

Statusbericht des Projektes bvif02

Rupert Klein^{1,2} und Markus Uhlmann¹

¹Potsdam Institut für Klimafolgenforschung, D-14412 Potsdam

²FB Mathematik & Informatik, Freie Universität Berlin, Konrad Zuse Zentrum für Informationstechnik, Takustr. 7, D-14195 Berlin

(Januar 2001)

1. Einleitung

Das Projekt `bvif02` dient der Realisierung der numerischen Aufgaben welche im von der DFG geförderten Programm “Kleinskalige Instabilitäten als Bausteine der turbulenten Energiekaskade” (Kennung KL 611/10) gestellt wurden. Dabei handelt es sich in erster Linie um die Erzeugung von zeitlich und räumlich sehr hoch aufgelösten Felddaten in turbulenten Strömungen *via* direkte numerische Simulation (DNS). Diese Daten werden in Nachbearbeitungsschritten digitalen Filteroperationen unterzogen. Im ursprünglichen Antrag (Klein & Uhlmann 2000) wurden die wissenschaftlichen Zielsetzungen skizziert, weitere Details sind dem DFG-Antrag auf Gewährung einer Sachbeihilfe (Klein 1998) zu entnehmen. Fortschritte des Projektes in Form von Zwischenberichten und Animationen sind öffentlich zugänglich unter der URL: www.pik-potsdam.de/~uhlmann/projects.html.

Mit Datum des 18. Februar 2000 hat der Zulassungsausschuß für das ZIB für den Zeitraum 2/2000 – 1/2001 insgesamt 8000 ZPL von beantragten 20000 ZPL bewilligt. Am 30. November 2000 wurden aufgrund eines Erweiterungsantrages zusätzliche 2500 ZPL für das Quartal 1/2001 bereitgestellt.

Im vorliegenden Antrag weisen wir die bisherige Verwendung von Ressourcen für die Simulation und Analyse der verschiedenen Strömungsfälle nach (§2-4). Außerdem detaillieren wir in §3.1-3.3 unser Verfahren zur Simulation homogen-isotroper Turbulenz, da dieser Fall im ursprünglichen Antrag nicht erwähnt wurde. In §5 präsentieren wir schließlich den geplanten weiteren Verbrauch, was uns zur Beantragung eines Kontingentes von 10100 ZPL für den Zeitraum 2/2001 – 1/2002 führt.

2. Simulation ebener Kanalströmung

Der ursprüngliche Projektantrag (Klein & Uhlmann 2000) beschreibt ausführlich den DNS *code* für ebene Kanalströmung: Modell, numerische Methoden, paralleler Lösungsalgorithmus. In der Folge wurden u.a. mit Hilfe von Herrn K. Ketelsen von SGI diverse T3E-spezifische Optimierungsschritte durchgeführt, welche insbesondere die Einzelprozessorleistung verbesserten (Fouriertransformation, Tridiagonallöser, einzelne kritische Schleifen) aber auch eine gute Skalierung für große Prozessoranzahlen ermöglichen (asynchrone Ausführung der Datentransponierung). Die Summe dieser Modifikationen führte zu einer Beschleunigung des Programmes um etwa einen Faktor 2 bei relevanten Problemen. Es sollte erwähnt werden, daß bei der größten realisierten Problemdimension ($600 \times 385 \times 600$ Freiheitsgrade) jetzt ein *speed-up* von nahezu 2 erreicht wird wenn 256 PEs gegenüber 128 PEs verwendet werden.

Tabelle 1 gibt eine Übersicht über die bisher durchgeführten Rechnungen mit dem

case	PEs	N_i	Re_τ	t_{init}	$t_{produce}$	$\sum t$	n_{snap}
kim590	128	$384 \times 257 \times 384$	590	8×150 ZPL	5×24 ZPL	1320 ZPL	249
kim590fi	200	$600 \times 385 \times 600$	590	11×150 ZPL	6×100 ZPL	2250 ZPL	240
kmmhires	200	$600 \times 385 \times 600$	200	16×150 ZPL	10×113 ZPL	3530 ZPL	250
						7100 ZPL	

TABELLE 1. Details der bisher auf **berte** durchgeführten Simulationen ebener Kanalströmung. N_i bezeichnet die Anzahl der numerischen Fourier/Tschebyscheff-Moden, Re_τ ist die relevante Reynoldszahl, t_{init} die notwendige Rechenzeit zum Erreichen eines statistischen Gleichgewichtes, $t_{produce}$ die Rechenzeit zur Generation einer zeitlich hochaufgelösten Serie von n_{snap} *snapshots*.

optimierten *code*. Die Erzeugung eines vollturbulenten Zustandes in statistischem Equilibrium (t_{init}) beansprucht den Hauptanteil der Rechenzeit. Dieser Anteil schließt die notwendige Integrationszeit für die Akkumulation von konvergierter Statistik der Momente zweiter Ordnung ein. Dabei ist zu bemerken, daß der Anteil von t_{init} dennoch weitaus geringer ausfällt als die 5/6, die in Klein & Uhlmann (2000) angenommen worden waren, da nicht – wie ursprünglich geplant – der laminar-turbulente Umschlag berechnet wurde, sondern jede Serie mit interpolierten Feldern von benachbarten Parameterpunkten initialisiert wurde. Auf diese Art liegen schon von Beginn “realistisch” korrelierte Lösungsanteile vor, und es stellt sich heraus, daß in einem kürzerem Zeitraum ein Gleichgewichtszustand erreicht wird. Die Details dieser Methode sind in Uhlmann (2000b) dargestellt worden.

Die eigentliche Produktion der nutzbaren Datenfelder wird in Uhlmann (2000a) ausführlich beschrieben. In o.g. URL sind auch Visualisierungen der Turbulenzstruktur abrufbar.

Die Datenproduktion zum Problem der ebenen Kanalströmung ist mit den vorliegenden den ca. 750 Feldern abgeschlossen.

3. Simulation homogen-isotroper Strömung

3.1. Konfiguration

Der zweite Strömungstyp, welcher im zugrunde liegenden DFG-Projekt untersucht wird, entwickelt sich in einem dreifach periodischen Integrationsgebiet. Die resultierende turbulente Strömung ist statistisch gesehen homogen und isotrop in der Ortsvariablen. Nach analytischer Elimination des Drucktermes lauten hierfür die Impulsgleichungen im Fourierraum (mit $\hat{\cdot}$ versehene Größen sind Fourierkoeffizienten):

$$\partial_t \hat{u}_i + \nu k^2 \hat{u}_i = \widehat{(u_i u_j)}_{,j} \left(\delta_{il} - \frac{k_i k_l}{k^2} \right) + \hat{f}(k) \quad , \quad (3.1)$$

wobei u_i die Geschwindigkeitskomponenten bezeichnet, ν die Viskosität, k den Wellenzahlvektor und f eine arbiträre Volumenkraft. Letzterer Term hat die Aufgabe, die Strömung—welche keine mittlere Geschwindigkeit besitzt—künstlich mit Energie zu versorgen, um das Erreichen eines etablierten Zustandes zu ermöglichen. Dies wird dadurch erreicht, dass ständig in diejenigen Fouriermoden, welche in einem schmalen Frequenzband um einen vorgegebenen Wellenzahlmodulus k_p liegen, Energie injiziert wird (entsprechend der gesamten Dissipation). Die genaue Form der “Einspeisung” ist weitgehend *ad hoc* (siehe auch Avelius 1999).

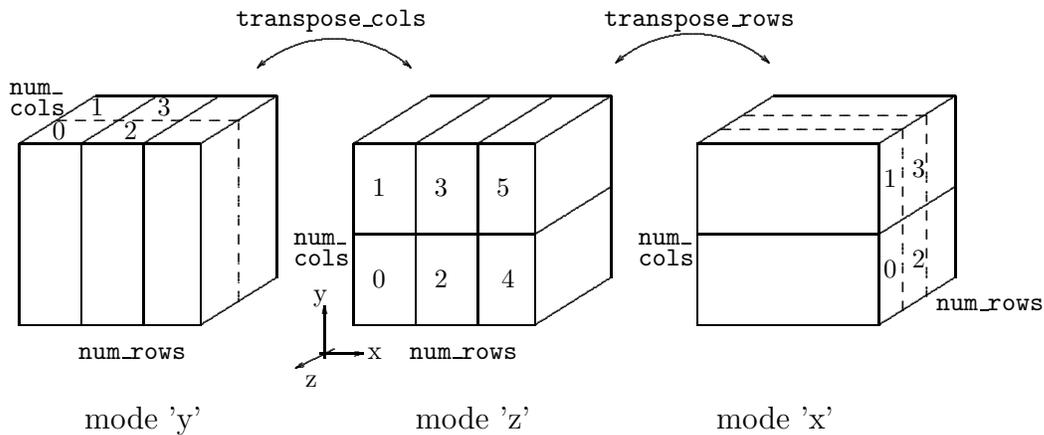


ABBILDUNG 1. Aufteilung der dreidimensionalen ortsabhängigen Daten auf die beteiligten Prozessoreinheiten (angedeutet durch Ziffern 0-5 für den Fall von 6 Prozessoren). Je nach gegenwärtigem “Modus” (‘x’, ‘y’ oder ‘z’) befindet sich eine volle Datendimension im lokalen Speicher.

3.2. Numerische Methode

Das Simulationsprogramm ist uns von A. Wray, NASA Ames, zur Verfügung gestellten worden. Darin findet direkt obige Formulierung (3.1) in primitiven Variablen Verwendung. Es handelt sich hierbei um ein Fourier-Pseudospektralverfahren, wobei *de-aliasing* mittels der 2/3-Regel durchgeführt wird. Die Zeititeration erfolgt durch ein Runge-Kutta Schema dritter Ordnung. Die durch lineare Analyse ermittelte, maximal erlaubte CFL Zahl beträgt in diesem Fall 2.

3.3. Parallelisierung

Dieses Rechenprogramm wurde von uns für die Nutzung auf der T3E angepaßt und schrittweise optimiert. Die Parallelität wird *via* MPI unter FORTRAN durch ein *pencil*-Datenmodell realisiert, d.h. es wird zunächst die Anzahl der Prozessoren möglichst quadratisch in `numop=num_cols*num_rows` faktorisiert und es werden dann in zwei Koordinatenrichtungen die Daten über dieses “Gitter” verteilt. Es ist also jeweils nur in einer Ortsrichtung die volle Vektorlänge der Daten lokal (d.h. ohne Kommunikation) zugänglich. Auf diese Weise wird einerseits mehr Kommunikation zwischen den Prozessoreinheiten notwendig, jedoch wird andererseits sichergestellt, daß lokale Speicherzugriffe ausschließlich auf benachbarte Adressen erfolgen.

Abbildung 1 zeigt schematisch die Aufteilung der Daten auf die zur Verfügung stehenden Prozessoren je nach “Modus” (‘x’, ‘y’ oder ‘z’). Dabei wird zwischen den einzelnen Modi durch geeignete Transponierung entweder der “Spalten” (`transpose_cols`) oder der “Zeilen” (`transpose_rows`)—ähnlich den Operationen eines Rubik’s cube—gewechselt. Dieses Umschichten der Daten erfolgt durch nicht blockierende Kommunikation und doppelte temporäre Speicher. Auf diese Weise kann das Ver- und Entpacken der darauffolgenden Nachrichten schon während der aktuellen Kommunikationszeit erfolgen. Der nachstehende Algorithmus erläutert—als Beispiel—den Wechsel eines Feldes `u(mzr-num_rows,my,mx)` vom ‘z’ in den ‘x’ Modus durch die Funktion `transpose_rows`. Dabei sei vorausgesetzt, daß in einem Initialisierungsschritt die paarweisen Kommunikationspartner ermittelt worden sind und sich in den (lokalen) Zeigern `to_row(1:num_rows-1)` und `from_row(1:num_rows-1)` befinden. Diese Paare sind so errechnet, daß sich die *messages* nicht kreuzen.

```

1:  $ib \leftarrow 0$ 
2:  $stage \leftarrow 1$ 
3:  $stages(0:1) \leftarrow 0$ 
4:  $msg\_size \leftarrow m_{zr} \cdot m_x \cdot m_y$ 
5:  $row\_out(1:msg\_size, ib) \leftarrow$ 
6:      $u(1+m_{zr} \cdot to\_row(stage) : m_{zr} \cdot (to\_row(stage)+1), :, :)$ 
7: repeat
8:    $stages(ib) \leftarrow stage$ 
9:   if ( $stage < num\_rows$ ) then
10:     $destination \leftarrow my\_col + to\_row(stage) \cdot num\_cols$ 
11:    call MPI_IRecv( $row\_in(ib)$ ,  $msg\_size$ , ...,  $recv\_request$ )
12:    call MPI_Isend( $row\_out(ib)$ ,  $msg\_size$ ,  $destination$ , ...,  $send\_request$ )
13:  end if
14:   $ib \leftarrow 1 - ib$  {toggle between buffers}
15:   $stage \leftarrow stage + 1$ 
16:  if ( $stage < num\_rows$ ) then
17:     $row\_out(1:msg\_size, ib) \leftarrow$ 
18:         $u(1+m_{zr} \cdot to\_row(stage) : m_{zr} \cdot (to\_row(stage)+1), :, :)$ 
19:  end if
20:  if ( $stages(ib) > 0$ ) then
21:     $u(1+m_{zr} \cdot from\_row(stage) : m_{zr} \cdot (to\_row(stage)+1), :, :) \leftarrow$ 
22:         $row\_in(1:msg\_size, ib)$ 
23:  end if
24:  if ( $stage < num\_rows$ ) then
25:    call MPI_WAIT( $recv\_request$ , ...)
26:    call MPI_WAIT( $send\_request$ , ...)
27:  end if
28: until ( $stage > num\_rows$ )

```

Hier ist anzumerken, daß die interne Transponierung, also der Tausch der Laufindizes des lokalen Feldes, gesondert durch einen expliziten Kopiervorgang erfolgt.

Die Sequenz der verschiedenen Kommunikationsschritte während der zeitlichen Integration sei im folgenden angedeutet.

```

1: repeat
2:   Operationen in  $z$ -Richtung
3:    $\mapsto$  transpose_rows
4:   Operationen in  $x$ -Richtung
5:    $\mapsto$  transpose_rows
6:   mehr Operationen in  $z$ -Richtung
7:    $\mapsto$  transpose_cols
8:   Operationen in  $y$ -Richtung; Zeitschritt
9:    $\mapsto$  transpose_cols
10: until (Ende der Zeitintegration)

```

3.4. Performance

Die Tabelle 2 gibt Aufschluß über die Performance des Simulationsprogrammes auf der CRAY T3E. Nach unserer Optimierung wird eine globale Rate von 73 Mflop pro PE erreicht. Die Fouriertransformationen liefern ca. 100 (*complex-to-complex*) bzw. 160 Mflop (*real-to-complex*), wobei das zweifache Kopieren der Daten inbegriffen ist. Diese Werte liegen etwas höher als vergleichbare Resultate des Programmes für die Kanalströmung.

Performance output created by fort_pat			T3E-	0	8 PEs	
			Performance / PE			
Region_name	Av. Time [sec]	Total Gflop	average Mflop	Max	Min	Number calls
c-to-c fft	47.64	0.821	102.57	102.96	102.13	25603
r-to-c fft	35.52	1.314	164.24	164.36	163.79	30721
complete code	171.17	0.585	73.11	73.54	72.93	1

TABELLE 2. Performance des Simulationsprogrammes für homogen-isotrope Strömung auf der CRAY T3E. Die Problemdimension ist $N = 128$, es wurden 8 PE's mit 450 MHz verwendet und 10 Zeitschritte ausgeführt.

case	PEs	N	k_p	$k_{max}\eta$	Re_λ	L_{int}/L_{box}	t_{wall}	t_{phys}/t_{kol0}	t_{phys}/T_0
512_a	256	512	8	2	60	0.04	5×187 ZPL	100	6
512_b	256	512	2	2.5	150	0.10	7×187 ZPL	30	2.5
							2244 ZPL		

TABELLE 3. Parameter der bisherigen Simulationen homogen-isotroper Turbulenz: Anzahl der verwendeten Prozessoreinheiten PEs; Anzahl der Fouriermoden in jeder Koordinatenrichtung N ; grossskalige Wellenzahl k_p , in deren Umgebung künstlich Energie mittels Volumenkraft hinzugefügt wird; Auflösung der kleinsten Skalen $k_{max}\eta$, wobei η das Kolmogoroff'sche Längenmaß ist; Taylor'sche Reynoldszahl Re_λ ; Größe des Integrationsgebietes im Verhältnis zum integralen Längenmaß L_{int}/L_{box} ; Rechenzeit t_{wall} ; physikalische Zeit t_{phys}/t_{kol0} in Kolmogoroff'schen Zeiteinheiten des Anfangsfeldes; physikalische Zeit t_{phys}/T_0 in Zeiteinheiten der Großen Skalen des Anfangsfeldes. Bei der angegebenen Rechenzeit ist zu beachten, daß diese Läufe teilweise vor den letzten in §3.3 beschriebenen Optimierungsschritten ausgeführt wurden.

3.5. Bisheriger Verbrauch

Der bisherige Verbrauch von Rechenzeit für die Simulation homogen-isotroper Strömung umfaßt neben der Entwicklungszeit und einigen Testrechnungen kleineren Umfangs (siehe erste Ergebnisse in Uhlmann 2000c; weitere Visualisierung und Animationen sind unter der gleichen URL zugänglich) den Beginn zweier Simulationsreihen 512_a, 512_b. Die Tabelle 3 zeigt die jeweils relevanten physikalischen und numerischen Parameter. Dabei sollte beachtet werden, daß die lineare Dimension N —im Gegensatz zum Fall der Kanalströmung—die Anzahl der nicht mit *aliasing* behafteten Moden angibt, die tatsächliche Datenlänge also um einen Faktor 3/2 höher liegt.

Die Rechnungen zum Fall mit höherer Reynoldszahl (512_b) befinden sich zum gegenwärtigen Zeitpunkt noch in der Phase des Aufbaus eines realistischen turbulenten Zustandes.

4. Wavelet-basierte Filteroperationen

Zwecks Filterung der Daten bei der Nachbearbeitung haben wir uns auf Wavelet-basierte Methoden konzentriert, da diese gleichzeitig Zugang zu Skalen- und Ortsinformationen bieten können. Dies ist ein wichtiger Punkt in unserem wissenschaftlichen Programm: Detektion und Extraktion von lokalisierten, kleinskaligen Lösungsanteilen

case	t_{tot}	t_{done}	T/t_{kol}	t_{wall}/t_{kol}	$\sum t_{wall}$
512_a	$7T + 500 t_{kol}$	8T	17	8.5 ZPL	4395 ZPL
512_b		2.5T	40	10.1 ZPL	6868 ZPL
					11263 ZPL

TABELLE 4. Details zu den für 2/2001–1/2002 geplanten Simulationen homogen-isotroper Turbulenz: gesamte physikalische Simulationszeit t_{tot} ; bisher simulierte physikalische Zeit t_{done} (Übertrag aus Tabelle 3); Verhältnis des grossskaligen zum kleinskaligen Zeitmaß T/t_{kol} ; Simulationsdauer pro Zeitmaß t_{wall}/t_{kol} bei Verwendung von 256 PEs; verbleibende Simulationsdauer $\sum t_{wall}$. Es ist zu beachten, daß die Effizienz des Simulationsprogrammes durch weitere Optimierung nicht mit derjenigen der Tabelle 3 übereinstimmt.

im turbulenten Feld. Unsere bisherige Arbeit umfaßt daher die Implementierung einer diskret-orthogonalen Wavelettransformation auf Multiprozessormaschinen mit *distributed memory*. Dieses Programm ist für die Ausführung auf der T3E angepaßt worden. Uhlmann (2000d) gibt einen genauen Einblick in den Algorithmus sowie einige beispielhafte Ergebnisse. Auch zu diesem Teilbereich sind unter der URL weitere Animationen und Darstellungen zur Filterung und Kompression der Strömungsfelder erhältlich.

Obwohl die Motiviation für die Parallelisierung im Speicheraufwand begründet war, hat es sich herausgestellt, daß auch die für die Transformation notwendige Rechenzeit gut skaliert.

5. Plan der weiteren Rechnungen

5.1. Simulation homogen-isotroper Strömung

Wir planen, weitere Rechnungen mit 512^3 Freiheitsgraden und den Reynoldszahlen von $Re_\lambda = \{80, 150\}$. Dazu sollen die oben erwähnten Fälle 512_a, 512_b in folgender Weise fortgeführt werden. Nach Beendigung der initialen Phase ($t_{init} + t_{stat}$) werden realistische Strömungsfelder in statistischem Gleichgewicht vorliegen. Im Anschluß werden wir Datenfelder über einen Zeitraum von $t_{produce}$ produzieren, und zwar sowohl (i) bei weiterer Injektion von Energie in die großen Skalen, als auch (ii) nach Abschalten der Energiezufuhr (abklingende Turbulenz) und einer kurzen Anpassungszeit t_{adjust} . Somit würde sich in beiden Fällen eine Gesamtzeit von

$$t_{tot} = t_{init} + t_{stat} + t_{adjust} + 2 t_{produce} \quad (5.1)$$

ergeben.

Basierend auf Erfahrungswerten (siehe auch Jiménez *et al.* 1993) schätzen wir $t_{init} + t_{stat} \approx 6T$, wobei T ein charakteristisches Zeitmaß der großen Skalen ist. Auf ähnliche Weise gelangen wir zu $t_{adjust} \approx T$. Datenfelder sollen jeweils über ein Zeitfenster von $t_{produce} \approx 250 t_{kol}$ generiert werden (t_{kol} ist ein Zeitmaß der kleinsten relevanten Skalen).

Tabelle 4 zeigt die Details der Berechnung von Formel (5.1) mittels der Werte für das Verhältnis der beiden Zeitmaße und unter Berücksichtigung der notwendigen Rechenzeit pro Zeitmaß bei Ausführung auf 256 PEs. Insgesamt stehen damit 11263 ZPL aus.

5.2. Wavelet-basierte Filteroperationen

Die geplanten Filteroperationen müssen aufgrund des Datenvolumens ebenfalls auf **berte** ausgeführt werden. Die notwendige Rechenzeit ist gering. Ein Beispiel: die diskrete Wavelettransformation und inverse -transformation eines 512^3 Feldes kostet 0.025 ZPL bei

Ausführung auf 128 PEs. Wir werden eine Vielzahl $\mathcal{O}(1000)$ von Feldern bearbeiten und dennoch bleibt die gesamte zu erwartende Rechenzeit unter dieser Rubrik klein.

6. Beantragtes Kontingent für die Fortführung

Die Summe aus bisher verbrauchter und geplanter Rechenzeit beträgt 20607 ZPL. Nach Abzug der bisher gewährten 10500 ZPL beantragen wir demnach ein neues Kontingent von

10100 ZPL

für das Projektjahr 2/2001–1/2002.

REFERENZEN

- AVELIUS, K. 1999 Random forcing of three-dimensional homogeneous turbulence. *Phys. Fluids* **11** (7), 1880–1889.
- JIMÉNEZ, J., WRAY, A., SAFFMAN, P. & ROGALLO, R. 1993 The structure of intense vorticity in isotropic turbulence. *J. Fluid Mech.* **255**, 65–90.
- KLEIN, R. 1998 Kleinskalige Instabilitäten als Bausteine der turbulenten Energiekaskade. DFG-Antrag auf Gewährung einer Sachbeihilfe, Projekt KL 611/10 (in German).
- KLEIN, R. & UHLMANN, M. 2000 Projektbeschreibung zum Großprojektantrag für den Parallelrechner T3E des ZIB (in German). available under www.pik-potsdam.de/~uhlmann/-projects.html.
- UHLMANN, M. 2000*a* Generation of a temporally well-resolved sequence of snapshots of the flow-field in turbulent plane channel flow. Intermediate Report, available under www.pik-potsdam.de/~uhlmann/projects.html, Potsdam Institute for Climate Impact Research, Potsdam, Germany.
- UHLMANN, M. 2000*b* Generation of initial fields for channel flow investigation. Intermediate Report, available under www.pik-potsdam.de/~uhlmann/projects.html, Potsdam Institute for Climate Impact Research, Potsdam, Germany.
- UHLMANN, M. 2000*c* Homogeneous isotropic turbulence: Parameters. Intermediate Report, available under www.pik-potsdam.de/~uhlmann/projects.html, Potsdam Institute for Climate Impact Research, Potsdam, Germany.
- UHLMANN, M. 2000*d* A parallel algorithm for the discrete orthogonal wavelet transform. Technical Note No. 68, available under www.pik-potsdam.de/~uhlmann/projects.html, Potsdam Institute for Climate Impact Research, Potsdam, Germany.