, \dots, r_{[k]} \leq R_k}} \dots \int dr_{[1]} \dots dr_{[k]}. \quad (\text{A1.21})$$

(A1.21) ist aber für  $k = K$  mit (A1.16) identisch.

q. e. d.



## A2 Testprogramme

### A2.1 VTEST unter MS-Fortran

```
program vtest

integer i,j,n
real a(1:10000)
real sum1,sum2,sum3
real ex,d2x,d3x

print *,"Anzahl der Durchgaenge:"
read *,n
do j=1,n
  sum1=0.
  sum2=0.
  sum3=0.
  do i=1,10000
    a(i)=exp(-i/10000.)
  enddo
  do i=1,10000
    sum1=sum1+a(i)
  enddo
  ex=sum1/10000
  do i=1,10000
    sum2=sum2+(a(i)-ex)**2
    sum3=sum3+(a(i)-ex)**3
  enddo
  d2x=sum2/10000
  d3x=sum3/10000
enddo
print *,ex
print *,d2x
print *,d3x

end
```

## A2.2 VTEST unter Turbo-Pascal

```
program vtest;

var i,j,n:integer;
    a:array[1..10000] of real;
    sum1,sum2,sum3:real;
    ex,d2x,d3x:real;

begin
    write('Anzahl der Durchgaenge:');
    readln(n);
    for j:=1 to n do
        begin
            sum1:=0;
            sum2:=0;
            sum3:=0;
            for i:=1 to 10000 do a[i]:=exp(-i/10000);
            for i:=1 to 10000 do sum1:=sum1+a[i];
            ex:=sum1/10000;
            for i:=1 to 10000 do
                begin
                    sum2:=sum2+(a[i]-ex)*(a[i]-ex);
                    sum3:=sum3+(a[i]-ex)*(a[i]-ex)*(a[i]-ex);
                end;
            d2x:=sum2/10000;
            d3x:=sum3/10000;
        end;
    writeln(ex);
    writeln(d2x);
    writeln(d3x);
end.
```

## A2.3 AR unter MS-Fortran

```
program ar
double precision vmean,turbograd,c1,c2,messzeit,theta,
/vertfkt
character*128 vertstr
integer ran1
real ari(0:9999)
double precision vx,ymax,vxaerf
double precision z1,z2
open (unit=1,file='ar.pmt')
read (1,1) vmean
read (1,1) turbograd
read (1,1) c1
read (1,1) c2
read (1,1) messzeit
read (1,1) theta
read (1,1) vertfkt
read (1,2) vertstr
read (1,3) ran1
close(1)
call lese(vertfkt,vertstr,ari)
call variance(ari,vx,ymax)
call turbo(vmean,turbograd,c1,c2,vxaerf)
call init(ari,vmean,c1,c2,vxaerf,vx,ymax,theta,z1,z2,ran1)
call aufzeich(ari,vmean,turbograd,c1,c2,messzeit,vertfkt,
/vertstr,vxaerf,vx,ymax,z1,z2)
1 format (e15.0)
2 format (a128)
3 format (i15)
end
c
double precision function gamma(x)
double precision x
double precision fac
double precision b1,b2,b3,b4,b5,b6,b7,b8
b1=-.577191652
b2= .988205891
```

```

b3=-.897056937
b4= .918206857
b5=-.756704078
b6= .482199394
b7=-.193527818
b8= .035868343
fac=1.
if (x .gt. 2) then
10     fac=fac*(x-1)
        x=x-1
        if (x .gt. 2) then
            goto 10
        endif
endif
if (x .lt. 1) then
20     fac=fac/x
        x=x+1
        if (x .lt. 1) then
            goto 20
        endif
endif
x=x-1
gamma=1+b1*x+b2*x**2+b3*x**3+b4*x**4
gamma=gamma+b5*x**5+b6*x**6+b7*x**7+b8*x**8
gamma=fac*gamma
return
end

c
subroutine lese(vertfkt,vertstr,ari)
double precision gamma
double precision vertfkt
character*128 vertstr
real ari(0:9999)
integer i
if (vertfkt .eq. 5) then
    open (unit=2,file=vertstr)
endif
do 10 i=0,9999

```

```

        if (vertfkt .eq. 0) then
            ari(i)=exp(-(i/1000.)**2/2)
        elseif (vertfkt .eq. 1) then
            ari(i)=1.
        elseif (vertfkt .eq. 2) then
            ari(i)=1/gamma(1+i/500.d0)
        elseif (vertfkt .eq. 3) then
            ari(i)=(i/200.)*exp(-i/200.)
        elseif (vertfkt .eq. 4) then
            ari(i)=exp(-i/300.)
        elseif (vertfkt .eq. 5) then
            read (2,20) ari(i)
        endif
10    continue
        if (vertfkt .eq. 5) then
            close(2)
        endif
        return
20    format (e15.9)
        end
c
        subroutine variance(ari,vx,ymax)
        real ari(0:9999)
        double precision vx,ymax
        double precision sum0,sum2
        integer i,j
        ymax=0
        sum0=0
        sum2=0
        do 10 i=0,9998
            j=i+1
            sum0=sum0+(ari(i)+ari(j))/2
            sum2=sum2+ari(j)*(i**2/2+2*i/3+1/4)
            sum2=sum2+ari(i)*(i**2/2+i/3+1/12)
            ymax=max(ymax,max(ari(i),ari(j)))
10    continue
        vx=sum2/sum0
        return

```

```

end
c
logical function drin(ari,x,y)
real ari(0:9999)
double precision x,y
double precision f
integer i,j
if ((x .ge. 9999) .or. (x .lt. 0)) then
    drin=.false.
else
    i=int(x)
    j=i+1
    f=ari(j)*(x-i)-ari(i)*(x-j)
    if (y .le. f) then
        drin=.true.
    else
        drin=.false.
    endif
endif
return
end
c
subroutine turbo(vmean,turbograd,c1,c2,vxaerf)
double precision vmean,turbograd,c1,c2,vxaerf
double precision vxzerf
vxzerf=(vmean*turbograd/100)**2
vxaerf=vxzerf*((1-c2)**2-c1**2)*(1+c2)/(1-c2)
return
end
c
subroutine init(ari,vmean,c1,c2,vxaerf,vx,ymax,theta,z1,z2,
/ran1)
double precision vmean,c1,c2,vxaerf,vx,ymax,theta,z1,z2
real ari(0:9999)
integer ran1
double precision out
integer i,thet
thet=int(theta)+1

```

```

z1=0.
z2=0.
call seed(ran1)
do 10 i=1,thet
    call next(ari,vmean,c1,c2,vxaerf,vx,ymax,z1,z2,out)
10 continue
return
end

c
subroutine next(ari,vmean,c1,c2,vxaerf,vx,ymax,z1,z2,out)
double precision vmean,c1,c2,vxaerf,vx,ymax,z1,z2,out
real ari(0:9999)
double precision x,y,a,z
real ran
logical drin
10 call random(ran)
x=10000.d0*ran
call random(ran)
y=ymax*ran
if (.not. drin(ari,x,y)) then
    goto 10
endif
call random(ran)
a=nint(ran)
if (a .lt. .5) then
    a=-1.
endif
a=a*x*sqrt(vxaerf/vx)
z=c1*z1+c2*z2+a
z2=z1
z1=z
out=vmean+z
return
end

c
subroutine aufzeich(ari,vmean,turbograd,c1,c2,messzeit,
/vertfkt,vertstr,vxaerf,vx,ymax,z1,z2)
double precision vmean,turbograd,c1,c2,messzeit,vertfkt,

```

```

/vxaerf,vx,ymax,z1,z2
real ari(0:9999)
character*128 vertstr
double precision out,kout(0:999)
double precision zeit
double precision sum0,sum1,sum2
double precision vmeanerr,turboerr,vxzerr
integer k,l
zeit=messzeit*1000
sum0=0.
sum1=0.
sum2=0.
open(unit=3,file='temp.tmp')
10 if (zeit .gt. 0) then
        call next(ari,vmean,c1,c2,vxaerf,vx,ymax,z1,z2,out)
        sum0=sum0+1
        sum1=sum1+out
        write(3,40) out
endif
if (sum0 .lt. zeit) then
        goto 10
endif
vmeanerr=sum1/sum0
rewind(3)
sum0=0
20 if (zeit .gt. 0) then
        read(3,40) out
        sum0=sum0+1
        sum2=sum2+(out-vmeanerr)**2
endif
if (sum0 .lt. zeit) then
        goto 20
endif
vxzerr=sum2/sum0
turboerr=100*sqrt(vxzerr)/vmeanerr
open(unit=4,file='temp.roh')
write(4,50) 'Grundserie           '
write(4,50) '

```



```

write(4,50) 'Einstellungen          '
write(4,50) '-----          '
write(4,60) 'mittl.Geschwindigkeit : ',vmean
write(4,60) 'Turbulenzgrad ( % )   : ',turbograd
write(4,60) 'c1                    : ',c1
write(4,60) 'c2                    : ',c2
write(4,60) 'Messzeit ( / 1000 )   : ',messzeit
write(4,60) 'Verteilungsfunktion   : ',vertfkt
write(4,70) 'externe Verteilungsfkt. : ',vertstr
write(4,50) '
write(4,50) 'Resultate            '
write(4,50) '-----          '
write(4,60) 'mittl.Geschwindigkeit : ',vmeanerr
write(4,60) 'Varianz der Geschwindigkeit : ',vxzerr
write(4,60) 'Turbulenzgrad         : ',turboerr
write(4,50) '
rewind(3)
sum0=0
k=0
30  if (zeit .gt. 0) then
      read(3,40) kout(k)
      sum0=sum0+1
      k=k+1
      if (k .eq. 1000) then
          do 35 k=0,999
              write(4,40) kout(k)
35          continue
          k=0
      endif
  endif
  if (sum0 .lt. zeit) then
      goto 30
  endif
  if (k .ne. 0) then
      l=k-1
      do 37 k=0,l
          write(4,40) kout(k)
37          continue

```

```
endif
close(3)
close(4)
open(unit=5,file='ar.pmt')
write(5,40) vmeanerr
write(5,40) turboerr
close(5)
return
40 format(e15.9)
50 format(a30)
60 format(a30,e15.9)
70 format(a30,a128)
end
```

## A2.4 AR unter Turbo-Pascal

```
program ar;

type tari=array[0..9999] of real;
    pari=^tari;

var vmean,turbograd,c1,c2,messzeit,theta,vertfkt:real;
    vertstr:string;
    ran1:integer;
    ari:pari;
    vx,ymax,vxaerf:real;
    z1,z2:real;
    vmeanerr,turboerr:real;
    f:text;

function gamma(x:real):real;
var fac:real;
const b1=-0.577191652;
      b2= 0.988205891;
      b3=-0.897056937;
      b4= 0.918206857;
      b5=-0.756704078;
      b6= 0.482199394;
      b7=-0.193527818;
      b8= 0.035868343;
begin
    fac:=1;
    while x>2 do
    begin
        fac:=fac*(x-1);
        x:=x-1;
    end;
    while x<1 do
    begin
        fac:=fac/x;
        x:=x+1;
    end;
end;
```

```

    end;
    x:=x-1;
    gamma:=(1+x*(b1+x*(b2+x*(b3+x*(b4+x*(b5+x*(b6+x*(b7+x*b8
        )))))))*fac;
end;

procedure lese(vertfkt:real;vertstr:string;var ari:pari);
var i:integer;
    f:text;
begin
    ari:=new(pari);
    if vertfkt=5 then
        begin
            assign(f,vertstr);
            reset(f);
        end;
    for i:=0 to 9999 do
        begin
            if vertfkt=0 then ari^[i]:=exp(-(i/1000)*(i/1000)/2);
            if vertfkt=1 then ari^[i]:=1;
            if vertfkt=2 then ari^[i]:=1/gamma(1+i/500);
            if vertfkt=3 then ari^[i]:=(i/200)*exp(-i/200);
            if vertfkt=4 then ari^[i]:=exp(-i/300);
            if vertfkt=5 then readln(f,ari^[i]);
        end;
    if vertfkt=5 then close(f)
end;

procedure variance(ari:pari;var vx,ymax:real);
var sum0,sum2:real;
    i,j:longint;
begin
    ymax:=0;
    sum0:=0;
    sum2:=0;
    for i:=0 to 9998 do
        begin
            j:=i+1;

```

```

        sum0:=sum0+(ari^[i]+ari^[j])/2;
        sum2:=sum2+ari^[j]*(i*i/2+2*i/3+1/4)
            +ari^[i]*(i*i/2+i/3+1/12);
        if ari^[i]>ymax then ymax:=ari^[i];
        if ari^[j]>ymax then ymax:=ari^[j];
    end;
    vx:=sum2/sum0;
end;

function drin(ari:pari;x,y:real):boolean;
var f:real;
    i,j:integer;
begin
    if ((x>=9999) or (x<=0)) then drin:=false
    else
        begin
            i:=trunc(x);
            j:=i+1;
            f:=ari^[j]*(x-i)-ari^[i]*(x-j);
            drin:=(y<=f);
        end;
    end;
end;

procedure turbo(vmean,turbograd,c1,c2:real;var vxaerf:real);
var vxzerf:real;
begin
    vxzerf:=(vmean*turbograd/100)*(vmean*turbograd/100);
    vxaerf:=vxzerf*((1-c2)*(1-c2)-c1*c1)*(1+c2)/(1-c2);
end;

procedure next(ari:pari;vmean,c1,c2,vxaerf,vx,ymax:real;
    var z1,z2,out:real);
var x,y,z,a,ran:real;
begin
    repeat
        ran:=random;
        x:=10000*ran;
        ran:=random;

```

```

        y:=ymax*ran
    until drin(ari,x,y);
    ran:=random;
    a:=(ran-0.5)/abs(ran-0.5)*x*sqrt(vxaerf/vx);
    z:=c1*z1+c2*z2+a;
    z2:=z1;
    z1:=z;
    out:=vmean+z;
end;

procedure init(ari:pari;vmean,c1,c2,vxaerf,vx,ymax,theta:real;
               var z1,z2:real;ran1:integer);
var out:real;
    thet,i:integer;
begin
    thet:=trunc(theta)+1;
    z1:=0;
    z2:=0;
    randomize;
    for i:=1 to thet do
        next(ari,vmean,c1,c2,vxaerf,vx,ymax,z1,z2,out)
    end;

procedure aufzeich(ari:pari;vmean,turbograd,c1,c2,messzeit,
                  vertfkt:real;vertstr:string;vxaerf,vx,ymax,
                  z1,z2:real);
var out:real;
    kout:array[0..999] of real;
    zeit,sum0,sum1,sum2:real;
    vmeanerr,turboerr,vxzerr:real;
    k,l:integer;
    fs,ft:text;
begin
    zeit:=messzeit*1000;
    sum0:=0;
    sum1:=0;
    sum2:=0;
    assign(fs,'temp.tmp');

```

```

rewrite(fs);
repeat
  if zeit>0 then
  begin
    next(ari,vmean,c1,c2,vxaerf,vx,ymax,z1,z2,out);
    sum0:=sum0+1;
    sum1:=sum1+out;
    writeln(fs,out:15);
  end;
until (sum0>=zeit);
vmeanerr:=sum1/sum0;
close(fs);
reset(fs);
sum0:=0;
repeat
  if zeit>0 then
  begin
    readln(fs,out);
    sum0:=sum0+1;
    sum2:=sum2+(out-vmeanerr)*(out-vmeanerr);
  end
until sum0>=zeit;
close(fs);
vxzerr:=sum2/sum0;
turboerr:=100*sqrt(vxzerr)/vmeanerr;
assign(ft,'temp.roh');
rewrite(ft);
writeln(ft,'Grundserie           ');
writeln(ft,'                               ');
writeln(ft,'Einstellungen                 ');
writeln(ft,'-----                         ');
writeln(ft,'mittl.Geschwindigkeit         : ',vmean:15);
writeln(ft,'Turbulenzgrad ( % )           : ',turbograd:15);
writeln(ft,'c1                             : ',c1:15);
writeln(ft,'c2                             : ',c2:15);
writeln(ft,'Messzeit ( / 1000 )           : ',messzeit:15);
writeln(ft,'Verteilungsfunktion           : ',vertfkt:15);
writeln(ft,'externe Verteilungsfkt.      : ',vertstr:15);

```

```

writeln(ft,'');
writeln(ft,'Resultate');
writeln(ft,'-----');
writeln(ft,'mittl.Geschwindigkeit : ',vmeanerr:15);
writeln(ft,'Varianz der Geschwindigkeit : ',vxzerr:15);
writeln(ft,'Turbulenzgrad : ',turboerr:15);
writeln(ft,'');
reset(fs);
sum0:=0;
k:=0;
repeat
    if zeit>0 then
        begin
            readln(fs,kout[k]);
            sum0:=sum0+1;
            k:=k+1;
            if k=1000 then
                begin
                    for k:=0 to 999 do writeln(ft,kout[k]:15);
                    k:=0;
                end;
            end;
        until(sum0>=zeit);
        if k<>0 then
            begin
                l:=k-1;
                for k:=0 to l do writeln(ft,kout[k]:15);
            end;
        close(fs);
        close(ft);
        assign(ft,'ar.pmt');
        rewrite(ft);
        writeln(ft,vmeanerr);
        writeln(ft,turboerr);
        close(ft)
end;

begin

```



```
assign(f,'ar.pmt');
reset(f);
readln(f,vmean);
readln(f,turbograd);
readln(f,c1);
readln(f,c2);
readln(f,messzeit);
readln(f,theta);
readln(f,vertfkt);
readln(f,vertstr);
readln(f,ran1);
close(f);
lese(vertfkt,vertstr,ari);
variance(ari,vx,ymax);
turbo(vmean,turbograd,c1,c2,vxaerf);
init(ari,vmean,c1,c2,vxaerf,vx,ymax,theta,z1,z2,ran1);
aufzeich(ari,vmean,turbograd,c1,c2,messzeit,vertfkt,vertstr,
          vxaerf,vx,ymax,z1,z2);
end.
```