STUDIES OF TURBULENCE STRUCTURE AND TURBULENT MIXING USING PETASCALE COMPUTING

A Thesis
Presented to
The Academic Faculty

by

Kartik P. Iyer

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
August 2014

STUDIES OF TURBULENCE STRUCTURE AND TURBULENT MIXING USING PETASCALE COMPUTING

Approved by:

P. K. Yeung, Committee Chair
School of Aerospace Engineering
Georgia Institute of Technology

Suresh Menon
School of Aerospace Engineering
Georgia Institute of Technology

Richard Vuduc
School of Computational Science and Engineering
Georgia Institute of Technology

Mitchell Walker II
School of Aerospace Engineering
Georgia Institute of Technology

Katepalli R. Sreenivasan
Courant Institute of Mathematical Sciences
New York University

Date Approved: May 16, 2014

In memory of my beloved tata

Shri K.V. Iyer

ACKNOWLEDGEMENTS

This work marks a major milestone in my life, so some reflection on the people who have made it possible, only seems apropos.

I would like to thank Prof. P.K. Yeung, my adviser for his guidance and care. He has taught me the virtue of critically examining ones own work, before deriving conclusions. I have hugely benefited from attending the *SC* and *APS* conferences over the years, made possible due to his support. I can confidently state that I am more focused and confident about my abilities today, than I was, when I first met him. I thank him for all this and more. I would like to take this opportunity to thank Prof. Sreenivasan for guiding me and encouraging him through some difficult times in my doctoral work. I have been introduced to new problems and ideas by reading his papers and interacting with him on a regular basis. I can safely conclude that the quality of my work would have been greatly diminished if not for his care and support. For this I will be indebted to him forever. I take this opportunity to thank Prof. Vuduc who has always kept an open door to listen to my travails, in HPC and beyond. I continue to learn more about HPC and algorithmic development from him and hope that this will continue in the future. His gracious financial support in the last semester of my work helped me a long way in completing my thesis successfully. I deeply and sincerely thank him for his largess. I would also like to thank my thesis committee including Prof. Menon and Prof. Walker for critiquing my thesis. Over the years I have had the opportunity to learn from some of the leading experts in the fields of turbulence research and HPC. My work has been greatly influenced by Dr. Susan Kurien, Prof. Toshi Gotoh, Dr. Dmitiry Pekurovsky, Dr. Reginald Hill, Prof. Kai Schneider, Prof. Marie Farge and Dr. Rajeev Thakur. I thank them for the inspiration and guidance with various stages of my thesis. I would also like to thank Prof. Jeichel Jagoda and the school chair Prof. Vigor Yang for their support, encouragement and unending supply of free coffee. The contributions of my lab mates Diego Donzis, Jason Hackl, Dhawal Buaria and Shine Xiaomeng Zhai have also

been invaluable during the course of my thesis. Special thanks to Susheel Shekhar, my friend and former room mate for often providing an alternate perspective of things around us.

My doctoral work has proven to be very demanding. It has required lots of support and care in my personal life. I thank my coaches and friends at CFED, especially Lauren Sandoval, Erinn Garner, Damon Mosley and Jenn Purdy for helping me stay positive and fit during the last one year of my thesis work. I would like to thank my extended family back home and here in the US, for all the help, love and patience. I thank my aunt Gowri and uncle Govind for being the bedrock of my life over all these years. Their hard work and love continues to inspire my everyday. I would like to thank Gaachi for all the loving emails, help and encouragement she has given me over throughout this journey. Special thanks and fist bumps to my cousins Vishal, Sirish and Nutan who have helped me immensely through these years. I thank my crew Preetish, Pupp, Jawahar and Richa for putting up with me through all this. I deeply appreciate the love and support of Shubhangi and Gopal. They have not only provided me a place I can call home here in Atlanta, but also sprung into action whenever I have needed help. Their unwavering support has proven to be incredibly comforting. My love and gratitude for my brother-in-law Ganesh and sister Veena grows each day. I thank them for all the encouragement, laughs and love they continue to give me. I would like to thank the two most important people of my life, who have made all this possible, my amma and appa. They have been with me through this journey from day one, supported me, loved me and encouraged me, unconditionally. I could not have done this without them. Finally, I dedicate this work to the memory of my late grandfather who instilled the spirit of scientific curiosity in me during my childhood days. I also feel that I could not have singularly pursued this end without inheriting some of his iron will.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

A large direct numerical simulation database spanning a wide range of Reynolds and Schmidt number is used to examine fundamental laws governing passive scalar mixing and turbulence structure. Efficient parallel algorithms have been developed to calculate quantities useful in examining the Kolmogorov small-scale phenomenology. These new algorithms are used to analyze data sets with Taylor scale Reyolds numbers (R_λ) as high as 650 with grid-spacing as small as the Kolmogrov length scale.

Direct numerical simulation codes using pseudo-spectral methods typically use transpose based three-dimensional (3D) Fast Fourier Transforms (FFT). The ALLTOALL type routines to perform global transposes have a quadratic dependence on message size and typically show limited scaling at very large problem sizes. A hybrid MPI/OpenMP 3D FFT kernel has been developed that divides the work among the threads and schedules them in a pipelined fashion. All threads perform the communication, although not concurrently, with the aim of minimizing thread-idling time and increasing the overlap between communication and computation. The new algorithm is seen to give a 30% improvement over pure-MPI based algorithms at higher core counts.

Turbulent mixing is important in a wide range of fields ranging from combustion to cosmology. Schmidt numbers range from $O(1)$ to $O(0.01)$ in these applications. The Schmidt number dependence of the second-order scalar structure function and the applicability of the so-called Yaglom's relation is examined in isotropic turbulence with a uniform mean scalar gradient. At the moderate Reynolds numbers currently achievable, the dynamics of strongly diffusive scalars is inherently different from moderately diffusive Schmidt numbers. Results at Schmidt number as low as 1/2048 show that the range of scales in the scalar field become quite narrow with the statistics of the small-scales approaching a Gaussian state. A much weaker alignment between velocity gradients and principal strain rates and a strong

departure from Yaglom's relation have also been observed. Evaluation of different terms in the scalar structure function budget equation assuming statistical stationarity in time shows that with decreasing Schmidt number, the production and diffusion terms dominate at the intermediate scales possibly leading to non-universal behavior for the low-to-moderate Peclet number regime considered in this study.

One of the few exact, non-trivial results in hydrodynamic theory is the so-called Kolmogorov 4/5th law. Agreement for the third-order longitudinal structure function with the 4/5 plateau is used to measure the extent of the inertial range, both in experiments and simulations. Direct numerical simulation techniques to obtain the third order structure structure functions typically use component averaging, combined with time averaging over multiple eddy-turnover times. However, anisotropic large scale effects tend to limit the inertial range with significant variance in the components of the structure functions in the intermediate scale ranges along the Cartesian directions. The net result is that the asymptotic 4/5 plateau is not well attained. Motivated by recent theoretical developments we present an efficient parallel algorithm to compute spherical averages in a periodic domain. The spherically averaged third-order structure function is shown to attain the $K41$ plateau in time-local fashion, which reduces the need for running direct numerical simulations for multiple eddy-turnover times.

It is well known that the intermittent character of the energy dissipation rate leads to discrepancies between experiments and theory in calculating higher order moments of velocity increments. As a correction, the use of three-dimensional local averages has been proposed in the literature. Kolmogorov used the local 3D averaged dissipation rate to propose a refined similarity theory. An algorithm to calculate 3D local averages has been developed which is shown to scale well up to 32768 cores. The algorithm, computes local averages over overlapping regions in space for a range of separation distances, resulting in N^3 samples of the locally averaged dissipation for each averaging length. In light of this new calculation, the refined similarity theory of Kolmogorov is examined using the 3D local averages at high Reynolds number and/or high resolution.

CHAPTER I

INTRODUCTION

Turbulence is a commonly occurring state of fluid motion both in nature and engineering. Pollutant dispersion in the atmosphere, weather patterns such as hurricanes, mixing inside a combustor and astrophysical phenomena including cosmology and supernovae, all involve turbulence. Despite its ubiquity, the "problem of turbulence" has been called the last unsolved problem of classical physics. Turbulent flows are unsteady, irregular and are characterized by seemingly random fluctuations that arise over a wide range of scales in time and three-dimensional space. The resulting set of equations are highly non-linear in nature and difficult to solve. Experiments and simulations have contributed to the physical understanding of turbulence and advancements in turbulence theory (Warhaft 2009). In particular, rapid advances in computing power and technology have facilitated substantial progress in the study of turbulence through large-scale computations (Ishihara *et al.* 2009). In the following, a brief review of the turbulence problem and its treatment is given.

A characteristic feature of all turbulent flows is a high Reynolds number ($R_l \equiv Ul/\nu$, where U and l are characteristic velocity and length scales of the flow and ν is the kinematic viscosity of the fluid), which is a measure of the range of scales in the flow. Energy is transferred downscale locally via an energy cascade (Richardson 1922), in which energy enters the turbulence at the largest scales of motion through the production mechanism and is transferred to smaller and smaller scales until, at the smallest scales, the energy is dissipated by viscosity. The resulting non-linear system of equations are acutely sensitive to perturbations in initial conditions, boundary conditions and material properties. Experiments in turbulence are thus non-repeatable and as such the flow variables in such flows, such as the velocity, pressure etc. can be considered as "random" (even though equations such as the Navier-Stokes equations are deterministic). Consequently a statistical description of the turbulent flow field is more natural. A convenient approach is to decompose

the instantaneous flow field into its mean and fluctuating parts, such that the mean of the fluctuation is zero. This method is known as the Reynolds decomposition and forms the basis for many turbulence theories and models (for a review see Jackson & Launder 2007). However in the resulting equations, the so-called "closure" problem in turbulence arises. That is an equation for a moment of any order generated from the Navier-Stokes equation always contains higher-order moments resulting in more unknowns than the number of available equations. In turbulence, higher-order moments cannot be explicitly expressed as a function of lower-order moments as in a Gaussian process. Thus, some additional assumptions are required to close the set of equations at any order. Various statistical models (like Reynolds-stress models, subgrid scale models, spectral models and probability density function models) have been used to overcome the closure problem with varying degrees of success (Speziale 1991; Pope 1994). Alternatively, the Navier-Stokes equations can be solved numerically to determine the instantaneous velocity field directly. This methodology, known as Direct numerical simulations (DNS) avoids the closure problem arising out of Reynolds decomposition and is unrivalled in accuracy and the level of description it can provide. However, it is computationally very intensive with the computing cost increasing roughly as the cube of the Reynolds number R_l. Because DNS of high Reynolds number turbulence is expensive, it is often used to study simplified types of turbulent flows. One such simplification is achieved using the concept of homogeneous, isotropic turbulence, that is, turbulence that is statistically invariant under translation, rotation and reflection of coordinate axes. The introduction of homogeneous, isotropic turbulence marked the beginning of the use of of formal statistical methods involving correlations, Fourier transforms and power spectra in the turbulence literature (Taylor 1935, 1938) and has served as a test-bed for many analytical theories in turbulence.

Perhaps the best-known classical turbulence theory is that of Kolmogorov (1941a) [hereafter $K41$]. The first hypothesis of $K41$ states that, at sufficiently large Reynolds numbers, the small scales of turbulence are statistically isotropic, regardless of how the turbulence is produced. Stated differently, at sufficiently high Reynolds numbers, the small scales are statistically de-coupled from the large scales (which are closely coupled to the mean flow

and must differ appreciably from one turbulent flow to another) and hence universal. As per the second $K41$ hypothesis, towards the upper end of the small-scale range (in the so-called inertial range), the energy spectral density varies with the wavenumber k according to $E(k) = C\langle\epsilon\rangle^{2/3}k^{-5/3}$. Here $\langle\epsilon\rangle$ is the mean energy dissipation rate and C is a universal constant. This theory is consistent with Richardson's notion of an energy cascade in that, it supposes energy injected at the large scales (from the mean flow) is transferred to the small scales (where most of the dissipation occurs) through a series of (nearly) inviscid steps, involving the interaction of neighbouring scales only. Experiments by many authors (Grant *et al.* 1962; Saddoughi & Veeravalli 1994) have provided considerable support for the Kolmogorov hypotheses, although deeper explorations have revealed that strong departures from $K41$ universality exist. This is in part due to the oversimplified notion of the energy cascade which assumes that the energy transfer results from the interaction between local scales only, instead of all possible triads of wavenumber interactions allowed by the Navier-Stokes equations. Furthermore, the energy dissipation rate is characterized by intense fluctuations in space and time, a phenomenon known as intermittency. The unconditional mean of the dissipation rate $\langle\epsilon\rangle$ in $K41$ does not account for the extreme fluctuations of dissipation in space/time and in essence does not account for intermittency.

To address the shortcomings of $K41$, Oboukhov (1962) suggested examining the dissipation rate averaged over local regions of varying scale sizes. Using the idea of locally averaged dissipation rate, Kolmogorov (1962) [hereafter K62] introduced the Refined Similarity Hypothesis. One of its manifestations is that the various scaling exponents characterizing small-scale statistics are anomalous, that is, exponents for each order of the moment have to be determined individually and does not follow classical $K41$ arguments. The anomaly of the scaling exponents is due to small scale intermittency, which means that intense events are far more probable (in time and space) than that can be expected from a Gaussian field and statistics of increasingly smaller scales are increasingly non-Gaussian. An understanding of small-scale phenomena (such as intermittency) is essential, not only for turbulence theory, but also for turbulence modelling purposes. A major effort in turbulence theory aided by experiments and simulations has been the study of intermittency and anomaly of

scaling exponents (Sreenivasan & Antonia 1997). However, testing of K62 has been hampered by difficulties in laboratory measurement as well as in numerical simulations. In the latter, calculations involving 3D averages require heavy computations and inter-processor communications, making it prohibitively expensive to reach larger problem sizes (or higher Reynolds numbers). Consequently, both experiments (Stolovitzky *et al.* 1992, 1995) and simulations (Wang *et al.* 1996, 1999) so far have used local averages of dissipation rate averaged over a one-dimensional (1D) region rather than a three-dimensional (3D) volume in order to test K62. One of the objectives of the work reported in this thesis is to use a large numerical simulation database to examine the behavior of local 3D averages of the dissipation rate and related quantities.

An important consequence of turbulence is efficient mixing of scalar contaminants in the fluid. The disorderly velocity fluctuations break up local inhomogeneities of a transported substance like chemicals or other material properties such as heat into smaller and smaller scales, where molecular diffusion becomes important. Turbulent mixing is important in a broad range of situations, such as combustion, pollutant dispersion, underwater oil spills and accidental radioactive releases. The basic physics of turbulent mixing can be represented by the transport of a passive scalar (Sreenivasan 1991; Shraiman & Siggia 2000; Warhaft 2000), such as a small temperature fluctuation or a low solute concentration, which does not alter the flow dynamics. Weakly heated flows and mixing of non-reacting tracers such as pollutants and low-concentration dyes in a liquid offer typical examples of passive scalar mixing. Turbulent mixing may also be coupled to the flow dynamics such as mixing of different density fluids in an gravitational field, and in many cases the mixing produces changes to the fluid itself as in most combustion phenomena, detonations and thermonuclear supernova explosions. While such scenarios are clearly more complex (Dimotakis 2005), the mixing of passive scalars can be considered as the rate-limiting process in phenomena such as non-premixed combustion (Bilger 2004).

The efficiency of turbulent mixing is dependent upon the process by which large scale inhomogeneities in the scalar field are broken down into smaller and smaller scales by the turbulence, which in turn depends on how the velocity and scalar fluctuations interact at

different scales. The range of scales in the velocity field, expressed as the ratio between the integral length scale to the Kolmogorov length scale, can be expressed as a power of the Reynolds number (Pope 2000), while for scalars an additional parameter required is the ratio of the fluid kinematic viscosity to the molecular diffusivity, called the Schmidt number (Sc). The value of Sc varies widely, from $O(0.01)$ in liquid metals, to $O(1)$ in gas-phase combustion, to as high as $O(1000)$ for organic dyes. Since turbulent mixing in many fields (such as combustion) typically involve moderately diffusive scalars ($Sc \lesssim 1$), this regime has received the most attention in the literature (Corrsin 1951; Warhaft 2000). In particular, the smallest scale is the Obukhov-Corrsin scale, $\eta_{OC} = \eta Sc^{-3/4}$, and in the so-called inertial-convective range (Obukhov 1949; Corrsin 1951) at high Reynolds number the spectrum is expected to scale as $k^{-5/3}$. Indeed by contrast, the regimes of high Schmidt numbers and low Schmidt numbers are much less understood. For $Sc \gg 1$ the smallest scale is the Batchelor scale, $\eta_B = \eta Sc^{-1/2}$ which is smaller than the Kolmogorov length scale, and leads to challenges in resolving the small scales. Recent numerical work (Donzis *et al.* 2010) has provided strong evidence of a k^{-1} spectrum in the so-called viscous-convective scaling range, at intermediate scale sizes ($1/k$) such that diffusive effects are not important. In contrast, the strongly diffusive case (of $Sc \ll 1$) has received the least attention partly due to the difficulty in handling liquid metals in experiments. The range of scales becomes narrower ($\eta_{OC} \gg \eta$ for $Sc \ll 1$) and spectral transfer can be significantly reduced with increasing molecular diffusivity. Indeed, recent work with Sc as low as 1/2048 (Yeung & Sreenivasan 2013, 2014) suggests that mixing at very low Sc is fundamentally different in many aspects. The study of issues related to local isotropy in the case of low-Schmidt number mixing is another objective of this thesis.

The behavior of the small scales in the scalar field is not only important from a theoretical standpoint, but also from a practical perspective, as they influence reaction rates, dispersion and mixing of scalars. Traditionally, Kolmogorov (1941a) has been the basis for phenomenology of passive scalars as well, with the Kolmogorov-Obukhov-Corrsin (KOC) argument extending the theory of local isotropy to fluctuations of passive scalars (Obukhov

1949; Corrsin 1951). However, most data sources suggest that there are substantial departures from local isotropy (Sreenivasan 1991; Warhaft 2000; Yeung *et al.* 2002). For example a nontrivial skewness of scalar gradient fluctuations in the direction of an imposed uniform (non-zero) mean gradient in the scalar field (Sreenivasan & Tavoularis 1980) suggests the effects of the large scales may be felt directly at the small scales. Another aspect unaccounted for in the KOC theory is the phenomenon of intermittency at the small scales in the scalar field as evidenced by the large excursions from the mean of the scalar dissipation rate at sufficiently high Reynolds number (Sreenivasan & Antonia 1997). Analogous to the velocity field, scalar field intermittency leads to the anomalous scaling problem in the scalar field, whereby higher order statistics of the small scales do not obey usual KOC arguments. It is well known that scalar fields at $Sc \sim O(1)$ are more intermittent that the velocity field (Prasad *et al.* 1988; Donzis & Yeung 2010). Scalar field intermittency is known to occur even in the absence of velocity field intermittency (Kraichnan 1994; Holzer & Siggia 1994) and is closely related to the departure from local isotropy at the small scales (Warhaft 2000) indicating that the mechanisms for intermittency may be different in the scalar and velocity fields. In spite of these differences, in order to account for the strong fluctuations in the local rates of dissipation in the scalar field, K62 theory has been extended to passive scalars (Stolovitzky *et al.* 1995). As for the velocity field, the testing of the refined similarity theory for passive scalars has been restricted by challenges in the complexity of the measurements involved. The local 1D average of dissipation rate has been used instead of 3D average, both in experiments and simulations (Sreenivasan & Kailasnath 1993; Wang *et al.* 1996, 1999). Objectives of the work reported in this thesis include the examination of intermittency characteristics of the scalar dissipation rate from the viewpoint of local 3D averages.

Another measure of the spatial structure of turbulent flows is the statistics of products of two-point differences of scalar and velocity fluctuations. The averages of such products are called structure functions and are useful in theories of similarity scaling (Frisch 1995; Sreenivasan & Antonia 1997). In the inertial range of a sufficiently high Reynolds number turbulent flow, wherein scales are far smaller than the anisotropic large scales but far larger

than the dissipative small scales, Kolmogorov (1941b) theory predicts that the m-th order longitudinal velocity structure function, scales as $(\langle\epsilon\rangle r)^{m/3}$, where $\langle\epsilon\rangle$ is the mean energy dissipation rate. For the case of $m = 3$ an exact equation (exact in the asymptotic sense) can be derived from the Karman-Howarth equation as $\langle(\Delta_r u)^3\rangle = -(4/5)\langle\epsilon\rangle r$ (Kolmogorov 1941b), which is free from any intermittency corrections and is called the Kolmogorov's 4/5-th law. Analogously, there exists an exact relation for the scalar field called Yaglom's relation in the inertial-convective range, $\langle(\Delta_r u)(\Delta_r \phi)^2\rangle = -(2/3)\langle\chi\rangle r$ (Yaglom 1949) where $\langle\chi\rangle$ is the mean scalar dissipation rate. Previously, DNS data in both moderately diffusive ($Sc \lesssim 1$) and weakly diffusive ($Sc \gg 1$) regimes have been found to provide good support for Yaglom's relation. For $Sc \sim O(1)$ and higher, $\langle\chi\rangle$ also follows the concept of dissipative anamoly (Donzis *et al.* 2005), which means that $\langle\chi\rangle$ is determined by the large scales through a spectral cascade, independent of the molecular diffusivity. A strong spectral cascade, which is increasingly clear with increasing Reynolds number and/or Schmidt number, is also a requirement for Yaglom's relation to be valid. However in the strongly diffusive case, if the Schmidt number is very low, the range of scales becomes narrow leading to a weaking of the spectral cascade (Yeung & Sreenivasan 2014). Thus, there are strong reasons to expect that Yaglom's relation may not hold in the decreasing Sc regime. To examine more carefully the conditions required for Yaglom's relation, we verify analytically and study numerically the budget of the scalar structure function equation (Gotoh & Yeung 2013). Apart from the moments of structure functions, we also focus on the probability distribution of scalar increments as a function of scale size, as well as the joint distribution of velocity and scalar increments. Thus, the study of structure functions (and mixed structure functions) which provide important information about isotropy and intermittency at various scale sizes, is another objective of this work.

In this work, we examine issues in intermittency and turbulent mixing using Direct Numerical Simulations (DNS), where all relevant scales of motion are resolved according to exact conservation equations (for a review see Moin & Mahesh 1998). The value of DNS not only lies in the completeness of data they provide, but also in the opportunity they provide for examining turbulence theories under controlled conditions. For instance,

spectral energy transfer is difficult to measure experimentally, whereas it can be extracted from DNS (Domaradzki & Rogallo 1990) with relative ease. Furthermore, DNS is invaluable as a research tool for development of turbulence models (for example see Kadoch *et al.* 2011). However, DNS is inherently computationally intensive as it solves for all relevant scales of motion both spatially and temporally. The range of scales both in space and time increase with Reynolds number (roughly as $R_\lambda^{9/2}$ in 3D space and $R_\lambda^{3/2}$ in time, where R_λ is the Reynolds number based on the Taylor micro-scale), causing the number of floating-point operations required to roughly vary as ${R_\lambda}^6$, thus imposing limitations on the time step and grid spacing that can be used in the simulations. Thus, achieving realistic Reynolds numbers in DNS is a major challenge. Historically, the DNS approach was infeasible until the $1970s$ when computers of sufficient power became available, although the Reynolds numbers (Orszag & Patterson 1972) that could be attained were significantly lower than the Reynolds numbers that were achieved in wind-tunnel experiments. As the computing power began to increase as predicted by Moore (Moore 1965), DNS started becoming more feasible as a research tool. Besides increase in processor speeds, the advances in parallel computing has made DNS more viable, capable of (today) reaching Reynolds numbers comparable to those in laboratory measurements (Kaneda *et al.* 2003; Ishihara *et al.* 2007; Yeung *et al.* 2012). Typically, the solution domain is divided among multiple processors, such that each processor computes data in its own memory. The processors synchronize and exchange data as needed using a communication protocol such as the Message Passing Interface (Gropp *et al.* 1999). Obviously, the manner in which the processors perform computations and communicate data with each other has a bearing on the performance, thus determining the complexity of the physical problem that can be addressed (for example, rotating flows, flows with density stratification or reacting flows).

Our philosophy is to push the envelope of available computing power, which requires us to delve into some current state-of-the-art issues in high performance computing. We have been able to access several computers with theoretical peaks in excess of 1 Petaflops (10^{15} floating point operations per second). In particular, we have the privilege of accessing the Blue Waters supercomputer at the National Center for Supercomputing Applications

(NCSA). Blue Waters is a Cray machine with over $700,000$ XE6 cores and is expected to deliver a sustained performance of one Petaflops. To use such massive machines effectively, we map the $3D$ solution domain onto a two-dimensional ($2D$) computational grid. This 2D domain decomposition, in principle allows up to N^2 central processing unit (CPU) cores for a solution domain of N^3 grid points (Donzis *et al.* 2008*a*). The cores communicate with one another using MPI to pass messages across the network. The current production DNS code has been tested up to 262144 cores on Cray $XE6$ (Blue Waters) with reasonable scalability.

In the last decade, two important developments in the computing industry have shaped our efforts into experimenting with different programming paradigms, (1) symmetric multiprocessors (SMP) and (2) graphics processing unit (GPU). Almost all the supercomputers today use SMPs, wherein a group of identical processors are connected to a single shared main memory. This group of processors together with the main memory is commonly referred to as a node. The trend towards SMP architectures is primarily due to energy considerations which has limited single processor clock rates. Supercomputers generally use NUMA (non-uniform memory access) architectures, which contains a memory hierarchy with processors accessing local memory quickly and remote memory (which is shared among processors in a node) more slowly. Traditionally, our codes have used MPI to perform the inter-processor communication which send messages across the network even for processors residing inside the same node (which share a main memory), thus treating the whole system as a distributed memory machine. In order to use the shared memory characteristics of multicore systems, we have developed a hybrid programming algorithm, which uses MPI to communicate across nodes as before, but uses OpenMP threads (Chapman *et al.* 2007) to share the memory within a node. In theory this obviates the exchange of messages across the network between cores sharing the same main memory and thus helps decrease communication costs. The light-weight threads also help with balancing the work load among cores inside a node. The production DNS code is capable of using the shared memory characteristics of multicore processors through a hybrid programming model that uses OpenMP threads within each node and message-passing communication (per the standard MPI interface) across the nodes. We have also explored a modified strategy that

allows overlapping between computation and communication. In particular, one approach that will enable overlap at a fine-grained level is to divide work among OpenMP threads in a pipelined fashion, so that only a single thread performs the MPI communication at any given time. This has been implemented in a kernel for the most time-consuming computation in the DNS code, namely the 3D Discrete Fourier Transforms (DFT), but still needs further optimization before eventual incorporation into our production DNS code.

In the past decade GPU's have been used successfully in conjunction with CPU's to accelerate scientific computations (Kirk & Hwu 2010). This use of GPUs along with CPUs is commonly referred to as heterogeneous computing. By offloading compute-intensive portions of applications to the GPU, while running the remainder of the code on the CPU, the GPU offers possible improvements in application performance, provided the data transfer time from the CPU to the GPU (and back) can be efficiently managed. The launch of the compute unified device architecture (CUDA) by NVIDIA in 2006, which is a software and hardware architecture that enabled the GPU to be programmed with a variety of high level programming languages was an important step in harnessing the power of the GPU for general purpose computing (Sanders & Kandrot 2010). Today's GPUs greatly outpace CPUs in arithmetic throughput and memory bandwidth, making them well suited for data parallel applications. We have attempted to offload the most computationally intensive portions of the DNS algorithm (3D FFTs) to GPUs, while the CPUs performs the required inter-processor communication. Tests conducted on NCSA's (now retired) Lincoln cluster which is a heterogeneous machine with NVIDIA Tesla accelerator units have shown a significant decrease in the computation time, although the inter-processor communication (performed by the CPUs) time remains unchanged. Because the 3D FFT algorithm is communication dominated, the use of GPUs has not yet proven beneficial to the overall performance, although preliminary steps have been taken to enter the era of heterogeneous computing.

In summary, the study of small-scale turbulence and passive scalar mixing using DNS can offer deeper insights into the flow physics. To this end we use the latest tools available to us in the field of high performance computing. The development of new algorithms to efficiently compute 3D FFTs and 3D local averages at the Petascale level are interesting

problems in their own right. Furthermore, it offers impetus to rapid improvements in parallel processing technologies like communication protocols and parallel system hardware. The central theme of this work is to develop new algorithms using cutting-edge computing resources to investigate some fundamental issues in turbulence research. In what follows, we briefly outline the main objectives of this work.

1.1 Objectives

The general goals of this thesis are as follows.

1. To develop a hybrid DNS code that uses OpenMP threads inside processors while using MPI processes for communication between threads across processing units in a serialized manner. Overlap between communication and computation is to be achieved by using a pipe-lined strategy, ensuring that the communication time is masked behind other useful work.

2. To study the scalar field structure at very low Schmidt numbers with the aim of understanding the dynamics behind the mixing process. Statistical properties as a function of scale size are to be analyzed in order to test local isotropy assumptions. Furthermore, the balance of the scalar structure function equation will be studied with an emphasis towards low Schmidt number mixing.

3. To develop an efficient algorithm to compute local 3D averages, which can be used to study the scaling properties of intermittent quantities such as the energy dissipation rate. The intermittency of the scalar field is to be compared to that of the velocity field and the results are to be contrasted against the often reported, local $1D$ averages. The $K62$ hypotheses are to be tested using the existing database that spans a substantial range of Reynolds numbers.

More details about these problems along with results are given in the remaining sections. The rest of this thesis is organized as follows. Chapter 2 discusses the numerical and computational issues including the equations solved, numerical methods employed and the parallel algorithms developed. In chapter 3, a pipelined version of all-to-all data exchange

is explained, that allows overlap between computation and communication using OpenMP threads and MPI processes. In chapter 4 we discuss turbulent mixing of passive scalars with a focus on the very low Schmidt number regime. In chapter 5, a new parallel algorithm to compute spherical averages is presented. The results of spherical averaging are contrasted with that of the usual techinique of component averaging. In chapter 6 we discuss current issues in the scaling properties of intermittent quantities (such as the energy dissipation rate) in the context of local averages as postulated in K62 theory. A detailed summary of the results and a discussion of future work is the subject of chapter 7.

CHAPTER II

NUMERICAL ALGORITHM

A numerically exact solution to the Navier-Stokes equations is sought for all relevant length and time scales considered, in the form of velocity fields represented as discrete Fourier modes in wavenumber space. A pseudo-spectral method is employed in which bilinear products are evaluated in physical space instead of using convolution in wavenumber space. The aliasing errors thus incurred are controlled by a combination of truncation and phase shifting techniques (Rogallo 1981). In addition, a scalar transport equation can be solved for, in order to study how turbulence effects the mixing process. The main numerical and computational aspects of the algorithm used are the time-stepping strategy, control of aliasing errors and the implementation on distributed-memory parallel computers. After a summary of the equations involved, each of these aspects is briefly explained in the following.

2.1 Velocity field

Assuming a constant density flow with no mean velocity, the fluctuating velocity field in Cartesian tensor notation is governed by the following equations

$$\frac{\partial u_j}{\partial t} + \frac{\partial (u_j u_k)}{\partial x_k} = -\frac{1}{\rho}\frac{\partial p}{\partial x_j} + \nu \frac{\partial^2 u_j}{\partial x_k \partial x_k} \,, \tag{2.1}$$

$$\frac{\partial u_k}{\partial x_k} = 0 \,, \tag{2.2}$$

where, (u_1, u_2, u_3) denote the fluctuating velocity components in the Cartesian notation, p is the fluctuating pressure, ρ is the density (assumed constant) and repeated indices imply summation. The dependence of u_j on 3D space and time $(\mathbf{x}, t)$ has been suppressed in Eqs. 2.1 and 2.2 for brevity. The boundary conditions are periodic and are given as,

$$u_j(\mathbf{x} + nL_o\mathbf{e}_k, t) = u_j(\mathbf{x}, t) \qquad n = 1, 2, 3, \ldots \,, \tag{2.3}$$

where L_0 denotes the box length, $\mathbf{e}_k$ is a unit vector in the k^{th} direction and $\mathbf{x} = (x_1, x_2, x_3)$ denotes a position vector in the domain. The use of periodic boundary condition enables

the representation of the velocity field as a finite Fourier series,

$$u_j(\mathbf{x},t) = \sum_{\mathbf{k}} \hat{u}_j(\mathbf{k},t) e^{i\mathbf{k}\cdot\mathbf{x}} \ , \tag{2.4}$$

where $\mathbf{k}$ denotes a wavenumber vector, $\hat{u}_j$ is the Fourier coefficient (or a Fourier mode) of u_j and $i \equiv \sqrt{-1}$. Typically we use a box of length $L_0 = 2\pi$, although we use a larger domain (such as $L_0 = 4\pi$) for studying problems in which the large scales become larger. Typical examples where the large scales grow include rotating flows and low Schmidt number mixing. The choice of L_0 as a multiple of 2π is made for convenience in the Fourier representation of the velocity field.

In physical space, the solution domain is a cube of side L_0 with N^3 grid points which are located at $\mathbf{x}$, where $\mathbf{x}$ is defined as $(l_1\Delta x, l_2\Delta y, l_3\Delta z)$, l_1, l_2, l_3, are integers between 0 and $N-1$, inclusive and $\Delta x = \Delta y = \Delta z = L_0/N$. In wavenumber space there are N^3 discrete wavenumbers, $(m_1 k_0, m_2 k_0, m_3 k_0)$, where m_1, m_2, m_3, are integers between $1-N/2$ and $N/2$, both inclusive and k_0 is the lowest non-zero wavenumber magnitude, and is equal to $2\pi/L_0$. The flow field variables like $\mathbf{u}(\mathbf{x},t)$ are real valued and hence their Fourier coefficients obey conjugate symmetry, $\hat{\mathbf{u}}(-\mathbf{k},t) = \hat{\mathbf{u}}^*(\mathbf{k},t)$, where the asterisk denotes a complex conjugate, due to which only half the modes need to be stored in memory. Thus we use the same memory location to alternatively store the velocity field in Fourier space ($N^3/2$ complex modes) and physical space (N^3 real space grid values).

The numerical scheme is a Fourier pseudo-spectral algorithm based on Rogallo (1981) which advances the Fourier modes in time according to the Navier-Stokes equations (Eqs. 2.1 and 2.2) in wavenumber space. The equations for the time evolution of $\hat{u}_j(\mathbf{k},t)$ can be obtained by taking the Fourier transform of Eqs. 2.1 and 2.2 and eliminating the pressure term using the incompressiblity condition (Eq. 2.2) and are as follows,

$$\Big(\frac{d}{dt} + \nu k^2\Big)\hat{u}_j(\mathbf{k},t) = -\Big(\delta_{jl} - \frac{k_j k_l}{k^2}\Big)\hat{N}_l(\mathbf{k},t) \ , \tag{2.5}$$

$$k_k \hat{u}_k = 0 \ , \tag{2.6}$$

where δ_{jl} is the Kronecker delta function defined as $\delta_{jl} = 1$ if $j = l$ and $\delta_{jl} = 0$ if $j \neq l$. The right-hand-side term in Eq. 2.5, $\hat{N}_l(\mathbf{k},t)$ denotes the Fourier transform of the nonlinear

term in Eq. 2.1 and is given by the following convolution sum,

$$\hat{N}_l(\mathbf{k},t) = ik_m \sum_{\mathbf{k}'} \hat{u}_l(\mathbf{k}',t)\hat{u}_m(\mathbf{k}-\mathbf{k}',t) \ , \tag{2.7}$$

where, $i \equiv \sqrt{-1}$. The term on the right hand side of Eq. 2.5 denotes the component of $\mathbf{N}$ perpendicular to $\mathbf{k}$. It can be seen in wavenumber space (from Eq. 2.7) that the convection term is nonlinear and non-local, involving the interaction of wavenumber triads, $\mathbf{k}$, $\mathbf{k}'$ and $\mathbf{k}$", such that $\mathbf{k}' + \mathbf{k}" = \mathbf{k}$. Thus, the partial differential equations (Eqs. 2.1 and 2.2) are transformed into a set of coupled ordinary differential equations (Eqs. 2.5 and 2.6) which can be integrated in time to obtain the velocity field $u_j(\mathbf{x},t)$. The computation of the non-linear terms ($\hat{N}_l(\mathbf{k},t)$ in Eq. 2.5) and the time integration are the main aspects of the algorithm and are briefly discussed in the following.

A direct calculation of $\hat{N}_l(\mathbf{k},t)$ using Eq. 2.7 is prohibitively expensive since it requires on the order of N^6 operations. To avoid this large cost, the variables are transformed into physical space where the nonlinear terms are computed and are then transformed back to wavenumber space. Hence the algorithm is classified as pseudo-spectral. The non-linear term $\hat{N}_l(\mathbf{k},t)$ in Eq. 2.5 is computed as

$$\hat{N}_l(\mathbf{k},t) = \widehat{\frac{\partial u_j u_l}{\partial x_j}} \ , \tag{2.8}$$

where, $\widehat{(\ \ )}$ denotes a backward transform from physical space to Fourier space. Thus the pseudo-spectral method requires a forward transform from Fourier space to physical space where the non-linear terms are formed and a backward transform to get the non-linear terms back into Fourier space. The transforms are computed using the Fast Fourier transform algorithm (Cooley & Tukey 1965) and require on the order of $(15/2)N^3 \log_2 N$ operations. The factor of 1/2 in the flop count arises due to conjugate symmetry, because of which only half the modes need to be transformed. As a result of computing the nonlinear terms by forming products in physical space and transforming back to Fourier space, aliasing errors are introduced which are controlled by a combination of phase shifts and spherical truncation (Rogallo 1981). The spherical truncation involves neglecting all Fourier modes outside the sphere,

$$k_1^2 + k_2^2 + k_3^2 \leq k_{max}^2 \ , \tag{2.9}$$

where, $k_{max} = \sqrt{2}Nk_0/3$ is the magnitude of the highest resolvable wavenumber (Canuto *et al.* 1987). Since only the truncated modes with magnitudes inside the sphere of radius k_{max} are considered, savings in both communication and computation costs in the parallel algorithm are possible (as discussed later).

The time integration in the set of coupled ordinary differential equations (Eqs. 2.5 and 2.6) is done using an explicit second-order Runge-Kutta method (Canuto *et al.* 1987). At time step t_n, Eq. 2.5 can be written as

$$\frac{d(F\hat{u}_j(\mathbf{k}, t_n))}{dt} = F\hat{c}_j(\mathbf{k}, t_n) \ , \tag{2.10}$$

where $\hat{c}_j(\mathbf{k}, t_n)$ denotes the right hand side of Eq. 2.5 at time step t_n and $F(t) = \exp\left(\int_{t_n}^{t} \nu k^2 dt\right)$, represents an integrating factor with $F(t_n) = 1$. We first compute a first-order estimate of $\hat{u}_j(\mathbf{k}, t_{n+1})$ in the predictor step as

$$F(t_{n+1})\hat{u}_j^p = \hat{u}_j(\mathbf{k}, t_n) + \hat{c}_j(\mathbf{k}, t_n)\Delta t \tag{2.11}$$

where $\Delta t = t_{n+1} - t_n$ is the integration time step. A second-order estimate of $u_j(t_{n+1})$ can then be obtained in the corrector step as

$$F(t_{n+1})\hat{u}_j(\mathbf{k}, t_{n+1}) = \hat{u}_j(\mathbf{k}, t_n) + \frac{\Delta t}{2}[\hat{c}_j(\mathbf{k}, t) + F(t_{n+1})\hat{c}_j^p] \ , \tag{2.12}$$

where $\hat{c}_j^p$ is the convective term computed using $\hat{u}_j^p$ (or the predictor field). For the time advancement to be accurate, it is necessary that a fluid particle move only a fraction of the grid spacing Δx in time Δt. In practice, the time step Δt is governed by Courant number restrictions. For a 3D problem, the Courant number is defined as

$$C \equiv \frac{\Delta t}{\Delta x}(|u| + |v| + |w|)_{max} \ , \tag{2.13}$$

where u, v and w are the Cartesian components of velocity at a grid point and the maxima is taken over the entire domain. For finite-difference schemes, numerical stability considerations usually require $C \leq 1$. For spectral methods, stability analysis is not so well established (Peyret & Taylor 1983), but it is common to follow the same criterion. As demonstrated by Eswaran & Pope (1988), using a Courant number greater than unity produces large errors in the calculation. A small value of C can improve accuracy in time but it also increases

the simulation time and thereby the computational cost. Typically, in our simulations we use $C = 0.6$, although smaller values of C are required for simulations involving scalars of low Sc , where the time scales are limited by the strong molecular diffusion.

The non-dimensional parameter $k_{max}\eta$ is a convenient parameter for resolution of the simulation ($(k_{max}\eta)(\Delta x/\eta) = 2\sqrt{2}\pi/3$). Typically a value of $k_{max}\eta$ between 1.0 and 2.0 is used in simulations (Yeung & Pope 1989; Kaneda *et al.* 2003; Ishihara *et al.* 2009),with $k_{max}\eta \approx 1.4$ (or $\Delta x/\eta \approx 2.1$) being a popular choice. However, a higher resolution is required for studying small scale statistics reliably (Ishihara *et al.* 2007). The work by Donzis *et al.* (2008*b*) shows that a $k_{max}\eta$ of at least 3 is needed to determine accurately the moments of dissipation and enstrophy up to order four and that the constraint becomes stronger with increasing order of the moment. Apart from resolving the small scales accurately, it is important to have a sufficiently large domain to minimize the effects of the periodic boundary conditions (Eq. 2.3) on the integral scales. Typically the ratio of the box size to the integral scale is between 5-6, although larger box sizes are used in simulations involving low Schmidt number mixing (as discussed later) as the scalar integral length scales are known to grow with decreasing Schmidt number (Donzis *et al.* 2005).

In our simulations, we extend the time integration over several large eddy turnover times $T_E \equiv L/u'$ where u' is the root mean square (rms) velocity fluctuation, since long-time averages are consistent with ensemble averages of the original K41 theory due to ergodicity (Frisch 1995; Galanti & Tsinober 2004). However Eqs. 2.1 and 2.5 correspond to the case of decaying turbulence due to zero mean velocity, such that time averages may not be very meaningful. In order to generate a statistically stationary velocity field we "force" the low wavenumber (i.e. large scale) velocity components in the simulation by artificially adding energy using the forcing scheme of Eswaran & Pope (1988). The velocity field is stochastically forced by adding acceleration increments to the largest scales only, such that continuity is satisfied and on an average dissipation equals the artificial production. Only the wavenumbers inside a sphere of radius k_F are forced (excluding the zero wavenumber mode). In most of our simulations $k_F = 2.1$. The inherent assumption in low wavenumber forcing is that the small scales only depend on the energy transfer from the large scales, but

not on the details of the production mechanism. This is increasingly valid as the range of scales increases, as suggested by Kolmogorov (1941a) hypotheses and is well supported by data in the literature (Sreenivasan 1998). The resulting velocity fields are stationary and isotropic, to a good approximation (Eswaran & Pope 1988).

An estimate for the computational cost of a DNS of 3D homogeneous, isotropic turbulence can be obtained from classical scaling estimates (Kolmogorov 1941a). The number of grid points required to resolve all scales varies as (Pope 2000) $N^3 \sim (L/\eta)^3 \sim R_\lambda^{9/2}$, where R_λ is the Taylor-scale Reynolds number. If R_λ is increased by (say) decreasing ν, then the grid spacing Δx should also be correspondingly decreased in order to resolve the small scales fully, otherwise the quality of the simulation will degrade. A decrease in grid spacing leads to a decrease in the time step as $\Delta t \sim \Delta x/u'$ (from Eq. 2.13). As mentioned previously, the duration of the simulation (T) needs to be large enough to span several large eddy turnover times. This gives the estimate for the number of time-steps ($M = T/\Delta t$) as $M \sim T_E u'/\Delta x \sim L/\Delta x$. Because the grid spacing should be of the order of the Kolmogorov length scale $\Delta x \sim \eta$, we finally obtain the estimate for the number of time steps as $M \sim L/\eta \sim R_\lambda^{3/2}$ (Pope 2000). Hence the CPU cost for DNS of turbulent flows roughly scales as

$$N^3 M \sim R_\lambda^6 \, . \tag{2.14}$$

The work by Yakhot & Sreenivasan (2005) points to an even stricter requirement ($N^3 M \sim R_\lambda^8$) for studies of intermittency. Since the computational resources needed increases rapidly with Reynolds number, DNS of high Reynolds number turbulence is a major challenge in parallel computing.

2.2 Scalar transport

In addition to the velocity field we consider a passive scalar denoted by $\phi(\mathbf{x}, t)$. Assuming Fickian diffusion with constant molecular diffusivity D_ϕ, the basic transport equation is

$$\frac{\partial \phi}{\partial t} + u_i \frac{\partial \phi}{\partial x_i} = -u_i \frac{\partial \langle \Phi \rangle}{\partial x_i} + D_\phi \frac{\partial^2 \phi}{\partial x_i \partial x_i} \, , \tag{2.15}$$

which governs the evolution of the fluctuation $\phi(\mathbf{x}, t)$ of a passive scalar Φ (with mean $\langle \Phi \rangle$) in an incompressible fluid velocity field $\mathbf{u}(\mathbf{x}, t)$ that satisfies the Navier-Stokes equations

(Eqs. 2.1 and 2.2). The mean scalar gradient term in Eq. 2.15 represents a source term as in Overholt & Pope (1996), hence no external forcing of the scalar field is necessary. For simplicity, we let the mean scalar gradient be spatially uniform, which is compatible with statistical homogeneity of scalar fluctuations in space. If the velocity field is statistically isotropic, the orientation of $\nabla\langle\Phi\rangle$ is arbitrary and it is sufficient to take it to be aligned with (any) of the three Cartesian coordinate axes. Hence we arbitrarily choose $\nabla\langle\Phi\rangle = (G, 0, 0)$ where G is a constant. The boundary and initial conditions for the mixing problem are as follows,

$$\phi(\mathbf{x} + nL_o\mathbf{e}_k, t) = \phi(\mathbf{x}, t) \qquad n = 1, 2, 3, \ldots \,, \tag{2.16}$$

$$\phi(\mathbf{x}, 0) = 0 \,. \tag{2.17}$$

The periodic boundary conditions make it convenient to represent the scalar field as a finite Fourier series. Since Eq. 2.15 is linear in the mean gradient and the calculations are started with zero initial conditions (Eq. 2.17), the resulting scalar fluctuations are proportional to the magnitude of the mean gradient (G) everywhere and at all times. Consequently, the scalar variance $\langle\phi^2\rangle$, the scalar flux $\langle u\phi\rangle$ and the mean scalar dissipation $\langle\chi\rangle$ are proportional to G^2, G and G^2 respectively, while (in contrast) statistics of the normalized quantities such as $\chi/\langle\chi\rangle$ which are considered in this study are independent of the magnitude of G.

The Fourier space representation of Eq. 2.15 can be written as

$$\Big(\frac{\partial}{\partial t} + D_\phi k^2\Big)\hat{\phi} = -\widehat{u_i \frac{\partial \phi}{\partial x_i}} - G\hat{u}_1 \,. \tag{2.18}$$

In this case we form the products such $u_i\partial\phi/\partial x_i$ in physical space and then transform to Fourier space in order to avoid the expensive convolution sum. The aliasing errors are treated as with the velocity field by a combination of phase shifts and spherical truncation. The number of scalars in the simulation (N_c) typically ranges from 1 to 4, representing scalars with different molecular diffusivities. In our experience each scalar requires roughly 30-40% more CPU time than the velocity field itself.

The simulation parameters such as the grid spacing (Δx), the time step (Δt) and the box length (L_0), may depend on the choice of the Schmidt number of the passive scalar.

For weakly diffusive scalars ($Sc \gg 1$) the smallest scale in the scalar field η_B is smaller than the Kolmogorov length scale η. Hence the relevant resolution parameter is $k_{max}\eta_B$ or equivalently $\Delta x/\eta_B$. A smaller grid spacing also implies a smaller time step Δt (Eq. 2.13) which in turn can increase the CPU hours needed to run the simulation for multiple large eddy turnover times. Hence the spatial and temporal resolution are the main limiting factors on how high a Schmidt number can be simulated reliably using a given amount of computing resources.

In the the case of strongly diffusive scalars ($Sc \ll 1$), the smallest scale η_{OC} is larger than the Kolmogorov length scale, hence $k_{max}\eta$ is still the relevant resolution parameter for the simulation. However, since the scalar integral scales (L_ϕ) grow with decreasing Schmidt number (Donzis *et al.* 2005; Yeung & Sreenivasan 2014), a larger solution domain is required to minimize the possible constraining effects of periodic boundary conditions. Accordingly a solution domain of larger length ($L_0 = 4\pi$) is used (verses the usual $L_0 = 2\pi$), while keeping the grid spacing unchanged. This requires eight times as many grid points for (nominally) the same Reynolds number, but also results in improved sampling for the large-scale statistics of the velocity field, whose motions are maintained by stochastic forcing.

A more demanding aspect of low Schmidt number simulations is that the fast molecular diffusion results in very small time scales ($\Delta t \sim \Delta x^2/D_\phi$) which requires the time step Δt to be much smaller than that required by numerical stability constraints (Yeung & Sreenivasan 2013). For the lowest Schmidt numbers that have been studied in this work, this constraint requires Δt to be an order of magnitude smaller than that usually chosen based on the Courant number associated with convective transport, which in turn increases the CPU time taken to reach a statistically stationary state. In addition, the use of very small Δt's makes the use of double precision arithmetic important in preventing round-off errors accumulating over tens of thousands of time steps. The time step Δt in such computations corresponds to Courant numbers lower than 0.075 and is only about 1% of a Kolmogorov time scale. At such small Δt's it is reasonable to expect the velocity field to change very little, or to follow a linear variation closely. Consequently, to reduce the overall CPU expense, in some of the larger runs involving low Schmidt number scalars

(performed by the lead author of the group), the velocity field is linearly interpolated over four consecutive time steps and is only then updated according to the exact equations. This strategy reduces the CPU time by about 25% from not having to take Fourier transforms of the velocity field at every time step.

2.3 Parallel algorithm

We use a parallel implementation of the pseudo-spectral algorithm of Rogallo (1981) in which the most time-consuming task is the computation of the 3D FFT. A transpose-based method, in which the FFT is computed locally, along a single dimension at a time, with global data exchanges done in between, is used (as described in Pekurovsky 2012). Given the stringent computational requirements (see Eq. 2.14) it is evident that the DNS of high Reynolds number turbulence requires the most powerful state-of-the-art computational resources that are available (*www.top500.org*). We have access to computing resources on machines at five supercomputing centers, namely Blue Waters at the National Center for Supercomputing Applications (NCSA), Edison at National Energy Research Scientific Computing Center (NERSC), Titan at National Center for Computational Sciences (NCCS), Kraken at National Institute of Computational Sciences (NICS) and Stampede at Texas Advanced Computing Center (TACC). Each of these machines consists of tens of thousands of processors, with a theoretical flop count in the Petaflop range. In this section, we give a brief description of the domain decomposition technique used, which attempts to utilize the massive computational resources at our disposal, followed by a discussion of the computation and communication characteristics of the parallel algorithm.

A two-dimensional (2D) domain decomposition technique (Donzis *et al.* 2008*a*) is used to map the 3D physical domain onto a 2D computational grid. The use of 2D computational grid allows the use of a maximum of N^2 processors in theory, for an N^3 domain with N points along each direction, thus enabling the use higher core counts. Decomposition of the grid among more processors decreases the memory required per processor, which is especially beneficial for larger simulations such as 4096^3 and beyond. A schematic of the mapping of the N^3 domain onto a 2D processor grid with $P = P_{row} \times P_{col}$ MPI processes (hereafter, a

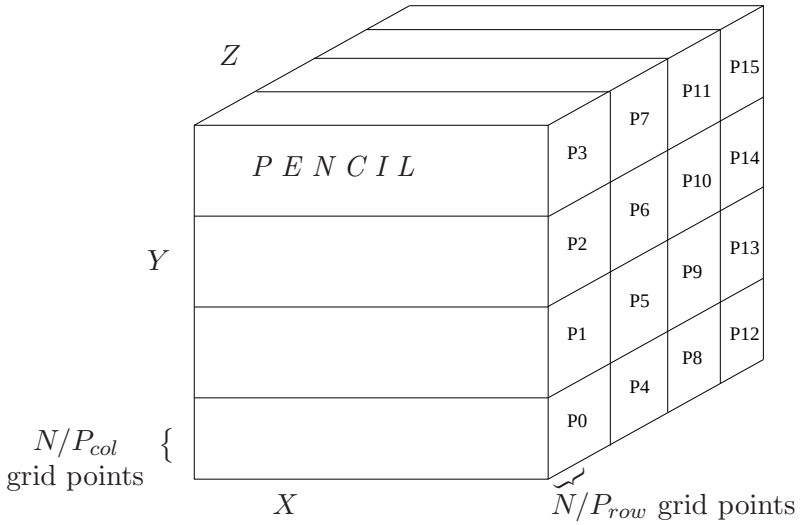

Figure 2.1: Mapping the N^3 domain into a 2D computational grid with $P = P_{row} \times P_{col}$ processors. Here $P_{row} = P_{col} = 4$.

MPI process is referred to as a process), is shown in Fig. 2.1. The actual mapping of the physical grid onto the 2D computational grid containing P processors is done using an MPI intrinsic function (MPI_CART_CREATE) which assigns a column (pencil) of data of size $N \times (N/P_{row}) \times (N/P_{col})$ to each process as shown in Fig. 2.1. The grid parameters P_{row} and P_{col} can be specified by the user to get optimal performance. As a result of the 2D decomposition, only data along a single direction is local to each process at any given time. Thus in order to perform a serial FFT along the three orthogonal directions, two global data transposes are required.

As mentioned previously, the 3D FFT computations which is the most time-consuming part of the algorithm is performed using a transpose-based FFT as opposed to a distributed FFT (Dubey & Tessera 2001). Distributed FFT relies on a parallel implementation of the 1D-FFT with each process communicating the necessary data with the other processes. Transpose-based FFT, on the other hand relies on a sequential version of 1D-FFT that performs the transform on one dimension at a time, transposing the data when needed and is generally considered to be faster than the distributed FFT strategy (Foster & Worley 1997; Dubey & Tessera 2001). In this work a "transpose" refers to (global) data exchange between processors, whereas an internal data movement (within a processor) is designated

as a "local transpose" to avoid confusion. Since the algorithm uses a 2D data decomposition pattern, each process first performs a 1D-FFT along the direction local to the process (say x-axis). Then a global transpose is carried out to align the pencils in the z-direction, followed by a local 1D-FFT along the z-axis. Finally, the data are transposed to pencils in the y-direction, followed by a 1D-FFT in this direction. A schematic of this procedure is shown in Fig. 2.2. Two transposes are taken, one on each of the two orthogonal sub-communicators namely, *row* and *col*. For example, in Fig 2.1, processes P_0, P_4, P_8 and P_{12} belong to the *row* sub-communicator, while P_0, P_1, P_2 and P_3 belong to the *col* sub-communicator. Each process belongs to one *row* sub-communicator and one *col* sub-communicator. Thus the algorithm confines the communication to processes in small local sub-communicators instead of communicating with all processes in the system.

The all-to-all operation requires exchange of non-contiguous data between all process-pairs involved in the communication. Hence we re-arrange the data into chunks (the number of chunks are equal to the number of processes taking part in the all-to-all) such that a given chunk contains all the data that is required to be sent to a particular process. Internal array transposes are required to perform this re-arrangement and the procedure is designated as *packing*. Packing the data manually as opposed to using a MPI intrinsic function such as MPI_PACK or using MPI's derived datatypes, allows us to optimize the packing operation by using loop blocking/tilling techniques. Loop blocking is an optimization technique which breaks down a given number of iterations into smaller blocks, such that the data required for a given block of iterations can completely fit into the cache (a memory space assigned to each core that has very fast access times as compared to the main memory access time), thus speeding up the operations. After the completion of the all-to-all we again have to *unpack* the data at each process to ensure that data corresponding to each variable (such as u_1, u_2, u_3) has unit stride. Stride is the number of memory locations between array elements that are successively accessed in the algorithm. Using a unit stride helps speed up the computations (such as FFTs) due to a decrease in the number of cache-misses. Both packing and unpacking operations require additional memory. An all-to-all on a given array requires temporary storage with count at least equal to that of the array being transposed.

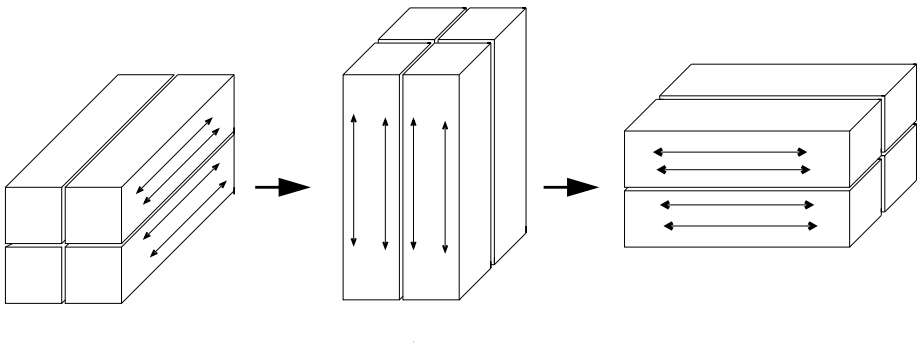

Figure 2.2: 1D FFT in each direction followed by a transpose for a 2D computational grid. Two global transposes are needed to complete a 3D FFT in this case. Here $P_{row} = P_{col} = 2$.

The local 1D-FFT performed within each pencil uses FFTW (Frigo & Johnson 2005) and is of $O(N \log N)$ complexity for a transform of length N. For a N^3 grid, N^2 such transforms are required resulting in a computational complexity $\sim O(N^3 \log N)$. The FFT computations are highly optimized partly because the arrays being transformed are of unit stride. The main bottleneck is the inter-processor communication required to perform the data transposes, in between local Fourier transforms, which are typically done using all-to-all type MPI routines. The scalability of global data transposes is known to degrade with increasing problem size and core count, especially at the Petascale level where hundreds of thousands of cores are involved in the communication (Kumar *et al.* 2008; Pekurovsky 2012). For most MPI implementations the all-to-all routines are implemented as a series of point-to-point communications operations, with each process sending and receiving data to/from every other process in the communicator. The number of messages that need to traverse the same network link in the interconnect increases quadratically with the communicator size (the number of processes involved in the communication) causing network congestion resulting in poor scalability with increasing problem size (Chan *et al.* 2008). Furthermore, the all-to-all performance is sensitive to the topology of the interconnect used in the parallel system, network latency, placement of the processes in the network (which cannot be ordinarily controlled for most Cray systems like Blue Waters) and other network traffic present in the system during run-time. While it is difficult to influence external factors such as network traffic, latencies and job placements, the communicator sizes can be controlled by changing the P_{row} parameter for a N^3 grid using P processes, such that

$P_{col} = P/P_{row}$. We have found that a small value of P_{row} (typically less than or equal to the number of cores within a node) to be optimal.

In order to understand the impact of the $2D$ computational grid parameters such as P_{row} on the communication characteristics of the algorithm, it is instructive to quantify the communication workload. In the following we give estimates for message sizes involved in the all-to-all communications in both the *row* and *col* transposes for a N^3 grid using $P = P_{row} \times P_{col}$ processes. The communication characteristics of both the forward and backward transposes are the same and do not require any distinction. The message count per process for the *row* transpose is N^3/P. This transpose consists of N/P_{col} MPI_alltoall's over the *row* sub-communicator with the message count per process-pair m_{row} being defined as,

$$m_{row} = \frac{N^2}{P_{row}^2} \, . \tag{2.19}$$

Breaking down the all-to-all into N/P_{col} stages is primarily done to minimize memory requirements that arise due to the need for allocating temporary buffers for the transposes as explained earlier. Using a small P_{row} (or equivalently a large P_{col}) ensures that as few all-to-all's as possible are performed with as large a message size as possible being sent across the network in each all-to-all, thus saturating the network bandwidth and leading to better performance. The *col* communicator transpose consists of a single MPI_alltoall with the message count per process being N^3/P, which is the same as that for the *row* transpose. The corresponding message count per process-pair is m_{col}, where

$$m_{col} = \frac{N^3}{P^2} P_{row} \, . \tag{2.20}$$

The message count for the *col* communicator is less than the total message count for the *row* all-to-all which is equal to $(N^3/P^2)P_{col}$, since $P_{row} \ll P_{col}$ and hence the *col* communication is done in a single all-to-all. In contrast to the *row* communication, the *col* communication is seen to improve with larger P_{row} (Eq. 2.20), due to an increase in message size, thereby enabling the all-to-all to achieve peak bandwidth. However, a change in P_{row} has a milder effect on the *col* communication than the *row* communication due to the quadratic dependence of m_{row} on P_{row} as opposed to the linear variation of m_{col} with

P_{row} (see Eqs. 2.19 and 2.20). Furthermore, the improvement in the *col* communication gained by using a large P_{row} is offset by the worsening load imbalance that a large P_{row} leads to (as explained below). Hence we use a small P_{row} (less than or slightly greater than the number of cores in a node) for optimal performance. This means that the processes involved in the *col* communication are not topologically adjacent in the network, which means that the messages have to travel more links to reach their destination, leading to network contention. Alternatively, the *col* communicator processes exchange smaller messages per pair (see Eq. 2.20) and exchange such short messages with a large number of processes (since P_{col} is large) leading to an under-utilization of the network bandwidth. Hence the *col* communicator transposes are the chief bottleneck in the algorithm.

The all-to-alls in the *col* sub-communicator are the main bottleneck in the algorithm. Dealiasing presents an oppurtunity to decrease the message sizes in the *col* communicator transpose. A necessary condition for the spherical truncation relation (see Eq. 2.9) to hold is given by the following inequality,

$$(k_x/N)^2 + (k_z/N)^2 \quad \leq \quad 2/9 \,, \tag{2.21}$$

since, if this condition is not satisfied, then the spherical truncation condition (Eq. 2.9) will not be satisfied for any k_y. The Fourier modes that do not satisfy this conditions need not be considered, thus enabling a reduction in both the computation and communication workload.

Figure 2.3 shows a slice of the k_x-k_z plane where the cylindrical truncation (Eq. 2.21) is enforced (in Fourier space). All points outside the half-cylinder are skipped leading to a decrease in the messages sizes of the *col*-communicator alltoall, the number of computations performed in Fourier space and the memory sizes of the temporary buffers needed for the transposes. This technique was developed and implemented in the parallel algorithm by Dr. Dmitry Pekurovsky, a consultant at San Diego Supercomputer Center (SDSC). With the use of the cylindrical truncation, the message count per process in the *col* communicator all-to-all decreases from N^3/P to approximately $(\sqrt{2}/3)N^3/P$ which is almost a 53% decrease

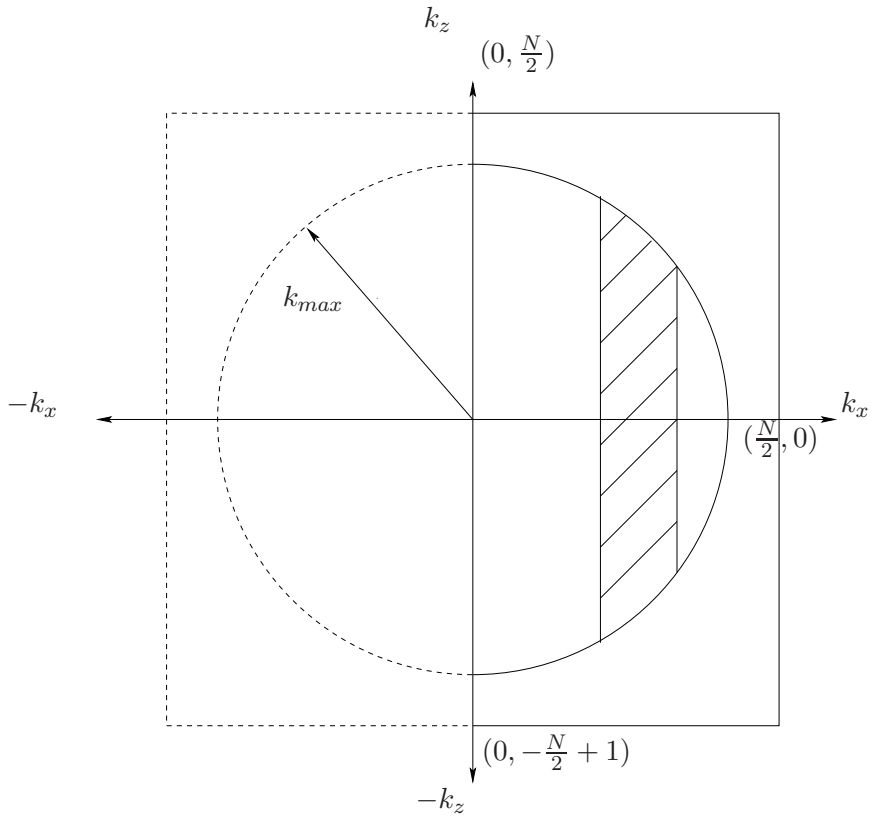

Figure 2.3: Schematic showing the cylindrical truncation in the k_x-k_z plane. Only points inside the half-cylinder with radius $k_{max} = \sqrt{2}N/3$ are considered. The left half-plane (shown with dashed lines) is not considered due to conjugate symmetry. Points in the k_x direction are local to each process, while points in the $\pm k_z$ direction within the cylinder are global. The shaded region shows the data contained by a typical process and is approximated as $k_{max}N/P_{row}$. Processes containing points outside the cylinder remain idle causing a load imbalance.

in the volume of data sent across the network. Furthermore, the message count per process-pair in the *col* communicator can be approximated as,

$$m_{col}^{tr} = \frac{\sqrt{2}}{3}\frac{N^3}{P^2}P_{row} \ , \tag{2.22}$$

which is less than m_{col} (see Eq. 2.20) by 53%. The cylindrical truncation also leads to a 30% decrease in the number of computations required in Fourier space. However, a load imbalance is created with the cylindrical truncation strategy as each process operates on a different number of Fourier modes. From Fig. 2.3 it can be seen that the number of points inside the cylinder, contained by different processors can be different, with processes that contain points in the region bounded by lines $k_x = k_{max}$ and $k_x = N/2$ containing no points inside the half-cylinder. In the algorithm such processes do not take part both in computations (in Fourier space) and the all-to-all communications in the *col* sub-communicator. The load imbalance increases with increasing P_{row} as more number of processes will contain data points outside the cylinder. On the other hand, using a small P_{row} ensures that this load imbalance is minimum thus giving a better overall performance, in spite of the persistent load imbalance present.

As mentioned previously the extent of performance gains obtained by using a small P_{row} strongly depends on the topology of the interconnect used by the parallel system. A small P_{row} has a favorable impact on the performance on the Blue Waters system which implements a 3D torus topology with wrap around links. Performance gains in similar algorithms, using a small P_{row} has been reported on other systems such as the Blue Gene/L supercomputer which also uses a 3D torus network (Chan *et al.* 2008). On the other hand, the dependence on a small P_{row} seems to be weaker on machines such as Stampede which employs a fat-tree topology. Another factor that influences the performance characteristics of the algorithm is the network traffic caused by other jobs running on the system. Tests conducted by other researchers in our group on closed sub-domains of the Blue Waters machine so as to avoid network contention due to external traffic also find that a small P_{row} is beneficial.

An important feature of the parallel DNS algorithm which also influences the communicator sizes is that the number of FFT transforms (and therefore the number of all-to-alls) in the *row* and *col* sub-communicators are not equal. As explained below, $7 + 3N_c$ and $8 + 5N_c$ transposes are required along the two orthogonal sub-communicators respectively in each step of the Runge-Kutta scheme for the three velocity field components and N_c scalars. Hence the communicator along which a greater number of transposes are required, is designated as the *row* communicator. Thus using a small P_{row} ensures that a majority of the transposes are performed among topologically adjacent cores, resulting in best possible performance. The unequal number of transposes required along the two communicators is designed with the aim of performing as few transposes as possible in the algorithm. In the following, a brief account on how a particular non-linear term is formed in physical space and transformed back into Fourier space so as to minimize the number of transposes required, is given.

To explain the reasons behind the unequal number of transposes performed along the *row* and *col* communicators we follow the computation of a typical non-linear term such as $\hat{G}_1$ (see Eq. 2.8). We start with y pencils in Fourier space, in which the variables are functions of (k_y, k_z, k_x) where the data along y direction is local to each process. We transform $\hat{u}_1$, $\hat{u}_2$ and $\hat{u}_3$ along the y direction into functions of (y, k_z, k_x). Next we transpose the data to obtain z pencils along (say) the *col* communicator, followed by a transform in z direction. We then perform the second transpose along the *row* communicator in order to do the final transform along the x direction to transform $\hat{u}_1$, $\hat{u}_2$ and $\hat{u}_3$ into u_1, u_2, u_3 which are functions of (x, z, y). In total we require three transposes each in the *col* and *row* communicator for the forward transform of variables $\hat{u}_1$, $\hat{u}_2$ and $\hat{u}_3$. We now form terms u_1^2, $u_1 u_2$ and $u_1 u_3$ that are needed to calculate $\hat{G}_1$.

After forming the non-linear terms in physical space, we perform a backward transform along the x direction followed by three *row* communicator transposes to obtain the data as z pencils. The z pencils are now transformed back to Fourier space along the z direction, to obtain the non-linear terms as functions of (k_z, k_x, y). At this stage, we multiply the partial transforms of u_1^2 and and $u_1 u_3$ with ik_x and ik_z respectively, sum them up and transpose

the resulting sum along with the partial transform of $u_1 u_2$ in a *col* communicator transpose to eventually compute $\hat{G}_1$. In essence we need six transposes in the *row* communicator and five transposes in the column communicator. The mixed terms (such as $u_i \partial\phi/\partial x_i$ needed in the evaluation of the advective term in Eq. 2.18) are similarly computed by combining terms and transposing as few variables as possible.

Thus by using a small P_{row}, we can place the processes in the *row* communicator inside a single node or topologically adjacent to each other in the network causing the majority of the transposes to be completed quickly. Intra-node MPI communication does not require the messages to travel through the network links thus easing network congestion, resulting in fast transposes. Using a small P_{row}, typically less than or equal to the number of cores in a node is seen to give the best performance. A small P_{row} results in a large P_{col} for a fixed number of processes, causing the the *col* communicator processes to be spread out in the network. As a result, the all-to-alls in the *col* communicator are the main communication bottleneck in the algorithm. Furthermore, the 2D processor decomposition and the conjugate symmetry requirement of the real-to-complex 3D-FFT together require the following conditions on the P_{row} and P_{col} parameters,

$$P_{row} \times P_{col} = P, \quad 2P/N \leq P_{col} \leq N, \quad P/N \leq P_{row} \leq N/2 \,, \tag{2.23}$$

for a N^3 grid using P processes. A value of $P_{row} = 1$ (possible only when $P \leq N$) corresponds to a slab decomposition and requires only one transpose. A 1D slab domain decomposition gives better performance for $P < N$, with the difference between 1D and 2D versions becoming smaller as P is increased up to $P = N$, beyond which the 1D case does scale (Pekurovsky 2012). The smallest value that P_{row} can take is P/N since P_{col} cannot exceed N. In practice we use $P_{row} = 4$ as a lower bound for P_{row}.

To summarize, we use an algorithm that uses a 2D grid decomposition technique which decomposes the N^3 domain into as many as N^2 processors, with the aim of running larger and larger simulations on massively parallel machines. The algorithm uses a transpose based FFT strategy in order to perform the 3D FFTs which continues to be the main bottleneck, taking up as much as 80% of the overall run-time for very large problem sizes. A number

of strategies such as optimal communicator sizes, unit stride array operations, loop blocking/tilling and cylindrical truncation techniques have been developed with active help from consultants at supercomputing centers to make the algorithm more efficient. However, one strategy that has not yet been adequately tested is the use of OpenMP threads (Chapman *et al.* 2007) in conjunction with MPI processes. The use of OpenMP threads along with MPI processes is popularly known as hybrid programming and has been widely reported in the literature (Tsuji & Sato 2009). The use OpenMP threads results in a reduction in the number of processes (for a fixed number of processors) and can potentially help ameliorate both memory and network latency issues. More importantly, such a programming model may help overlapping a part of the communication with the computation, with the aim of masking at least some of the communication time. A discussion on such an attempt is the subject of the next chapter.

CHAPTER III

HYBRID MPI/OPENMP ALGORITHM

3.1 Introduction

The number of processing units (or cores) in a given computing unit (or node) has seen a steady increase over the past decade, accompanied by a decrease in the memory exclusive to each core. This trend towards symmetric multi-processor (SMP) systems, where multiple processors share a centralized shared main memory in addition to possessing a secondary memory that is private to each core (also called "cache"), is mainly due to power considerations. The advent of SMP systems has been accompanied by the rise of shared memory programming protocols like openMP (Chapman *et al.* 2007) in which threads use the same main memory thus avoiding the need to communicate with one another. Consequently a hybrid programming paradigm that uses MPI to communicate across nodes and openMP threads to share work within a node has been gaining popularity (Tsuji & Sato 2009). In particular, openMP used in conjunction with MPI can be used to overlap computation with communication resulting in better scalability of communication dominated, all-to-all type algorithms (Doi & Negishi 2010). We have attempted to implement the hybrid MPI/openMP strategy in our parallel algorithm with the aim of improving the code's scalability at very large problem sizes and very high core counts by masking a part of the communication time behind the computation time. This chapter gives a detailed account of the various hybridization strategies attempted in this study along with the scaling results comparing the MPI/openMP hybrid model with other communication paradigms.

3.2 Memory locality in Hybrid MPI/openMP implementations

In order to obtain significant performance improvement using threads, it is important that the threads be allowed to work independent of each other as much as possible. Thread synchronization which is typically achieved in openMP using barriers and locks, increases

both the overhead costs and the time spent by threads idling. Another thread synchronization construct in openMP is the *flush* construct which when encountered forces threads to update their copy of the shared variables from/to the main memory. Although the use of the *flush* construct at times is essential to ensure correctness of the program, it can lead to performance degradation due to the time spent in reading from or writing to the main memory which is slower than the write and read times for the cache (Chapman *et al.* 2007). Thus the use of flush statements has to be done judiciously to ensure memory locality and optimal performance. Memory locality can also be preserved by using the "principle of first touch". That is, whenever a shared array is allocated, it is not written to the cache space of a thread, until the thread writes some data to it. Once a shared array is allocated, the threads can initialize that portion of the array on which they will be working, merely by assigning the respective portions of the array to some arbitrary value. This initialization will ensure that copies of the concerned portions of the shared array are written into the cache-space of the respective threads enabling the threads to work with their local copies. Now the threads can work with these temporary copies and write them back to the main memory only when needed. Hence maintaining memory locality is essential to avoiding slow read and write times from and to the main memory. In general, lesser thread synchronization and greater locality are key features that govern the way in which good openMP algorithms are designed.

3.3 Hybrid MPI/openMP paradigms

Since we are interested in using both MPI processes and openMP threads, we are concerned with how the processes and threads interface with each other. The main thread that initiates the MPI program is called the master thread, while the other threads spawned by the master thread in the thread parallel region are called the worker threads. The MPI standard allows four levels of thread-safety which specify the interaction between MPI processes and user-defined threads (Gropp & Thakur 2006) and are as follows, (i) thread-single: only one thread will execute (ii) thread-funneled: a process can spawn multiple threads but only the master thread performs the MPI communication, (iii) thread-serialized: any thread can perform

MPI communication although two distinct threads cannot make MPI calls concurrently, (iv) thread-multiple: multiple threads can make MPI calls with no restrictions. The thread-single model is the same as the MPI-only model and is designated as "pure MPI" in the following.

The thread-funneled model allows each process to be multi-threaded but limits the MPI communication only to the master thread. As a result of using threads, the number of MPI processes decreases for a given core count, which can potentially decrease the communication latency and help maximize the communication bandwidth. Since only the master thread performs the MPI communication, thread synchronization can be achieved easily achieved using a barrier. However, obtaining performance improvements using overlap between communication and computation is tough since it entails data copy to and from the main memory by the master and worker threads which disrupts memory locality. Hence, we let the worker threads idle while the master thread performs the MPI communication.

In the thread-serial model all threads can perform MPI communication, although not simultaneously. Similar to the thread-funneled model, the thread-serial model benefits from the decrease in the number of MPI processes taking part in the communication at any given time, thus ameliorating latency issues. In addition, the thread-serial model lends itself to overlapping communication with computation in a way so as to preserve memory locality and avoid shared memory access to a much greater extent than in the thread-funneled case. When using the thread-serial model, it is the user's responsibility to ensure that only one thread participates in the MPI communication at any given time. Furthermore, in the all-to-all calls needed in our algorithms, the thread with the same thread-id across all processes has to participate in the all-to-all communication to preserve correctness. In order to ensure that the threads enter the region in an orderly fashion, we use simple locks constructed using *flush* and *go to* statements to ensure an orderly entry into the all-to-all region.

The thread-multiple model is the most general of the threaded hybrid models with no restrictions on the communication patterns of the threads. The advantage with this model is that the user does not have to deal with thread synchronization as is the case with the

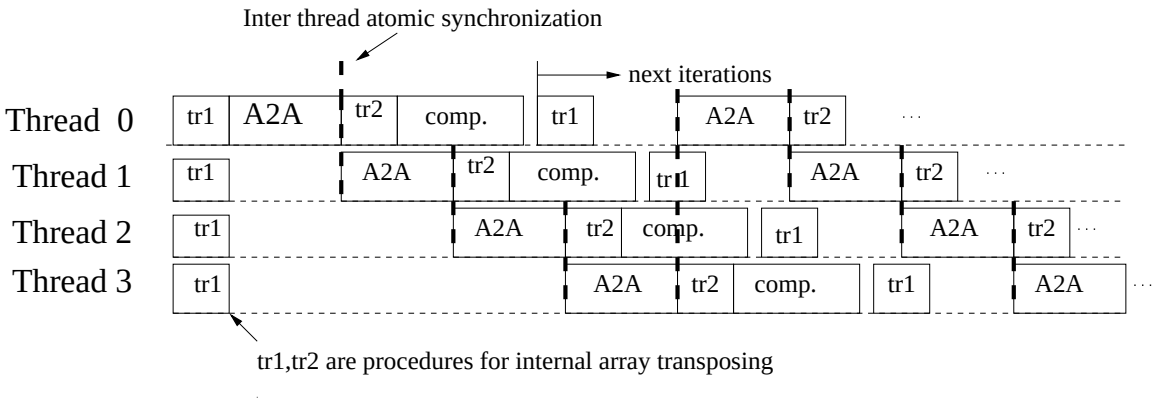

Figure 3.1: Pipelining of all-to-all (A2A) communication using four threads. To pack the appropriate data into send and receive buffers, two internal array transposes are performed, denoted here and henceforth as $tr1$ and $tr2$.

other two models, but the drawback is that, while doing MPI communication all the threads simultaneously contend for the same MPI resources since the MPI buffer size for each node is fixed. Thus, the thread-multiple model suffers from the same latency issues as the pure MPI model and is generally not considered to be a high performance option. Furthermore, the threads in the thread-multiple model communicate simultaneously, thus leaving little room for overlapping communication with computation.

3.4 Hybrid MPI/openMP thread-serialized implementation

Figure 3.1 shows scheduling of the threads in the the thread-serial model, done in order to obtain overlap between communication and computation for a case using four threads. The data is divided among the threads in order to obtain a pipe-lined procedure as shown in Fig. 3.1 so as to process the independent messages sequentially. Here, $tr1$ and $tr2$ refer to internal array transposes, which are needed to pack and unpack data from *send* and *receive* buffers, before and after the all-to-all region, respectively. The inter thread atomic synchronization in Fig. 3.1 refers to the thread synchronization that is needed to ensure an orderly entry into the all-to-all region. The threads are scheduled so that only one thread at a time performs the all-to-all communication with the help of simple locks constructed from *flush* and *go to* statements. The first thread starts its all-to-all as soon as it is done packing its outgoing message. Once this is completed it can proceed with other operations such as unpacking data and computations while another thread performs its all-to-all and similarly

Table 3.1: 4096^3, 32768 cores - timing data (in seconds) for the best $P_{row} \times P_{col}$ values. The timing data (Blue Waters) correspond to the MPI process that takes the maximum time. Number of variables transformed is 5. Repeat count of four is used.

	$P_{row} \times P_{col}$	no. of threads	FFT	internal transposes	all-to-all	total
pure MPI	16×2048	1	0.4	0.3	6.2	6.9
thread-serial	8×2048	2	0.4	0.4	4.7	5.5
thread-serial	8×1024	4	0.3	0.1	5.7	6.5
thread-multiple	16×1024	2	0.3	0.1	7.6	8.1
thread-funneled	8×1024	4	0.5	0.4	6.5	7.4

proceeds with other operations thus enabling another thread to continue the sequence. This leads to a pipe-lined procedure where the main idle time for all the threads is the time taken for the first all-to-all by the master thread.

3.5 Weak scaling analysis of 3D FFT

In order to study the effect of various communication paradigms on our parallel algorithm, we have isolated the main communication characteristics of the algorithm in the form of a 3D FFT kernel. The 3D FFT kernel performs the backward and forward transform of a 3D signal in a fashion similar to that in the DNS algorithm and is called the FFT kernel. We have developed implementations of the FFT kernel for each of the three thread-hybrid models. The goal is to compare the performance of the FFT kernel for the hybrid MPI/openMP models with that of pure MPI and Co-Array Fortran models. The data in Tables 3.1-3.3 below report times for the Blue Waters machine taken for the main operations involved in the 3D FFT operations which are the serial FFT (using FFTW), all-to-all global transpose and internal data transposes required for packing and unpacking buffers for the all-to-all operation. The "total" time taken in Tables 3.1-3.3 is the maximum time taken among all the MPI processes for executing a 3D FFT from physical space to Fourier space and back. The data corresponding to all-to-all, FFT and internal transposes give the breakdown of the time spent by the MPI process which takes the maximum "total" time in each of these operations.

From Tables 3.1-3.3 it can be seen that the thread-serial model seems to be faster than

Table 3.2: 2048^3, 4096 cores - timing data (in seconds) for the best $P_{row} \times P_{col}$ values. The timing data (Blue Waters) correspond to the MPI process that takes the maximum time. Number of variables transformed is 5. Repeat count of five is used.

	$P_{row} \times P_{col}$	no. of threads	FFT	internal transposes	all-to-all	total
pure MPI	32×128	1	0.3	0.3	2.1	2.7
thread-serial	4×512	2	0.3	0.2	1.6	2.3
thread-serial	8×128	4	0.3	0.1	1.9	2.6
thread-multiple	16×128	2	0.3	0.1	2.1	2.6
thread-funneled	16×128	2	0.4	0.3	2.3	3.0

Table 3.3: 1024^3, 512 cores - timing data (in seconds) for the best $P_{row} \times P_{col}$ values. The timing data (Blue Waters) correspond to the MPI process that takes the maximum time. Number of variables transformed is 5. Repeat count of five is used.

	$P_{row} \times P_{col}$	no. of threads	FFT	internal transposes	all-to-all	total
pure MPI	8×64	1	0.3	0.3	1.2	1.8
thread-serial	4×64	2	0.2	0.2	0.9	1.5
thread-serial	2×64	4	0.3	0.2	1.2	2.0
thread-multiple	4×64	2	0.3	0.2	1.1	1.6
thread-funneled	4×64	2	0.4	0.4	1.2	2.0

the others. The thread-serialized model appears to out perform the pure MPI code by roughly 20% for the 4096^3 problem size with a significant reduction in the all-to-all time. While the thread-multiple and thread-funneled models seem to fare worse than the pure MPI model with increasing problem size. The difference between the "total" column and the sum of the all-to-all, FFT and internal transposes columns is greatest for the thread-serial and thread-multiple versions. This is because all the threads have to wait until the master thread initiates the pipeline communication for the thread-serial case. In the thread-multiple case this may be due to the use of barriers needed to ensure correctness. Furthermore, in both the thread-serial and thread-multiple models, at the end of the algorithm, all threads have to wait for the last thread to complete its work. In spite of this idle time, the thread-serial model seems to be giving better results than the other models.

3.6 Discussion

In this work, results from weak scaling experiments where the ratio of the grid size to the number of MPI processes (N^3/P) is held constant, have been reported. The data show that the thread-serialized MPI implementation improves the weak scaling results by as much as 30% when compared to the pure MPI version. However, it is also clear that the performance of the alltoall degrades significantly with increasing core count for any given implementation. It should be noted that global communication patterns such as the alltoall are bound by the bisection bandwidth of the interconnect (Czechowski *et al.* 2012). The bisection bandwidth can be defined as the lowest bandwidth through any cross-sectional area of the network. The bisection bandwidth for a 3D torus network (the interconnect on Blue Waters) is $O(P^{2/3}\beta_{link})$ where β_{link} is the link bandwidth of the interconnect. The weak scaling of the 3D FFT kernel as a function of $P^{2/3}$ has been analyzed (although not shown) and the results do not show any appreciable difference. The time required to perform the alltoall exchange for a 3D FFT on a 3D torus network can be written as (Czechowski *et al.* 2012)

$$T_{comm} \approx 2\frac{N^3}{P^{2/3}\beta_{link}} \,, \tag{3.1}$$

where the factor two accounts for the alltoall in the two orthogonal sub-communicators. Although, this equation provides an estimate of the theoretical alltoall time, it has not been used in the analysis given in this work. Clearly, a more rigorous approach to the alltoall problem should consider the bisection bandwidth of the nodes. The results given in this chapter can be considered as the near-ideal weak scaling limit of the alltoall since it requires a constant bisection bandwidth per node. Alternatively the present analysis requires the bisection bandwidth $\sim P$. Noting these limitations and the near-idealness of the weak scaling analysis presented here, a more rigorous analysis of the performance of the alltoall communication paradigm is deferred for the future.

CHAPTER IV

TURBULENT MIXING

It is well known that turbulence expedites scalar mixing through the advection process, but the exact mechanics of this process is still less than clear. Several issues such as the status of local isotropy and small-scale universality of the scalar field remaining less unequivocal than that for the velocity field (Sreenivasan 1991; Sreenivasan & Antonia 1997). A systematic study of the scalar mixing warrants the classification of passive scalars on the basis of the Schmidt number ($Sc = \nu/D_\phi$, where ν is the kinematic viscosity and D_ϕ is the diffusivity of scalar ϕ), being low ($Sc < 1$), moderate ($Sc \sim O(1)$) or large ($Sc > 1$). There exists little available data in the literature on turbulent mixing in the low Schmidt number regime as compared to the other two, partly due to difficulties in experiments involving liquid metals. The study of the effects of turbulence on strongly diffusive scalars is not only important from a theoretical standpoint (to obtain a unified theory of scalar mixing) but also has applications in areas such as solar-wind MHD turbulence and electrostatic turbulence (Sorriso-Valvo *et al.* 2007; Lepreti *et al.* 2009).

Our group has conducted numerical simulations of mixing at Schmidt numbers as low as 1/2048 at Taylor-scale Reynolds numbers up to 390. Results in the case of decaying scalar fields without mean gradient (Yeung & Sreenivasan 2013) have provided strong support for the classical theory of Batchelor *et al.* (1959) that the scalar spectrum displays a $k^{-17/3}$ behavior under inertial-diffusive conditions. Examination of simulations with a uniform mean gradient (Yeung & Sreenivasan 2014) also suggests a $-17/3$ power law, but with a different prefactor, which can be traced to a reduction in spectral transfer as the range of scales in the scalar field becomes substantially narrower due to decreasing Schmidt number at a finite Reynolds number. On the basis of the above-mentioned findings differences can be expected between the low Schmidt number regime and the moderate-to-high Schmidt number regimes in issues such as local isotropy, small-scale universality and intermittency.

Computationally, for a given grid spacing, a low Schmidt number simulation has much stricter temporal resolution requirements than that in the other two regimes as fast molecular diffusion results in very small time scales. This requires the time step to be much smaller than that required by numerical stability constraints (Yeung & Sreenivasan 2013). The use of very small time steps makes the use of double precision arithmetic important in preventing round-off errors accumulating over time, which in turn increases both computation and inter-processor communication times. Spatially we require a larger solution domain to minimize the possible constraining effects of boundary conditions since the scalar integral scales grow with decreasing Schmidt number (Donzis *et al.* 2005). This gives the auxiliary benefit of improved statistical sampling for the scalar field, which is affected by the stochastic forcing of the velocity field (Eswaran & Pope 1988) and the imposed mean gradient in the scalar field (Overholt & Pope 1996), that are used to sustain the scalar fluctuations.

We focus on the scalar field structure by investigating the statistics of the spatial increments of velocity and scalar fluctuations with an emphasis towards the low Schmidt number regime. A two point equation which describes the scale separation in the scalar field is examined in detail with an analysis of the different mechanisms involved at various scale sizes, for a range of Schmidt and Reynolds numbers. The study of (joint) statistics of the scalar and velocity field fluctuations provides important information on local isotropy and intermittency as a function of scale size. The coupling between scalar gradients and velocity gradients (or the strain rates) which determines the rate at which the scalar gradients and hence the scalar dissipation fluctuations may become amplified by turbulence is also of interest. While this coupling as been studied in DNS before (Ashurst *et al.* 1987; Vedula & Yeung 1999), if spectral transfer is suppressed at low Schmidt number, then it is likely that nonlinear amplification would be weaker, thus attenuating the instantaneous scalar dissipation rate. Accordingly, the balance of terms in the scalar dissipation equation is studied in the low Schmidt number limit along with the geometric alignment between scalar gradients and strain rates.

In this chapter issues noted above are addressed by drawing upon a large DNS database

of passive scalar mixing in isotropic turbulence. Results from simulations of stationary mixing with scalar fluctuations maintained by a uniform mean gradient are reported. The work described in this chapter forms the core of a newly submitted journal article (Iyer & Yeung 2014). The rest of this chapter is organized as follows. In Sec. 2 we give a brief discussion about the intermediate and small scale ranges in the velocity and scalar fields. We also introduce the scalar structure function equation. In Sec. 3 we consider the procedures necessary to compute the balance of terms in the scalar structure function equation. In Sec. 4 we give a short summary of the characteristics of the simulation data sets analyzed in this chapter. In Sec. 5 we present statistics of scalar increments and scalar gradients, and of couplings between scalar gradients and principal strain rates, at different Reynolds number and Schmidt number combinations. In Sec. 6 we return to our analysis of the structure function budget, along with the validity conditions for obtaining a simpler form in the intermediate scale range. In particular, we study how individual terms in the structure function equation vary with Reynolds number and Schmidt number, and in comparison with one another. Appropriate discussion is given for challenges in resolution Donzis & Yeung (2010) for scalars of moderate and high Sc, and in sampling for scalars of very low Sc. The conclusions of this chapter are summarized in Sec.7.

4.1 Scalar field structure

Spatial structure of the scalar field in turbulence can be described by using the products of two-point differences of a scalar quantity and those differences multiplied with the difference of velocity and other quantities. Indeed, many theories of similarity scaling (see eg. Frisch 1995; Sreenivasan & Antonia 1997) have focused on the averages of such products called structure functions. For turbulence at sufficiently high Reynolds number it is well known that Kolmogorov (1941a) theory predicts the m-th order structure function of the longitudinal velocity increment over a distance r in the inertial range ($\eta \ll r \ll L_1$, where η is the Kolmogorov length scale and L_1 is the integral scale) would scale as $(\langle\epsilon\rangle r)^{m/3}$ where $\langle\epsilon\rangle$ is the mean energy dissipation rate. Although results of K41 are subject to intermittency corrections related to intense fluctuations of the energy dissipation rate, the third order

moment ($m = 3$) has special significance. In particular,

$$\langle (\Delta u_L(r))^3 \rangle = -\frac{4}{5}\langle \epsilon \rangle r \ , \tag{4.1}$$

where $\Delta u_L(r) = \Delta u_j(\mathbf{r}) r_j / r$ is the longitudinal velocity increment. Equation 4.1 is also known as the four-fifths law and can be derived from the Karman-Howarth equation that governs the second-order velocity structure function. Equation 4.1 is free of any unspecified scaling constants and is devoid of any intermittency corrections (since it is linear in dissipation). As a result it is often used as a criterion in assessing the presence and width of an inertial range in physical space in both experiments and simulations. Figure 4.1 shows the third order velocity structure function as a function of spatial separation for a 2048^3, $R_\lambda \sim 390$ simulation. The third order velocity structure function is strictly negative for the range of scales shown indicating a net transfer of energy down the cascade. The non-zero value of the third moment of the velocity structure function is indicative of the non-Gaussianity of the turbulent velocity fluctuations. The range of r satisfying the inequality $\eta \ll r \ll L$ and Eq. 4.1 corresponds to $30\eta < r < 200\eta$, which can be considered as the width of the inertial range for this particular simulation. The inertial range is expected to become wider with increasing Reynolds number.

The scalar field analog of Eq. 4.1 is the so-called Yaglom (1949) relation for the mixed velocity-scalar third order structure function, namely

$$\langle \Delta u_L(r) (\Delta \phi(r))^2 \rangle = -2\langle \chi \rangle r / 3 \ , \tag{4.2}$$

where $\langle \chi \rangle \equiv 2 D_\phi \langle \nabla \phi \cdot \nabla \phi \rangle$ is the mean scalar dissipation rate and scale size r is in the inertial-convective range (Obukhov 1949; Corrsin 1951). Similar to the four-fifths law, Yaglom's relation suggests that the velocity and scalar fields are jointly non-Gaussian. Previously, DNS data (Yeung *et al.* 2002) for both moderately diffusive ($Sc \sim O(1)$) and weakly diffusive ($Sc \gg 1$) scalars have provided good support for Yaglom's relation as written above. For scalars of $Sc \sim O(1)$ or higher, the concept of dissipative anomaly also appears to hold well (Donzis *et al.* 2005) which means that $\langle \chi \rangle$ is determined by the large scales through a spectral cascade, with little dependence on molecular diffusivity. Since the presence of a wide range of scales is a prerequisite for a strong spectral cascade, Eq. 4.2 is

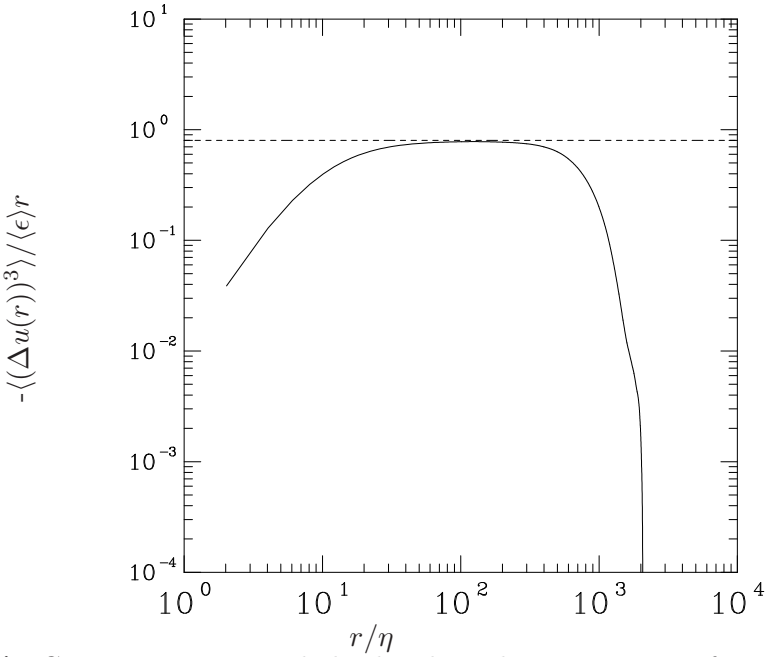

Figure 4.1: Component averaged third order velocity structure function for 2048^3, $R_\lambda \sim 390$, $(4\pi)^3$ simulation. The dashed line is drawn at 4/5 for comparison with Eq. 4.1. The ratio of the largest to the smallest scales is $L_1/\eta \approx 446.98$. The inertial range ($\eta \ll r \ll L_1$) can be taken as $30\eta < r < 200\eta$.

expected to hold well with increase in Reynolds and/or Schmidt number. However in the strongly-diffusive case if the Schmidt number is very low and the Reynolds number is not high enough (such that the Peclet number $Pe \equiv R_\lambda Sc$ is low) the range of scales becomes narrow and a strong spectral cascade is difficult to sustain. Consequently the validity of Eq. 4.2 in the limit of low Sc is uncertain. Indeed, recent work (Yeung & Sreenivasan 2013, 2014) at Sc as low as 1/2048 with emphasis on spectral aspects shows that mixing at very low Sc is fundamentally different in many aspects.

To obtain new insights into the scaling properties of structure functions and to examine more carefully the conditions required for Yaglom's relation it is useful to consider the scalar structure function budget in some detail. For homogeneous turbulence with a uniform mean scalar gradient the budget equation for scalar fluctuation (ϕ) advected by velocity fluctuation component ($u_i, i = 1, 2, 3$) can be written as (Gotoh & Yeung 2013)

$$\frac{\partial}{\partial t}\langle\Delta\phi(\mathbf{r},t)^2\rangle + \frac{\partial}{\partial r_i}\langle\Delta u_i(\mathbf{r},t)\Delta\phi(\mathbf{r},t)^2\rangle \;=\; -2\langle\Delta u_i(\mathbf{r},t)\Delta\phi(\mathbf{r},t)\rangle\frac{d\Phi}{dx_i} \;+\; 2D_\phi\frac{\partial^2\langle\Delta\phi(\mathbf{r},t)^2\rangle}{\partial r_i\partial r_i} - 2\langle\chi\rangle\ , \qquad (4.3)$$

where all terms are functions of a separation vector ($\mathbf{r}$) and time (t). The first and the second terms on the left hand side represent unsteadiness and advection respectively. While terms on the right hand side represent production, molecular diffusion and dissipation. Yaglom's relation can be derived from this equation by integration if unsteadiness, production and molecular diffusion are all negligible over a suitable range of scales, and if the budget becomes dominated by a balance between advection and dissipation terms. If the Yaglom's relation (see Eq. 4.2) does not hold in the low Schmidt number regime, then an important question is, how the balance of terms in Eq. 4.3 depend on Reynolds number and Schmidt number.

Yaglom's relation may also be taken as a specific statement on the joint statistics of velocity and scalar increments in space . More complete information on the properties of these increments can be obtained from the probability distribution of scalar increments at various scale sizes, as well as the joint distribution of velocity and scalar increments. If the distance r is sufficiently small, standard Taylor-series arguments relate two-point statistics of the spatial increments to one-point statistics of the gradients. Consequently the study of structure functions (and mixed structure functions) also provides important information on local isotropy and intermittency at small and intermediate scales.

In the following sections we address the issues noted above by drawing upon a large DNS database of passive scalar mixing in isotropic turbulence, which has been extended to very low Schmidt numbers. The computation of individual terms in the scalar structure function equation (Eq. 4.3) requires substantial care, which we also discuss. For scalars of very low Sc, at intermediate scales the mixed structure function is found to be much smaller in magnitude than predicted by Yaglom's relation. Scalar gradient fluctuations are found to become poorly correlated with velocity gradients and principal strain rates, while the structure function budget equation becomes dominated by production and dissipation. The importance of production even at intermediate and smaller scales is consistent with the non-universality observed in spectral quantities (Yeung & Sreenivasan 2014), even as the spectrum follows a power law with the same exponent as predicted by classical theory Batchelor *et al.* (1959). Furthermore, the assumption of local isotropy becomes less

valid with decreasing Peclet number.

4.2 Simulation database

We provide a brief summary of the Reynolds and Schmidt number ranges reached in past simulations of turbulent mixing that are analyzed in the following sections. These simulations include those which were intended to (i) reach as high a Reynolds number as possible with Sc held fixed at $O(1)$ (Donzis *et al.* 2005), (ii) reach as high a Schmidt number as possible at moderate Reynolds number (Donzis *et al.* 2010), or (iii) reach as low a Schmidt number as possible at moderate Reynolds number (Yeung & Sreenivasan 2014). Scalar fields in the first two of these three categories of simulations generally follow classical spectral cascade concepts quite closely, but may be limited in resolution at the small scales. In contrast, scalar fields of the third (low Schmidt number) type deviate substantially from classical cascade concepts, and are well resolved the small scales but limited by the finite size of the periodic domain employed.

Table 4.1: Summary of Reynolds number (R_λ), number of grid points (N^3), box length (L_0), Schmidt number (Sc) and resolution ($k_{max}\eta$,$k_{max}\eta_B$) for scalar fields studied in this work.

R_λ	N	$k_{max}\eta$	L_0	Sc	$k_{max}\eta_B$
140	512	1.4	4π	1/8	-
140	512	1.4	4π	1/32	-
140	512	1.4	4π	1/128	-
140	512	1.4	4π	1/512	-
240	1024	1.4	4π	1/128	-
240	1024	1.4	4π	1/512	-
240	1024	1.4	4π	1/2048	-
390	2048	1.4	4π	1/2048	-
650	4096	2.7	2π	1/8	-
650	4096	2.7	2π	1	2.7
140	2048	11.2	2π	4	5.59
140	2048	11.2	2π	64	1.4
240	2048	5.14	2π	1	5.14
240	2048	5.14	2π	8	1.82

Table 4.1 gives a list of basic parameters of the simulation data sets analyzed in this work. Resolution of the velocity field is presented by the dimensionless parameter $k_{max}\eta$ where $k_{max} = (\sqrt{2}N/3)k_0$ is the highest resolved wavenumber and $k_0 = 2\pi/L_0$ is the lowest

wavenumber represented in a domain of length L_0 on each side. The ratio of grid spacing Δx to the Kolmogorov length scale η is approximately $2.96/(k_{max}\eta)$. For Schmidt numbers less than or equal to unity, the smallest scales in the scalar field, denoted by the Obukhov-Corrsin scale η_{OC} (Obukhov 1949; Corrsin 1951) is larger than, or equal to the Kolmogorov length scale (η), since $\eta_{OC} = \eta Sc^{-3/4}$. Hence $k_{max}\eta$ serves as the relevant resolution parameter for the small scales in the case of $Sc \leq 1$. The most expensive aspect of low Schmidt number simulations is the very small time steps which is required to capture the effects of fast molecular diffusion properly ($\Delta t \sim (\Delta x)^2/D_\phi$). A small time step necessities the use of higher precision arithmetic (to prevent round-off errors accumulating) and also results in longer simulation times since significantly more time-steps are needed to reach steady state. In addition, low Schmidt number simulations are expensive because of the need to accommodate larger length scales through a larger domain (Donzis *et al.* 2005). Increasing the box size requires a proportional increase in the number of grid points such that the resolution remains the same. In contrast, simulations with very high Schmidt number are more demanding because of the need to resolve the Batchelor scale $\eta_B = \eta Sc^{-1/2}$ (Batchelor 1959) which (when $Sc > 1$) is smaller than the Kolmogorov scale for the velocity field. Here, $k_{max}\eta_B$ is the appropriate resolution parameter. Because of these challenges, results on mixing at high Reynolds number are available only for scalars of Schmidt number equal to or slightly less than unity. Nevertheless, trends for high Reynolds number turbulent mixing at high and low Schmidt numbers can be deduced using results from high and low Schmidt number simulations at moderate Reynolds numbers.

4.3 Statistics of scalar increments and gradients

4.3.1 Scaling of scalar increments and dissipation

The statistics of scalar increments at any scale size r depends on the range of scales present which in turn depends on the Reynolds number and the Schmidt number, and how r itself is located within this range of scales. In this section we examine the statistics of scalar increments and scalar gradients, specifically the scalar dissipation rate.

If the Reynolds number is sufficiently high then classical theory predicts the presence of

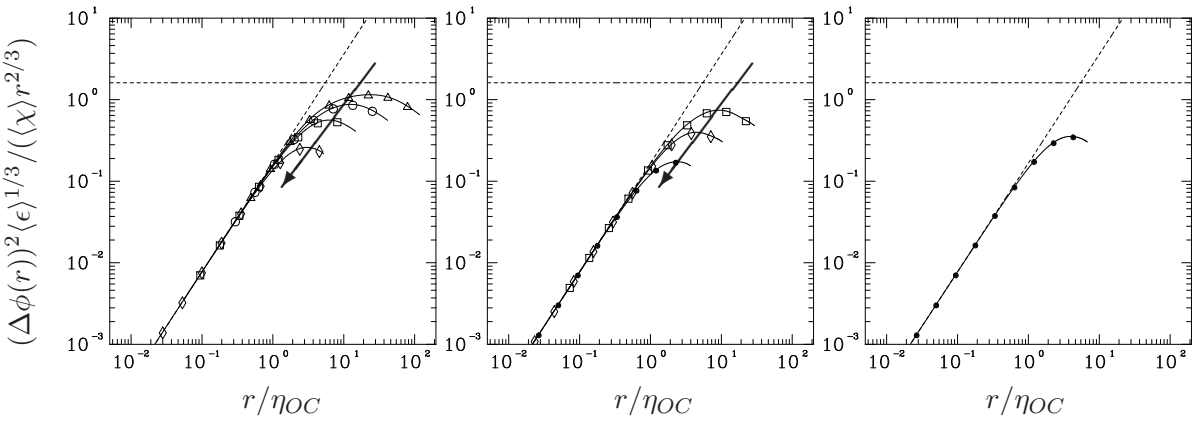

Figure 4.2: Second-order scalar structure function compensated by Obukhov-Corrsin scaling. Frames in the left, middle and right columns are for $R_\lambda \sim 140$, 240 and 390 respectively. Each choice of symbol on the curves denote a different value of Sc: 1/8 ($\triangle$), 1/32 ($\bigcirc$), 1/128 ($\square$), 1/512 ($\diamond$) and 1/2048 ($\bullet$). Arrows point in the direction of decreasing Schmidt number in each frame. Dashed lines indicate slope 4/3 at small scales (Eq. 4.5) and value of the Obukhov-Corrsin constant $C_2 \approx 1.608$ (Eq. 4.4).

an inertial-convective range (Obukhov 1949; Corrsin 1951). The inertial-convective range is the intersection of the convective and inertial subranges and is given by $max(\eta, \eta_{OC}) \ll r \ll min(L_1, L_\phi)$, with η and $\eta_{OC} = \eta Sc^{-3/4}$ being the Kolmogorov and Obukhov-Corrsin scales, and L_1 and L_ϕ being the integral length scales of the velocity and scalar fields respectively. The classical result for spatial separations in this range is given by

$$\langle(\Delta\phi(r))^2\rangle = C_2\langle\chi\rangle\langle\epsilon\rangle^{-1/3}r^{2/3} \ , \tag{4.4}$$

where $C_2 = 4.02\ C_\phi$ with C_ϕ being the Obukhov-Corrsin constant in the one-dimensional scalar spectrum in the inertial-convective range. In previous work (Yeung *et al.* 2005), inertial-convective range behavior has been observed in the scalar spectrum, with a value of C_ϕ close to 0.4 as suggested by a survey of data from laboratory experiments (Sreenivasan 1996). Data on structure functions presented in (Yeung *et al.* 2002) also show a scaling range consistent with where $C_2 = 4.02\ C_\phi$ with C_ϕ taken as 0.4. Unless the Reynolds number are extremely high, η_{OC} may be so large it approaches L_1, such that conditions for inertial-convective scaling cannot be satisfied.

Figure 4.2 shows second-order structure functions normalized by Obukhov-Corrsin scaling variables in the low-Schmidt-number simulations. Each frame of this figure represents a different Reynolds number (increasing from left to right) while each curve represents a

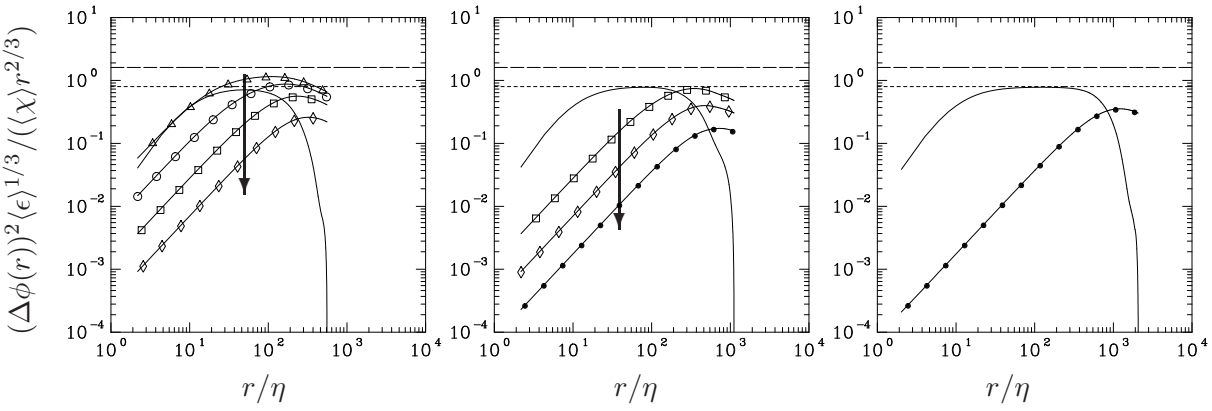

Figure 4.3: Second-order scalar structure function compensated by KOC variables as function of r/η using same choice of symbols as Fig. 4.2 . Arrows point in the direction of decreasing tendency towards an inertial-convective plateau. Solid line in each frame represents the longitudinal third order velocity structure function compensated by $K41$ variables (see Eq. 4.1). Horizontal line (long dashes) shows value of the Obukhov-Corrsin constant $C_2 \approx 1.608$ (Eq. 4.4), while horizontal line (small dashes) at 4/5 for determining inertial range extent.

different Schmidt number (decreasing in the direction of the arrow). As the Schmidt number is reduced to very low values (1/512 for R_λ 140 on the left, and 1/2048 for R_λ 240 and 390) it can be seen that the second-order structure function deviates increasingly from inertial-convective scaling. It can also be seen that the data collapse on an universal line at slope 4/3 for $r/\eta_{OC} \leq 1$. This collapse is related to the result $D_{\phi\phi}(r) \approx (\langle\chi\rangle/6D)r^2$ at small r, derived using Taylor series arguments, which implies

$$\frac{\langle(\Delta\phi(r))^2\rangle}{\langle\chi\rangle\langle\epsilon\rangle^{-1/3}r^{2/3}} = \frac{1}{6}\Big(\frac{r}{\eta_{OC}}\Big)^{4/3} \quad , r \ll \eta_{OC} \; . \tag{4.5}$$

Because η_{OC} becomes considerably larger than both the Kolmogorov scale and the grid spacing the small scales for low Sc scalars are well-resolved.

In order to examine the KOC scaling result more carefully, it is instructive to determine the extent of the inertial range in the flow field using Eq. 4.1. The inertial convective range is a sub-range within the inertial range and hence its limits can be obtained from the four-fifth's scaling (Eq. 4.1). Figure 4.3 shows the KOC normalized second order structure function plotted as a function of the spatial separation, normalized by the Kolmogorov length scale η. Also shown is the normalized third order velocity structure function (Eq. 4.1) to gauge the extent of the inertial range. We see that an inertial range just about begins to develop at $R_\lambda \sim 140$, which is consistent with the conclusions of Yeung & Zhou (1997). With

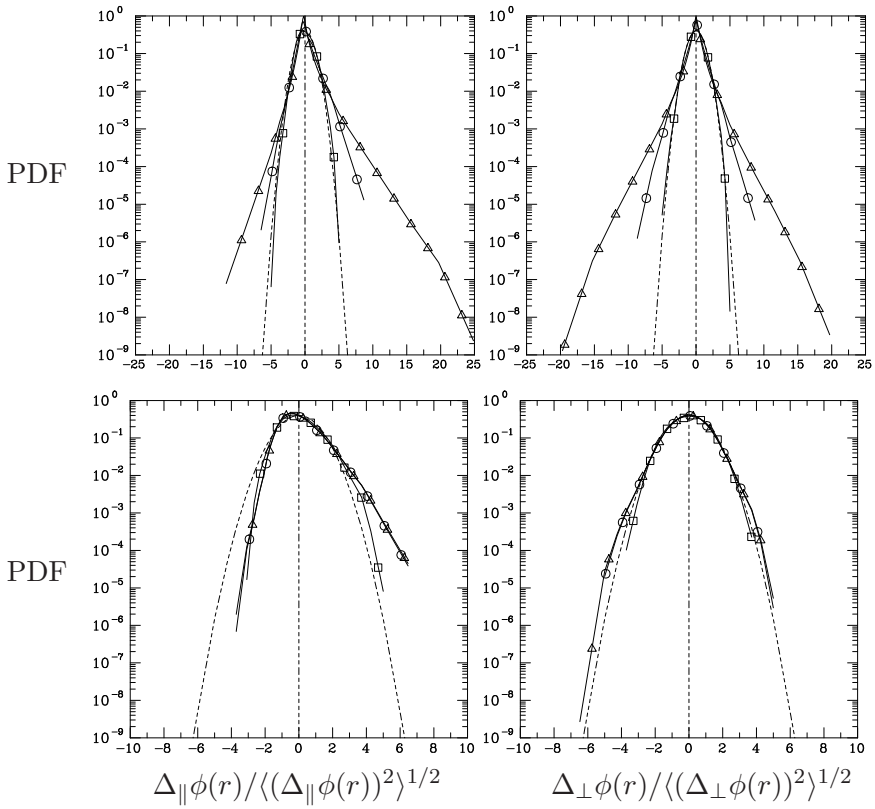

Figure 4.4: Standardized PDFs of scalar increments $\Delta\phi(r)$ with distance r taken over parallel (left) and perpendicular (right) to the mean gradient, for scalars with $Sc = 1/8$ (top) and $1/512$ (bottom) in $R_\lambda \sim 140$, 512^3 simulation on a $(4\pi)^3$ domain. Triangles, circles and squares correspond to $r/\Delta x = 2, 16, 128$ respectively. Dashed parabolic curves show a standard Gaussian distribution for comparison. (Note the difference in the range of x-axis between the top and bottom frames.)

decreasing Schmidt number at a given Reynolds number, we see a decreasing tendency towards a plateau in the inertial-convective range. due to the decreasing extent of the inertial-convective range. The decrease in the extent of the inertial-convective range with decreasing Schmidt number at a fixed Reynolds number is due to the fact that η_{OC} is growing faster than the scalar integral scale L_ϕ. Alternatively, as the Peclet number ($Pe \equiv R_\lambda Sc$) drops, η_{OC} grows and can even become comparable to the integral length scales in the velocity (L_1) and scalar fields (L_ϕ). In such a scenario we do not have a well defined inertial convective range. Also from Fig. 4.3 we can see that as the Reynolds number is increased at a fixed Schmidt number there is a tendency towards the inertial-convective plateau, suggesting that in the limit of infinite Peclet number, Eq. 4.4 may yet be satisfied.

For more information on the statistics of the scalar increments, we consider statistics of

one-dimensional ($1D$) increments of scalar fluctuations ($\Delta\phi(r)$) as a function of scale size r. Because of anisotropy induced by the mean gradient, the statistics of $\Delta\phi(r)$ are expected to differ between increments of r taken parallel or perpendicular to the mean gradient. Accordingly, we decompose $\Delta\phi(r)$ into components parallel and perpendicular to the mean gradient based on the direction of the separation distance r as

$$\Delta_{\parallel}\phi(r) = \phi(\mathbf{x} + r\mathbf{e}_{\parallel}) - \phi(\mathbf{x}) \ , \tag{4.6}$$

$$\Delta_{\perp}\phi(r) = \phi(\mathbf{x} + r\mathbf{e}_{\perp}) - \phi(\mathbf{x}) \ , \tag{4.7}$$

where, $\mathbf{e}_{\parallel}$ and $\mathbf{e}_{\perp}$ are unit vectors parallel and perpendicular to the mean scalar gradient respectively. The probability density function (PDF) of $\Delta_{\parallel}\phi(r)$ and $\Delta_{\perp}\phi(r)$ can be useful in studying intermittency and anisotropy as functions of scale size. In the small r limit, the PDF of scalar increments parallel and perpendicular to the mean gradient is expected to resemble that of the longitudinal and transverse scalar gradients respectively. Whereas, in the limit of large r, the PDF are expected to resemble that of the scalar fluctuation itself. Figure 4.4 shows the PDF of these increments, normalized by the root-mean-square (r.m.s) fluctuation in each case. For emphasis on low Schmidt number behavior we show data at $Sc = 1/8$ and $1/512$ at R_λ 140. For dependence on scale size we have chosen values of r that correspond closely to the Kolmogorov scale and the integral length scale as well as an intermediate r equal to the geometric mean of the two. The PDF at the smallest r in Fig. 4.4 are consistent with PDF of the corresponding scalar gradients reported in Yeung & Sreenivasan (2014). For small separation distances, it is clear from Fig. 4.4 that the PDF for the higher Schmidt number has longer tails than that for lower Schmidt number. This strongly indicates that the probability of intense fluctuations decreases with decrease in Schmidt number, owing to stronger diffusivity. The PDF of scalar increments perpendicular to the mean gradient has a vanishing skewness (for all separation distances) due to reflectional symmetry. However, for PDF of increments along the mean gradient, a non-zero skewness at small-r indicates that the assumption of local isotropy is weak. In Fig. 4.4, the PDF of $\Delta_{\parallel}\phi(r)$ at small-r, for both the Schmidt numbers are non-symmetric, indicating that the assumption of local isotropy is dubious at best. As the scale size increases the PDF of

$\Delta_\parallel \phi(r)$ tends to approach the Gaussian form for both scalars. These observations indicate that strong molecular diffusivity causes the scalar field to become less intermittent, with anisotropy persisting at the small scales. As the scale size increases, the fluctuations tend towards a normal distribution. The closeness between curves for different r at $Sc = 1/512$ also reflects the lack of scale separation when the Schmidt number is very low at a finite Reynolds number.

To quantify the shape of the PDF of scalar increments, it is useful to consider the normalized third and fourth order moments as a function of scale size r. We can define such moments both parallel and perpendicular to the mean gradient due to Eqs. 4.6 and4.7. The normalized third order structure function known as the "skewness structure function", is given as

$$\mu_3(r) \equiv \frac{\langle [\Delta_\parallel \phi(r)]^3 \rangle}{\langle [\Delta_\parallel \phi(r)]^2 \rangle^{3/2}} \,, \tag{4.8}$$

where r is in the direction of the mean gradient. The component of the skewness structure function perpendicular to the mean gradient is trivially zero due to reflection symmetry. The normalized fourth order structure functions known as the flatness structure functions can be defined for increments parallel and perpendicular to the mean gradient as

$$\mu_4^\parallel(r) \equiv \frac{\langle [\Delta_\parallel \phi(r)]^4 \rangle}{\langle [\Delta_\parallel \phi(r)]^2 \rangle^2} \,, \tag{4.9}$$

$$\mu_4^\perp(r) \equiv \frac{\langle [\Delta_\perp \phi(r)]^4 \rangle}{\langle [\Delta_\perp \phi(r)]^2 \rangle^2} \,. \tag{4.10}$$

In Fig. 4.5 we show the skewness structure function along the mean gradient and flatness structure functions both parallel and perpendicular to the mean gradient. The skewness structure function is non-negative for all scale sizes r. Moreover, contrary to local isotropy $\mu_3(r)$ becomes larger as r becomes smaller, for all Schmidt numbers. This is consistent with the asymmetric PDF of scalar increments parallel to the mean gradient at small r. At large r the skewness decreases towards zero as a result of homogeneity, while the flatness factors approach 3.0 since the large scales are approximately Gaussian. In the limit of $r \to 0$ the function $\mu_3(r)$ is expected to approach a constant equal to the (single-point) skewness of scalar gradient fluctuations in the direction of the mean gradient, while $\mu_4^\parallel(r)$ and $\mu_4^\perp(r)$ are expected to approach the longitudinal and transverse flatness factors of the

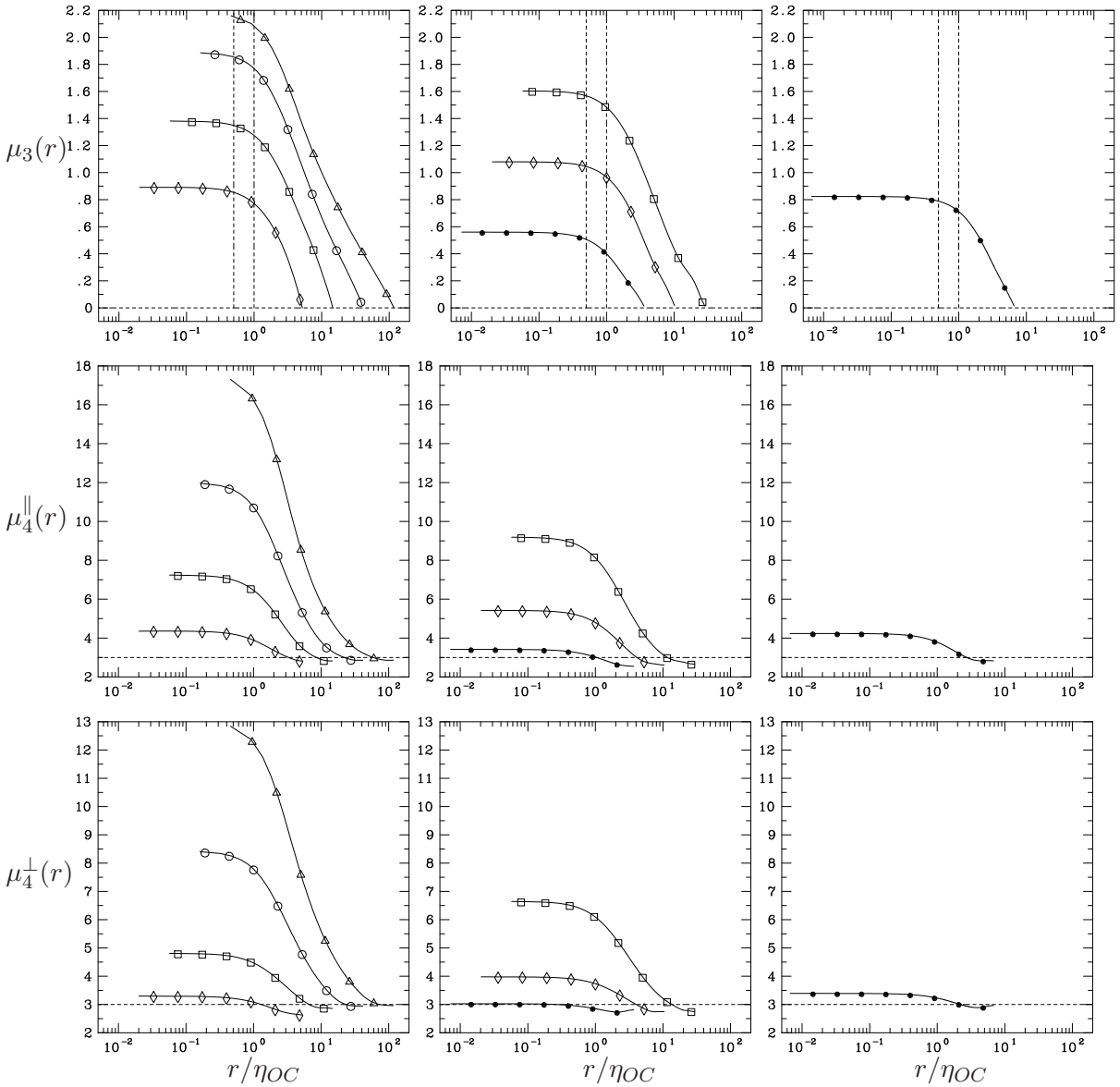

Figure 4.5: Normalized moments of two-point increments of scalars function versus Obukhov-Corrsin scaled separation at different Reynolds numbers and Schmidt numbers. Each choice of symbol on the curves denote a different value of Sc: 1/8 (△), 1/32 (◯), 1/128 (□), 1/512 (◇), and 1/2048 (•). The top, middle and bottom rows represent, respectively, the skewness structure function, $\mu_3(r)$ and the flatness structure functions $\mu_4^{\parallel}(r)$, $\mu_4^{\perp}(r)$, in directions parallel and perpendicular to the mean gradient. Frames in the left, middle and right columns are for $R_\lambda = 140$, 240 and 390 respectively. Vertical dashed lines are at $r/\eta_{OC} = 1/2$ and 1.

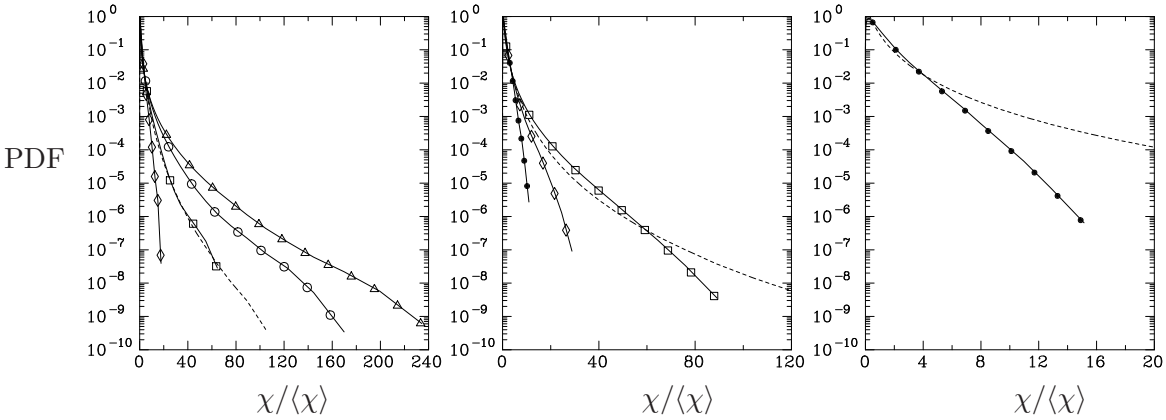

Figure 4.6: PDF of normalized scalar dissipation rate $\chi/\langle\chi\rangle$, using the same choice of symbols as in Fig. 4.5 for (a) (left) $R_\lambda \sim 140$, (b) (center) $R_\lambda \sim 240$ and (c) (right) $R_\lambda \sim 390$. Dashed lines show the PDF of normalized energy dissipation rate ($\epsilon/\langle\epsilon\rangle$) at each given Reynolds number. (Note that the ranges on the $\chi/\langle\chi\rangle$ axis are different for each frame in this figure.)

scalar gradients, respectively. It can be seen that data for r smaller than 1/2 of η_{OC} are nearly constant. This suggests a spacing close to 1/2 of η_{OC} may be sufficiently small for the scalar gradients to be reasonably well-resolved. The asymptotic values at small r are in very good agreement with the scalar gradient statistics reported in Yeung & Sreenivasan (2014) (Table III therein). On the other hand, comparison between different frames of this figure shows that an increase in Reynolds number with Schmidt number held fixed causes a mild increase in the skewness values. Flatness structure functions in the middle and bottom rows of this figure also follow similar trends. As Schmidt number decreases all of the flatness values become smaller, while the parallel direction still shows values higher than the perpendicular direction, which is indicative of anisotropy at scale r.

While the statistics of scalar increments focus on intermediate scale sizes, statistics of scalar gradients are more indicative of small-scale properties of the scalar field. In particular, the scalar dissipation rate, defined as

$$\chi = 2D_\phi \frac{\partial\phi}{\partial x_i}\frac{\partial\phi}{\partial x_i} \,. \tag{4.11}$$

is an important parameter of the small scales, since it is proportional to the scalar gradient fluctuations. The first moment of the scalar dissipation rate, $\langle\chi\rangle = 2D_\phi\langle(\partial\phi/\partial x_i)^2\rangle$, is the mean dissipation rate of the scalar variance $\langle\phi^2\rangle$. Although the scalar dissipation rate is a small scale quantity, the mean scalar dissipation rate is controlled by the large scales through

the advection-diffusion equation. The ratio of the scalar variance to the scalar dissipation rate ($\langle\phi^2\rangle/\langle\chi\rangle$), yields a time-scale that can be used to quantify the mixing efficiency. Smaller this ratio, faster (or more efficient) is the mixing process. Table 4.2 reports the ratio $\langle\phi^2\rangle/\langle\chi\rangle$ for various Reynolds and Schmidt number combinations considered in this work. As the Schmidt number decreases at a given Reynolds number the mixing time scale decreases, indicating a faster mixing process.

Table 4.2: Mixing efficiency quantified as the time scale $\langle\phi^2\rangle/\langle\chi\rangle$

N	512	512	512	512	1024	1024	1024	2048
R_λ	140	140	140	140	240	240	240	390
Sc	1/8	1/32	1/128	1/512	1/128	1/512	1/2048	1/2048
$\langle\phi^2\rangle/\langle\chi\rangle$	0.98	0.82	0.59	0.33	0.71	0.44	0.22	0.51

A number of studies in the literature (Pumir 1994; Overholt & Pope 1996; Yeung *et al.* 2005) suggest that small-scale intermittency is stronger for passive scalars than for the velocity field. For example the flatness values at small r seen in data at $Sc = 1/8$ and $R_\lambda \sim 140$ in Fig. 4.5 are larger than those for the velocity gradients at the same Reynolds number (5.4 and 8.0 for longitudinal and transverse gradients respectively). It is also clear that at any Reynolds number, scalars of some sufficiently low Schmidt number will be less intermittent than the velocity. For both velocity and scalar fields the form of the PDF of the dissipation rates contain useful information on the likelihood of extreme events in local deformation and property gradients in the flow.

In Fig. 4.6 we compare the PDFs of energy and scalar dissipation rates, normalized by their mean values, at the same Reynolds and Schmidt number combinations as in the preceding figures. As reported previously else where (Donzis *et al.* 2008*b*), the PDF of energy dissipation is well described by a stretched-exponential form, with tails that become wider at higher Reynolds number. It has also been reported that (Yeung *et al.* 2005), for $Sc = O(1)$ the PDF of scalar dissipation has even wider tails than that of energy dissipation. In Fig. 4.6(a) it can be seen that, at $R_\lambda \sim 140$, the the PDF of χ at $Sc = 1/8$ also has a well-defined stretched-exponential shape, and even at $Sc = 1/32$ the PDF of χ has wider tails than the PDF of ϵ. However, as the Schmidt number is reduced towards very low

values the tails become narrower, and the PDF resembles more nearly a simple exponential. In part (c) of the figure the PDF of χ at $Sc = 1/2048$ is much narrower than that of ϵ at the same Reynolds number.

Table 4.3: Moments up to fourth order for scalar dissipation and its logarithm.

N	512	512	512	512	1024	1024	1024	2048
R_λ	140	140	140	140	240	240	240	390
Sc	1/8	1/32	1/128	1/512	1/128	1/512	1/2048	1/2048
$\langle\chi\rangle$	2.98	2.75	2.11	1.03	2.06	1.55	0.77	1.20
$\langle\chi^3\rangle/\langle\chi\rangle^3$	211.9	86.7	22.3	6.4	39.8	10.0	4.9	6.6
$\langle\chi^4\rangle/\langle\chi\rangle^4$	10683	2797	302	31	686	64	17	30
$\mathrm{Var}(\ln\chi)$	2.92	2.09	1.45	1.06	1.80	1.25	0.99	1.15
$\mu_3(\ln\chi)$	0.01	-0.06	-0.32	-0.67	-0.18	-0.48	-0.77	-0.65
$\mu_4(\ln\chi)$	2.86	3.10	3.53	4.15	3.25	3.70	4.33	3.98

To provide more information on the statistical properties of scalar dissipation rate we show a number of statistical moments in Tab. 4.3 of χ and its logarithm. The mean scalar dissipation rate ($\langle\chi\rangle$) decreases with Schmidt number (at fixed R_λ) due to substantial reduction in the range of scales due to increasing molecular diffusivity. This is a departure from the the classical concept of dissipative anomaly (Donzis *et al.* 2005) in the low Schmidt number limit, and is consistent with previous work on this topic (Yeung & Sreenivasan 2014). The normalized third and fourth moments of χ, as well as the variance of $\ln\chi$, are all measures of intermittency, dominated by large values of χ. It can seen that all of these quantities become smaller when Sc is reduced to very low values at a fixed Reynolds number. Conversely as the Reynolds number is increased for a given Schmidt number, the normalized moments of χ increase, due to a wider range of scales. These observations are consistent with those from Fig. 4.6 and can be summarized by stating that intermittency decreases with an decrease in the Peclet number. Table 4.3 also shows that the skewness and flatness factors of $\ln\chi$ appear to move away from Gaussian values, thus showing that the log-normal hypothesis is less accurate at very low Sc at a finite Reynolds number. With increase in Reynolds number at a fixed Schmidt number, the skewness and flatness factors are closer to the Gaussian values, although the variance appears to deviate further away from unity.

The statistics of χ as presented above strongly suggest that any mechanism that tends

to generate extreme fluctuations of χ are suppressed when the Schmidt number is much lower than unity at least at moderate Reynolds numbers. In the section below we address this issue in greater detail.

4.3.2 Amplification by alignment with principal strain rates

In order to investigate the mechanisms that control extreme scalar gradient fluctuations, we investigate the dynamics of the instantaneous scalar dissipation rate fluctuations by analyzing its evolution equation (Vedula *et al.* 2001)

$$\frac{D\chi}{Dt} = D_\phi \frac{\partial^2 \chi}{\partial x_i \partial x_i} - 4D_\phi \frac{\partial \phi}{\partial x_j}\frac{\partial u_i}{\partial x_j}\frac{\partial \Phi}{\partial x_i} - 4D_\phi \frac{\partial \phi}{\partial x_i}\frac{\partial \phi}{\partial x_j} s_{ij} - 4D_\phi^2 \left(\frac{\partial^2 \phi}{\partial x_i \partial x_j}\right)^2 , \quad (4.12)$$

where D/Dt denotes the material derivative, and the four terms on the right-hand side represent, respectively: (i) transport by molecular diffusion, (ii) production by mean scalar gradient, (iii) nonlinear amplification or stretching by strain rate fluctuations (s_{ij}), and (iv) destruction of dissipation fluctuations by molecular diffusivity. The transport term has zero spatial average due to homogeneity, in which case the mean dissipation $\langle\chi\rangle$ is given by

$$\frac{d\langle\chi\rangle}{dt} = -4D_\phi \left\langle \frac{\partial \phi}{\partial x_j}\frac{\partial u_i}{\partial x_j} \right\rangle \frac{\partial \Phi}{\partial x_i} - 4D_\phi \left\langle \frac{\partial \phi}{\partial x_i}\frac{\partial \phi}{\partial x_j} s_{ij} \right\rangle - 4D_\phi^2 \left\langle \left(\frac{\partial^2 \phi}{\partial x_i \partial x_j}\right)^2 \right\rangle . \quad (4.13)$$

The unsteady term on the LHS vanishes in statistically stationary turbulence. In practice, the unsteady term can be taken as zero when ensemble averaging is performed over a sufficiently large number of realizations covering a sufficiently long time span. The classical scenario is that the nonlinear amplification and molecular destruction terms dominate, especially when the Reynolds number is large. The mechanism of nonlinear amplification is itself the result of preferential alignment between scalar gradient and the most compressive principal strain rate Ashurst *et al.* (1987), although the degree of alignment depends only weakly on the Reynolds number Vedula *et al.* (2001).

Table 4.4 shows the individual terms in Eq. 4.13. The mean of the nonlinear amplification (which is positive) decreases in magnitude with decreasing Schmidt number at a fixed Reynolds number, while the mean gradient and the dissipation terms balance each other out. The increasingly dominant role of the mean gradient term with decreasing Schmidt number at a given Reynolds number, points to increasing non-universality in the scalar field

Table 4.4: Mean gradient, nonlinear amplification and molecular destruction terms (unnormalized) in equation for the mean scalar dissipation rate (see Eq. 4.13). Average of the transport term is zero due to homogeneity.

N	512	512	512	512	1024	1024	1024	2048
R_λ	140	140	140	140	240	240	240	390
Sc	1/8	1/32	1/128	1/512	1/128	1/512	1/2048	1/2048
Mean Gradient	1.86	4.36	7.32	9.16	5.29	7.88	9.29	7.29
Amplification	41.7	30.1	13.1	2.78	25.9	8.52	1.45	3.72
Destruction	-43.6	34.4	-20.5	-12.0	-31.1	-16.4	-10.8	-10.9

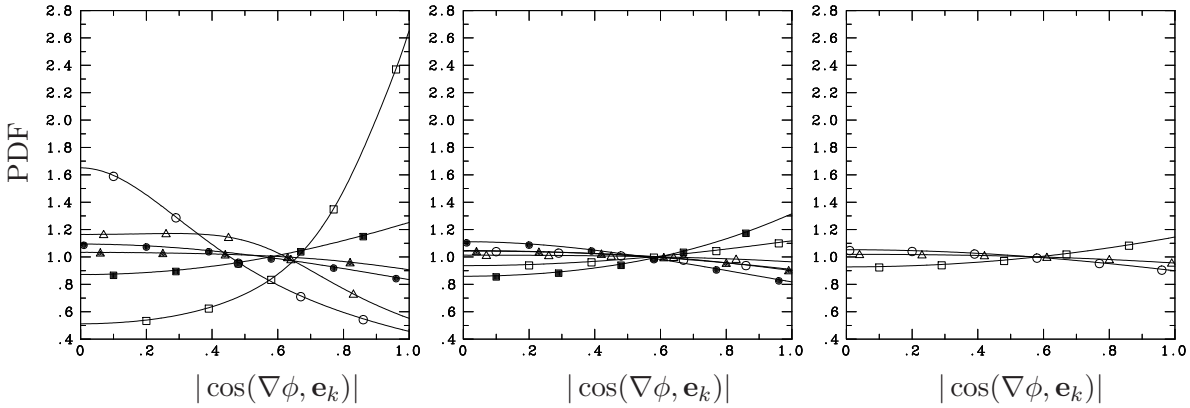

Figure 4.7: PDFs of direction cosines between fluctuating scalar gradient and principal strain axes. Triangles, circles and squares represent alignment with the axes $\mathbf{e}_\alpha$, $\mathbf{e}_\beta$, $\mathbf{e}_\gamma$ respectively. In (a) open and closed symbols are for $Sc = 1/8$ and $1/512$ respectively. In (b) closed and open symbols are for $Sc = 1/512$ and $Sc = 1/2048$. In (c) open symbols are for $Sc = 1/2048$.

dynamics. Due to the large diffusivity, the small scales in the scalar field are large enough to be directly effected by the large scales. In contrast, as the Reynolds number increases at a fixed Schmidt number, the reverse is true and the mean amplification term increases in magnitude while the mean gradient term becomes smaller. Furthermore, since the mean of the nonlinear amplification is directly related to the scalar transfer spectrum, we conclude that the spectral transfer is suppressed by the strong molecular diffusivity in the case of $Sc \ll 1$, at a fixed Reynolds number. This is consistent with the findings presented in Yeung & Sreenivasan (2014).

The nonlinear amplification term in Eq. 4.12 is of interest because it contains the coupling between velocity and scalar fields whereby scalar gradients are amplified by association with the local straining of fluid elements. The magnitude as well as the sign, of this term depends on the geometric alignment between scalar gradients and strain rates. Because of

Table 4.5: Statistical measures of the degree of alignment between fluctuating scalar gradients and principal strain rates. The symbols G_α,G_β and G_γ denote the normalized mean square gradients projected along the eigenvectors $\mathbf{e}_\alpha$,$\mathbf{e}_\beta$ and $\mathbf{e}_\gamma$.

Grid	512^3	512^3	512^3	512^3	1024^3	1024^3	1024^3	2048^3
R_λ	140	140	140	140	240	240	240	390
Sc	1/8	1/32	1/128	1/512	1/128	1/512	1/2048	1/2048
G_α	0.1983	0.2236	0.2702	0.3090	0.2623	0.3030	0.3238	0.3210
G_β	0.1303	0.1685	0.2338	0.2847	0.2236	0.2778	0.3099	0.3060
G_γ	0.6714	0.6079	0.4960	0.4064	0.5140	0.4192	0.3662	0.3730

the tensorial structure involved, this alignment is best studied in the coordinate frame defined by the principal axes of the strain-rate tensor. In this coordinate frame the strain-rate tensor is purely diagonal, with diagonal elements denoted by $\alpha \geq \beta \geq \gamma$, subject to the constraint $\alpha + \beta + \gamma = 0$ due to incompressibility. Previous work involving scalars with $Sc = 0(1)$ or higher Ashurst *et al.* (1987); Vedula *et al.* (2001) show that the scalar gradient fluctuations are likely to be aligned most strongly with the eigenvector corresponding to the most compressive strain rate and least so with that corresponding to the intermediate strain rate. An assessment of the alignment between scalar gradients ($\nabla\phi$) and principal strain axes is made in Fig. 4.7, which shows the PDFs of the direction cosines of the angles between $\nabla\phi$ and unit vectors $\mathbf{e}_\alpha$, $\mathbf{e}_\beta$ and $\mathbf{e}_\gamma$ along the principal axes corresponding to each principal strain rate. Preferential alignment with the most compressive strain rate is seen in the PDF of the alignment angle between $\nabla\phi$ and $\mathbf{e}_\gamma$ showing a strong peak near the value 1.0. It is clear that this effect becomes much weaker as the Schmidt number is reduced. The lack of significant alignment at $Sc \ll 1$ is indicated by all three curves for the R_λ and Sc combination in part (c) of this figure becoming nearly flat.

The preferential alignment noted above can be further characterized by comparing the mean squares of scalar gradient fluctuations projected along each of the principal strain eigen vectors. For example, for each of $k = \alpha, \beta, \gamma$, we can define

$$G_k \equiv \frac{\langle (\nabla\phi \cdot \mathbf{e}_k)^2 \rangle}{\langle |\nabla\phi|^2 \rangle} \ , \tag{4.14}$$

such that $G_\alpha + G_\beta + G_\gamma = 1$. Table 4.5 reports G_k $(k = \alpha, \beta, \gamma)$ for a range of Reynolds and Schmidt numbers considered. As the Schmidt number decreases at a fixed Reynolds

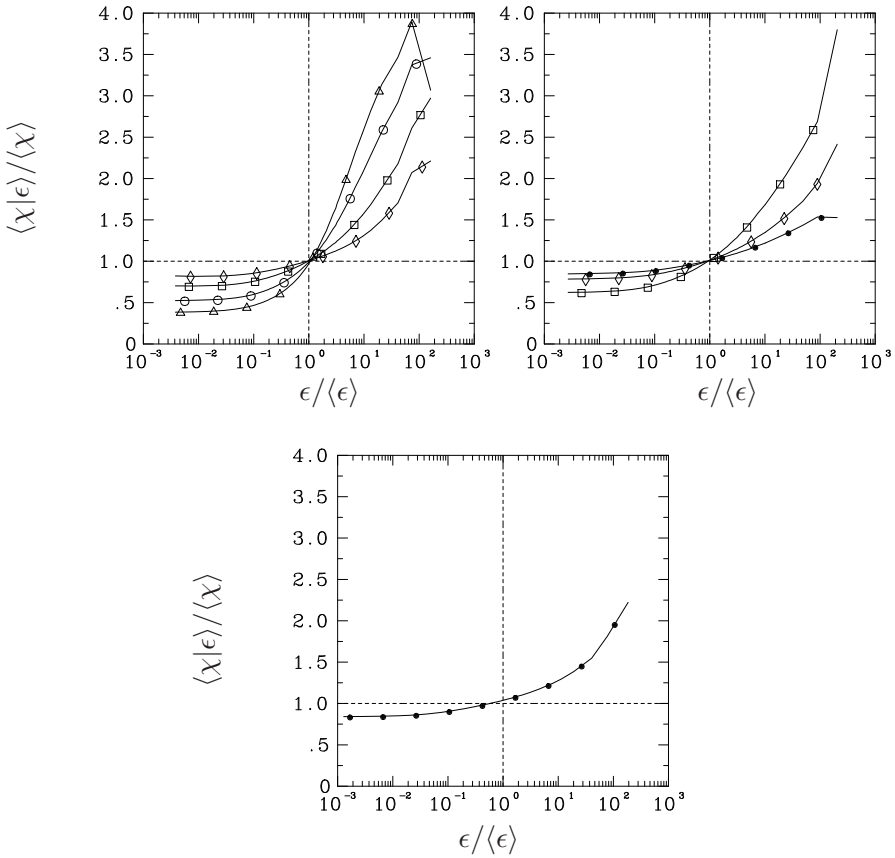

Figure 4.8: Conditional expectation of scalar dissipation given the energy dissipation. (a) (top left) $R_\lambda \sim 140$, (b) (top right) $R_\lambda \sim 240$, (c) (bottom) $R_\lambda \sim 390$. Symbols in each frame correspond to same as that in Fig. 4.5.

number we see that G_α, G_β and G_γ are closer to each other, indicating that the scalar gradient vector is equally likely to be aligned with any of the principal strain rate vectors in the low Schmidt number limit. The data in Tab. 4.5 appear insensitive to the increase in Reynolds numbers considered in the present work.

The reduced degree of alignment seen above also suggests that the likelihood of large scalar gradient and large strain rate occurring together is correspondingly reduced. In other words, the connection between local fluctuations of energy dissipation and scalar dissipation rates is also weakened in the low Sc limit. This is demonstrated in Fig. 4.8, which shows the conditional expectation of scalar dissipation given the energy dissipation. As expected there is some uncertainty in the results at large ϵ since the number of samples becomes limited. However it is clear that as the Schmidt number is reduced, the conditional mean becomes uniformly closer to the unconditional mean at all values of $\epsilon/\langle\epsilon\rangle$. In the lowest

Schmidt number case in all three simulations in this figure it is seen that, although, on average, a large ϵ still tends to lead to a larger χ, even an instantaneous ϵ at 100 times the mean produces a conditionally averaged χ merely of order $2\langle\chi\rangle$.

4.4 Two-point scalar structure function equation

In the previous sections the similarity scaling properties derived from the spatial structure of scalar fluctuations in the low Schmidt number limit have been shown to differ strongly from the classical picture in turbulent mixing. In this section, we focus our attention on the properties of mixed moments of the velocity and scalar increments in space. We begin by deriving the second order two-point scalar equation which relates the scalar fluctuations between two points in space. We provide theoretical limits for the various terms involved in this equation, following which we derive the Yaglom's relation. Finally we show results for the Yaglom's relation and the structure function budget and give asymptotic relations that are expected to hold for decreasing Schmidt number scalars.

The second order scalar structure function equation governs the the second moment of the differences in scalar fluctuations between two points in space. The two-point equation focuses on the scale separation, thus providing insights on scale similarity in the scalar field. In this section we derive the second order structure function equation from the advection-diffusion equation. We provide analytical expressions for the terms involved in forms that are amenable to numerical computations. As consistency checks, we also give the theoretical limits for the various terms in the structure function budget for the extreme cases of very small and very large separation distances.

Consider a passive scalar $\Phi(\mathbf{x},t)$ with mean $\langle\Phi(\mathbf{x},t)\rangle$ and fluctuation $\phi(\mathbf{x},t)$ such that $\langle\phi(\mathbf{x},t)\rangle = 0$ (here $\langle\cdot\rangle$ denotes a space average). The equation for the scalar fluctuation $\phi(\mathbf{x},t)$ advected by the fluctuating velocity component $u_i(\mathbf{x},t)$ (for $i = 1,2,3$) is

$$\partial_t\phi(\mathbf{x},t) + \partial_{x_m}[u_m(\mathbf{x},t)\phi(\mathbf{x},t)] = -u_m(\mathbf{x},t)\partial_{x_m}\langle\Phi(\mathbf{x},t)\rangle + D_\phi\partial_{x_m}\partial_{x_m}\phi(\mathbf{x},t)\ , \qquad (4.15)$$

where ∂ denotes partial differentiation with respect to its subscript variable and D_ϕ is the diffusivity. Summation is implied by repeated indices, for example, $\partial_{x_m}\partial_{x_m}$ is the Laplacian operator. The velocity field is incompressible so that the velocity fluctuation is divergence

free, i.e. $\partial_{x_m} u_m = 0$. The scalar variance evolution equation in the presence of a uniform mean gradient G in the x_1 direction, can be derived by multiplying Eq. 4.15 by 2ϕ, averaging and invoking homogeneity and incompressibility to give,

$$\partial_t \langle \phi^2 \rangle = -2\langle u_1 \phi \rangle G - \langle \chi \rangle \ , \tag{4.16}$$

where terms on the right hand side represent production and dissipation of the scalar variance respectively.

Equation 4.15 applies at every point $\mathbf{x}$ and in particular at a point $\mathbf{x}'$, which can be varied independently of $\mathbf{x}$. For brevity, let $\phi = \phi(\mathbf{x}, t)$, $\phi' = \phi(\mathbf{x}', t)$, $u_i = u_i(\mathbf{x}, t)$ and $u_i' = u_i(\mathbf{x}', t)$. The evolution equation $\partial_t(\phi - \phi')^2 = 2\phi\partial_t\phi - 2\phi\partial_t\phi - 2\phi\partial_t\phi' + 2\phi'\partial_t\phi'$ can be obtained by multiplying Eq. 4.15 by 2ϕ and by $2\phi'$ and the analogous operations on Eq. 4.15 written at $\mathbf{x}'$, then adding and subtracting the resultant equations to obtain

$$\begin{aligned} \partial_t(\phi - \phi')^2 + \partial_{x_m}[u_m(\phi^2 - 2\phi\phi')] + \partial_{x'_m}(u'_m(\phi'^2 - 2\phi\phi')] &= -2[(\phi - \phi')u_m]\partial_{x_m}\langle\Phi\rangle \\ -2[(\phi' - \phi)u'_m]\partial_{x'_m}\langle\Phi'\rangle D_\phi[\partial_{x_m}\partial_{x_m}(\phi^2 - 2\phi\phi') &+ \partial_{x'_m}\partial_{x'_m}(\phi'^2 - 2\phi\phi') \\ -2(\partial_{x_m}\phi)(\partial_{x_m}\phi) &- 2(\partial_{x'_m}\phi')(\partial_{x'_m}\phi')] \ , \end{aligned} \tag{4.17}$$

where the identity $f\partial_{x_m}\partial_{x_m}g + g\partial_{x_m}\partial_{x_m}f = \partial_{x_m}\partial_{x_m}(fg) - 2(\partial_{x_m}f)(\partial_{x_m}g)$ was used on the right-hand side. We use a constant mean scalar gradient $\nabla\langle\Phi(\mathbf{x}, t)\rangle = (G, 0, 0)$, which implies $\partial_{x_m}\langle\Phi(\mathbf{x}, t)\rangle = \partial_{x'_m}\langle\Phi(\mathbf{x}, t)\rangle' = G\delta_{1m}$, where δ_{ij} is the Kronecker-Delta function. Since $\mathbf{x}$ and $\mathbf{x}'$ are independent variables, we have $\partial_{x_i}\phi' = 0$, $\partial_{x'_i}\phi = 0$, which allows Eq. 4.17 to be written as

$$\begin{aligned} \partial_t\Delta\phi^2 + \partial_{x_m}(u_m\Delta\phi^2) + \partial_{x'_m}(u'_m\Delta\phi^2) &= \\ -2[\Delta\phi\Delta u_m]G\delta_{m1} + D_\phi(\partial_{x_m}\partial_{x_m}\Delta\phi^2 &+ \partial_{x'_m}\partial_{x'_m}\Delta\phi^2) - 2n \ , \end{aligned} \tag{4.18}$$

where the following notation has been used for convenience:

$$\begin{aligned} \Delta\phi &\equiv \phi - \phi' \ , \\ \Delta u_i &\equiv u_i - u'_i \ , \\ n &\equiv D_\phi[(\partial_{x_m}\phi)(\partial_{x_m}\phi) + (\partial_{x'_m}\phi')(\partial_{x'_m}\phi')] \ . \end{aligned} \tag{4.19}$$

The motivation for the definition of n is that, the average of n is the dissipation rate of the average of $(\phi^2 + \phi'^2)/2$, i.e. the dissipation rate of the scalar variance ($\langle\phi^2\rangle$).

Now we change the independent variables from $\mathbf{x}$ and $\mathbf{x}'$ to their sum and difference, using

$$\mathbf{X} \equiv (\mathbf{x} + \mathbf{x}')/2 \ , \quad \mathbf{r} \equiv \mathbf{x} - \mathbf{x}' \quad \text{and} \quad r = |\mathbf{r}| \ . \tag{4.20}$$

Since $\mathbf{x}$ and $\mathbf{x}'$ are independent, ∂_{x_i} and $\partial_{x'_i}$ are related to ∂_{r_i} and ∂_{X_i} by

$$\begin{aligned} \partial_{x_i} &= \partial_{r_i} + \frac{1}{2}\partial_{X_i} \ , \qquad \partial_{x'_i} = -\partial_{r_i} + \frac{1}{2}\partial_{X_i} \ , \\ \partial_{X_i} &= \partial_{x_i} + \partial_{x'_i} \ , \qquad \partial_{r_i} = \frac{1}{2}(\partial_{x_i} - \partial_{x'_i}) \ . \end{aligned} \tag{4.21}$$

The use of equations 4.21 in 4.18 gives

$$\begin{aligned} \partial_t \Delta\phi^2 + \partial_{X_m}[(u_m + u'_m)\Delta\phi^2/2] + \quad & \partial_{r_m}[\Delta u_m \Delta\phi^2] \quad = \\ -2[\Delta\phi\Delta u_m]G\delta_{m1} + \quad & 2D_\phi(\partial_{r_m}\partial_{r_m}\Delta\phi^2 \quad + \frac{1}{4}\partial_{X_m}\partial_{X_m}\Delta\phi^2) - 2n \ . \end{aligned} \tag{4.22}$$

We are interested in the statistics of the fluctuations, hence we average (in space) the individual terms. Homogeneity implies that the spatial derivatives $\partial_{X_m}\langle\cdot\rangle$ vanish, and that $\langle n\rangle = \langle\chi\rangle$. Since $\mathbf{x}$ is statistically immaterial due to homogeneity, we denote the two-point differences of the scalar and velocity component fluctuations as $\Delta\phi(\mathbf{r}, t) = \phi(\mathbf{x}, t) - \phi(\mathbf{x}', t)$ and $\Delta u_i(\mathbf{r}, t) = u_i(\mathbf{x}, t) - u_i(\mathbf{x}', t)$, to emphasize the dependence on the spatial separation vector $\mathbf{r} = \mathbf{x} - \mathbf{x}'$. We use the fact that the mean gradient is non-zero only in the x_1 direction, and average the above equation, to get the final form of the equation, which is given as

$$\frac{\partial}{\partial t}\langle\Delta\phi(\mathbf{r})^2\rangle + \frac{\partial}{\partial r_i}\langle\Delta u_i(\mathbf{r})\Delta\phi(\mathbf{r})^2\rangle = -2\langle\Delta u_1(\mathbf{r})\Delta\phi(\mathbf{r})\rangle G + 2D_\phi\frac{\partial^2\langle\Delta\phi(\mathbf{r})^2\rangle}{\partial r_i \partial r_i} - 2\langle\chi\rangle \ , \tag{4.23}$$

where for simplicity we have suppressed time dependence in the notation. The terms on left hand side represent unsteadiness (if nontrivial) and convection, while terms on the right hand side represent production by mean gradient, molecular diffusion and dissipation respectively. The term $\langle\Delta u_i(\mathbf{r})\Delta\phi(\mathbf{r})^2\rangle$, whose divergence in r-space is the convection term in Eq. 4.23 is called the mixed velocity-scalar structure function. In the following, we describe the large and small separation limits for the individual terms in Eq. 4.23.

4.5 Scalar structure function budget

We derive expressions to asses the various terms involved in the evolution of the second order scalar structure function $\langle \Delta\phi(\mathbf{r})^2 \rangle$ (Eq. 4.23). The final form of the analytical expressions is partly guided by computational considerations. The limiting behavior of large and small separation distances for each of the terms involved in the budget equation is discussed. Such an analysis is not only useful for a theoretical understanding of the various phenomena involved in the scalar structure function budget, but also serves as a check for the computations performed.

4.5.1 Statistical stationarity

If the scalar field is statistically stationarity, the unsteady term in Eq. 4.23 can be considered negligible, that is

$$\frac{\partial}{\partial t}\langle \Delta\phi(\mathbf{r})^2 \rangle = 0 \, . \tag{4.24}$$

From Eq. 4.16 it follows that $-2\langle u_1\phi \rangle G = \langle \chi \rangle$ if the scalar fluctuations reach a statistically steady state. The departure of $2\langle u_1\phi \rangle G/\langle \chi \rangle$ from unity can be used to quantify the extent to which a statistically stationary state is achieved. In the event that the scalar field has not achieved the desired level of stationarity, it is still possible to test the data at small and large separation distances using analytic expressions. Using homogeneity, the unsteady term can be written as

$$\frac{\partial \langle (\Delta\phi(\mathbf{r}))^2 \rangle}{\partial t} = 2\left[\frac{\partial \langle \phi^2 \rangle}{\partial t} - \frac{\partial \langle \phi(\mathbf{x})\phi(\mathbf{x}+\mathbf{r}) \rangle}{\partial t} \right] \, . \tag{4.25}$$

We are interested in the limiting cases of small separation distances ($r \to 0$) and large spatial separation ($r \to \infty$) of the above equation. The time derivative for the first term on the right hand side of above equation can be substituted directly from Eq. 4.16, while the limit and the derivative for the second term can be interchanged to evaluate the limiting behavior. Noting that, $\langle \phi(\mathbf{x})\phi(\mathbf{x}+\mathbf{r}) \rangle \to \langle \phi^2 \rangle$ as $r \to 0$ and $\langle \phi(\mathbf{x})\phi(\mathbf{x}+\mathbf{r}) \rangle \to \langle \phi(\mathbf{x}) \rangle \langle \phi(\mathbf{x}+\mathbf{r}) \rangle = 0$ as $r \to \infty$, in in the small-r limit we have

$$\lim_{r \to 0} \frac{\partial}{\partial t}\langle \Delta\phi(\mathbf{r})^2 \rangle = 0 \, . \tag{4.26}$$

Whereas, in the large-r limit we have from Eq. 4.16,

$$\lim_{r\to\infty}\frac{\partial}{\partial t}\langle\Delta\phi(\mathbf{r})^2\rangle = 2\frac{\partial\langle\phi^2\rangle}{\partial t} - \frac{\partial}{\partial t}[\lim_{r\to\infty}\langle\phi(\mathbf{x})\phi(\mathbf{x}+\mathbf{r})\rangle] = -4\langle u_1\phi\rangle G - 2\langle\chi\rangle\ . \tag{4.27}$$

4.5.2 Production by mean gradient

The mean gradient contribution to the evolution of the second order scalar structure is given by $P(\mathbf{r}) = -2G[2\langle\phi(\mathbf{x})u_1(\mathbf{x})\rangle - \langle\phi(\mathbf{x}+\mathbf{r})u_1(\mathbf{x})\rangle - \langle\phi(\mathbf{x})u_1(\mathbf{x}+\mathbf{r})\rangle]$. Since $\langle\phi(\mathbf{x}+\mathbf{r})u_1(\mathbf{x})\rangle \to \langle\phi(\mathbf{x})\rangle\langle u_1(\mathbf{x})\rangle = 0$ as $r\to\infty$, $P(\mathbf{r})$ is constant for large r and approaches monotonically to zero as r decreases. Using the velocity-scalar two-point correlation, $\rho_{u\phi}(\mathbf{r}) \equiv \langle u(\mathbf{x})\phi(\mathbf{x}+\mathbf{r})\rangle/(\sigma_u\sigma_\phi)$, the production term can be expressed as

$$P(\mathbf{r}) = -2G[2\langle u_1\phi\rangle - \sigma_u\sigma_\phi(\rho_{u\phi}(\mathbf{r}) + \rho_{u\phi}(-\mathbf{r}))]\ , \tag{4.28}$$

where σ_u and σ_ϕ denote root-mean-square fluctuations. In general, two-point cross-correlations are not even functions: i.e. $\rho_{u\phi}(\mathbf{r}) \neq \rho_{u\phi}(-\mathbf{r})$. However, ensemble-averaged results from multiple realizations suggest that, even in the direction of the mean gradient, the difference between $\rho_{u\phi}(r)$ and $\rho_{u\phi}(-r)$ is quite small. The mean gradient in the x direction also suggests some mild differences at intermediate r among $\rho_{u\phi}(r,0,0)$, $\rho_{u\phi}(0,r,0)$, and $\rho_{u\phi}(0,0,r)$ are expected, with the latter two closer to each other. Nevertheless, component averaging over results for $\mathbf{r} = (r,0,0)$, $(0,r,0)$ and $(0,0,r)$ is still applicable. Thus using the approximation $\rho_{u\phi}(\mathbf{r}) = \rho_{u\phi}(-\mathbf{r})$, the mean gradient contribution to $\langle\Delta_{\mathbf{r}}\phi)^2\rangle$ can be written as

$$P_1(\mathbf{r}) = -4G[\langle u_1\phi\rangle - \sigma_u\sigma_\phi\rho_{u\phi}(\mathbf{r})]\ . \tag{4.29}$$

As consistency checks, it is useful to derive analytical expressions for $P_1(\mathbf{r})$ in the small-r and large-r limits.

$$\lim_{r\to 0} P_1(\mathbf{r}) = 0\ , \tag{4.30}$$

$$\lim_{r\to\infty} P_1(\mathbf{r}) = -4G\langle u_1\phi\rangle\ . \tag{4.31}$$

Figure 4.9 shows the production calculated using Eqs. 4.28 and 4.29 for two different Schmidt numbers using long-time averages. The plots suggest that $P_1(\mathbf{r})$ (Eq. 4.29) can be used to evaluate the production term reliably. From Eqs. 4.16 and 4.31, the ratio $P_1(\mathbf{r})/2\langle\chi\rangle \to 1$

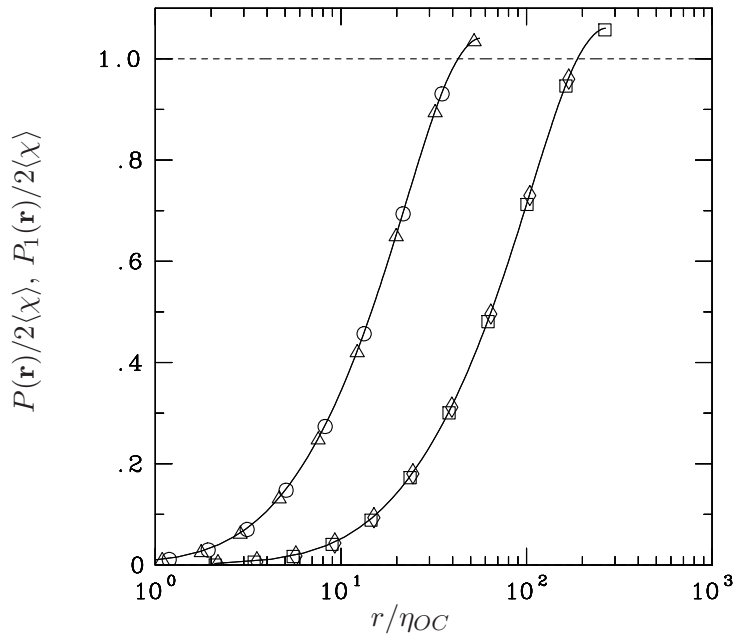

Figure 4.9: Production by mean scalar gradient in the scalar structure function equation for 256^3, $R_\lambda \sim 140$ simulation on a $(2\pi)^3$ domain using 21 snapshots, using two different expressions (Eqs. 4.28 and 4.29). Symbols ($\triangle$) and ($\bigcirc$) correspond to $P(\mathbf{r})$ and $P_1(\mathbf{r})$ for $Sc = 1/8$ while ($\square$) and ($\diamond$) are for $Sc = 1$. Dashed line at 1.0 shows large-r limit for a stationary state.

as $r \to \infty$ if the scalar field is statistically stationary. From Fig. 4.9 we see that the large-r limit for both scalars satisfy this limit approximately.

4.5.3 Molecular diffusion

In order to convert the molecular diffusion term (second term in right hand side of Eq. 4.23) into a form which is readily amenable to analytical and computational analysis, we use the following facts (i) ϕ depends on $\mathbf{x}$ only and ϕ' depends on $\mathbf{x}' = \mathbf{x}+\mathbf{r}$ only; and (ii) $\nabla_\mathbf{x} = -\nabla_\mathbf{r}$ whereas $\nabla_{\mathbf{x}'} = \nabla_\mathbf{r}$ (such that both $\nabla^2_\mathbf{x}$ and $\nabla^2_{\mathbf{x}'}$ are equivalent to the Laplacian operator $\nabla^2_\mathbf{r}$, i.e. $\partial^2/\partial r_i \partial r_i$ in r-space). We first consider the differentiation of $\langle \Delta_\mathbf{r}\phi^2\rangle$ with respect to $\mathbf{r}$, noting that only $\mathbf{x}'$ depends on $\mathbf{r}$,

$$\frac{\partial}{\partial r_i}\langle(\Delta\phi(\mathbf{r})^2\rangle = 2\langle\Delta\phi(\mathbf{r})\frac{\partial\phi}{\partial r_i}(\mathbf{x}')\rangle \ . \tag{4.32}$$

Differentiating again and making use of the fact that $\nabla_{\mathbf{r}} = \nabla_{\mathbf{x}'}$ and $\nabla^2_{\mathbf{r}} = \nabla^2_{\mathbf{x}}$ as noted above, we get

$$\frac{\partial^2 \langle (\Delta\phi(\mathbf{r}))^2 \rangle}{\partial r_i \partial r_i} = 2 \left\langle \frac{\partial \phi'}{\partial x_i'} \frac{\partial \phi'}{\partial x_i'} \right\rangle + 2 \left\langle \Delta\phi(\mathbf{r}) \frac{\partial^2 \phi}{\partial x_i^2}(\mathbf{x}') \right\rangle \ . \tag{4.33}$$

Finally, invoking homogeneity, summing over the choices $i = 1, 2, 3$ and multiplying by the factor $2D_\phi$ gives the result

$$2D_\phi \frac{\partial^2 \langle (\Delta\phi)^2 \rangle}{\partial r_i \partial r_i} = 2\langle \chi \rangle + 4D_\phi \langle \Delta\phi(\mathbf{r}) \nabla^2 \phi(\mathbf{x}') \rangle \ . \tag{4.34}$$

The above expression for the diffusion term can be readily computed using parallel codes based on a pencil decomposition. In order to check the behavior of the diffusion term for both small and large separation distances, we require the following analysis.

In the limit as $r \to 0$, $\Delta\phi(r) \to 0$, hence we get

$$\lim_{r \to 0} 2D_\phi \frac{\partial^2 \langle (\Delta\phi)^2 \rangle}{\partial r_i \partial r_i} = 2\langle \chi \rangle \ . \tag{4.35}$$

In order to get the large-r limit, we can use homogeneity and the product rule to show that

$$4D_\phi \langle \Delta\phi(\mathbf{r}) \nabla^2 \phi(\mathbf{x}') \rangle = -2\langle \chi \rangle - 4D_\phi \langle \phi(\mathbf{x}) \nabla^2 \phi(\mathbf{x}') \rangle \ . \tag{4.36}$$

Substituting the above result into Eq. 4.34 and taking the large-r limit, noting that $\langle \phi(\mathbf{x}) \nabla^2 \phi(\mathbf{x}') \rangle \to \langle \phi(\mathbf{x}) \rangle \langle \nabla^2 \phi(\mathbf{x}') \rangle \to 0$ as $r \to \infty$ we get

$$\lim_{r \to \infty} 2D_\phi \frac{\partial^2 \langle (\Delta\phi)^2 \rangle}{\partial r_i \partial r_i} = 0 \ . \tag{4.37}$$

4.5.4 Advective transport

The convective terms in Eq. 4.23 appear as the divergence of the vector $\mathbf{g}(\mathbf{r}) = \langle \Delta\mathbf{u}(\mathbf{r}) \Delta\phi(\mathbf{r})^2 \rangle$ and is expected to dominate with increasing Reynolds number. For the case when $\mathbf{r} = (r, 0, 0)$, the divergence can be written as

$$\nabla \cdot \mathbf{g}(\mathbf{r}) = \frac{\partial}{\partial r_1} g_1(\mathbf{r}) \bigg|_{\mathbf{r}=(r_1=r, r_2=0, r_3=0)} + \frac{\partial}{\partial r_2} g_1(\mathbf{r}) \bigg|_{\mathbf{r}=(r_1=r, r_2=0, r_3=0)} \frac{\partial}{\partial r_3} g_1(\mathbf{r}) \bigg|_{\mathbf{r}=(r_1=r, r_2=0, r_3=0)} \ . \tag{4.38}$$

The first term on the right hand side constitutes the longitudinal term while the other two correspond to the transverse contributions to the divergence. Care must be taken

to carry out the differentiation before substituting the value of $\mathbf{r} = (r, 0, 0)$ in the above expression, as shown below

$$\begin{aligned}
\nabla \cdot \mathbf{g}(\mathbf{r}) &= \frac{\partial}{\partial r_1}\langle \Delta u_1(r,0,0)\Delta\phi(r,0,0)^2\rangle \Big|_{\mathbf{r}=(r_1=r,r_2=0,r_3=0)} \\
&+ \frac{\partial}{\partial r_2}\langle \Delta u_2(r,r_2,0)\Delta\phi(r,r_2,0)^2\rangle \Big|_{\mathbf{r}=(r_1=r,r_2=0,r_3=0)} \\
&+ \frac{\partial}{\partial r_3}\langle \Delta u_3(r,0,r_3)\Delta\phi(r,0,r_3)^2\rangle \Big|_{\mathbf{r}=(r_1=r,r_2=0,r_3=0)} . \qquad (4.39)
\end{aligned}$$

The first term on the right hand side of Eq. 4.39 is the longitudinal contribution to the divergence term. Since the differentiation and the averaging operations commute, their order can be interchanged. Noting that $\Delta u_1(r,0,0) = u_1(x+r,y,z) - u_1(x,y,z)$ with $\partial u_1(x+r,y,z)/\partial r = \partial u_1(x+r,y,z)/\partial x$ and $\partial u_1(x,y,z)/\partial r = 0$ etc., we get

$$\begin{aligned}
&\frac{\partial}{\partial r_1}\langle \Delta u_1(r,0,0)\Delta\phi(r,0,0)^2\rangle \Big|_{r_1=r} = \\
&= \left\langle \frac{\partial u_1(x+r,y,z)}{\partial x}\Delta\phi(r,0,0)^2 \right\rangle + 2\left\langle \Delta u_1(r,0,0)\Delta\phi(r,0,0)\frac{\partial\phi(x+r,y,z)}{\partial x} \right\rangle \qquad (4.40)
\end{aligned}$$

The second term on the right hand side of Eq. 4.39 is evaluated as

$$\begin{aligned}
\frac{\partial}{\partial r_2}\langle \Delta u_2(r,r_2,0)\Delta\phi(r,r_2,0)^2\rangle \Big|_{r_2=0} &= \left\langle \left(\frac{\partial}{\partial r_2}\Delta u_2(r,r_2,0)\right)\Delta\phi(r,r_2,0)^2 \right\rangle \Big|_{r_2=0} \\
&+ \left\langle \Delta u_2(r,r_2,0)\left(\frac{\partial}{\partial r_2}\Delta\phi(r,r_2,0)^2\right) \right\rangle \Big|_{r_2=0} \qquad (4.41)
\end{aligned}$$

Noting that $\Delta u_2(r,r_2,0) = u_2(x+r,y+r_2,z) - u_2(x,y,z)$ and $\Delta\phi(r,r_2,0) = \phi(x+r,y+r_2,z) - \phi(x,y,z)$ and that $u_2(x,y,z)$, $\phi(x,y,z)$ are independent of r_2, we get

$$\begin{aligned}
\frac{\partial}{\partial r_2}\langle \Delta u_2(r,r_2,0)\Delta\phi(r,r_2,z)^2\rangle \Big|_{r_2=0} &= \left\langle \frac{\partial u_2(x+r,y+r_2,0)}{\partial r_2}\Big|_{r_2=0} \Delta\phi(r,0,0)^2 \right\rangle \\
&+ 2\left\langle \Delta u_2(r,0,0)\Delta\phi(r,0,0)\frac{\partial\phi(x+r,y+r_2,z)}{\partial r_2}\Big|_{r_2=0} \right\rangle \\
= \left\langle \frac{\partial u_2(x+r,y,z)}{\partial y}\Delta\phi(r,0,0)^2 \right\rangle &+ 2\left\langle \Delta u_2(r,0,0)\Delta\phi(r,0,0)\frac{\partial\phi(x+r,y,z)}{\partial y} \right\rangle . \qquad (4.42)
\end{aligned}$$

Similarly, the third term on the right hand side of Eq. 4.39 is

$$\frac{\partial}{\partial r_3}\langle \Delta u_3(r,0,r_3)\Delta\phi(r,0,r_3)^2\rangle\Big|_{r_3=0} = \left\langle \frac{\partial u_3(x+r,y,z)}{\partial z}\Delta\phi(r,0,0)^2 \right\rangle + 2\left\langle \Delta u_3(r,0,0)\Delta\phi(r,0,0)\frac{\partial\phi(x+r,y,z)}{\partial z} \right\rangle \quad (4.43)$$

Thus the expression in Eq. 4.40 is the longitudinal part of $\nabla\cdot\mathbf{g}(\mathbf{r})$ along $\mathbf{r}=(r,0,0)$, while those in Eqs. 4.42 and 4.43 constitute the transverse contributions. Similar expressions hold for $\mathbf{r}=(0,r,0)$ and $(0,0,r)$. The motivation behind decomposing the advective terms into longitudinal and transverse parts is that the integration of the longitudinal contribution to $\nabla\cdot\mathbf{g}(\mathbf{r})$ that potentially leads to the Yaglom relation (refer Eq. 4.2). The small-r limit of $\nabla\cdot\mathbf{g}(\mathbf{r})$ is trivially zero, while in the large-r limit, we have for instance,

$$\lim_{r\to\infty}\left\langle \frac{\partial u_1(x+r,y,z)}{\partial x}\Delta\phi(r,0,0)^2 \right\rangle \to \left\langle \frac{\langle\partial u_1(x+r,y,z)}{\partial x} \right\rangle \left\langle \Delta\phi(r,0,0)^2 \right\rangle = 0 \quad (4.44)$$

Similar argument holds for other contributions to the convective term which results in $\nabla\cdot\mathbf{g}(\mathbf{r})\to 0$ as $r\to\infty$.

We retain the constant dissipation term of $-2\langle\chi\rangle$ in Eq. 4.23 without any change, since it appears in the Yaglom relation (see Eq. 4.2). The unsteady term is negligible in our simulations, since the ensemble average is taken over sufficiently many instantaneous snapshots covering a sufficiently long time span, such that the statistics tend to reach a steady state. The convective term on the left hand side of Eq. 4.23 is computed using expressions such as Eqs. 4.40,4.42 and 4.43. We compute terms on the right hand side representing production, and molecular diffusivity according to the formulations given by 4.29 and 4.34.

4.6 Yaglom's relation

In this section we derive the Yaglom's relation from the two-point scalar equation (Eq. 4.23) for both isotropic and non-isotropic scalar fields. We present results for the Yaglom's relation for a range of Reynolds numbers and Schmidt numbers, with a focus on the low Schmidt number regime.

For a stationary scalar field, if there exists a range of scales where the diffusive and mean gradient effects can be neglected then from Eq. 4.23 we have

$$\frac{\partial}{\partial r_i}\langle \Delta u_i(\mathbf{r})\Delta\phi(\mathbf{r})^2\rangle = -2\langle\chi\rangle \ . \tag{4.45}$$

For weakly diffusive scalars ($Sc > 1$), neglect of the production and diffusive terms is appropriate in both the inertial-convective ($\eta \ll r \ll L_\phi$) and viscous convective ranges ($\eta_B \ll r \ll \eta$), if such ranges are well-formed. In the strongly diffusive case ($Sc < 1$), we can expect the production and diffusive terms to be negligible only in the inertial convective range ($\eta_{OC} \ll r \ll L$), if such a range indeed exists. If isotropy applies in these scale ranges, then the dependence on $\mathbf{r}$ reduces to the dependence on $r = |\mathbf{r}|$ only, and only the longitudinal velocity increment $\Delta u_L(r) = \Delta u_i(\mathbf{r})r_i/r$ is important. Consequently, if there exists a scale range where isotropy applies and such that the diffusive and production terms become negligible, then a straight forward integration of Eq. 4.45 with respect to r gives Yaglom's relation,

$$\langle \Delta u_L(r)\Delta\phi(r)^2\rangle = -2\langle\chi\rangle r/3 \ . \tag{4.46}$$

This is also known as the two-thirds law. Note that in Yaglom (1949), the definition $\langle\chi\rangle$ was a factor of 2 smaller, hence he obtained a factor of 4/3 in the right hand side of the above equation. If isotropy does not apply in the relevant range of scales where Eq. 4.45 holds, as is the case for a scalar field driven by a uniform mean gradient, then we have to integrate Eq. 4.45 in three-dimensional $\mathbf{r}$ space. Integrating Eq. 4.45 over a sphere of radius-r ($\mathcal{V}_r$) with volume V (Hill 2002; Gotoh & Yeung 2013), we get

$$\int_{\mathcal{V}_r} \frac{\partial}{\partial r_i}\langle \Delta u_i(\mathbf{r})\Delta\phi(\mathbf{r})^2\rangle dV = -2\langle\chi\rangle \int_{\mathcal{V}_r} dV \ . \tag{4.47}$$

We apply Gauss's theorem to convert the divergence term in Eq. 4.47 into an integral over the surface of a sphere ($\partial\mathcal{V}_r$) of radius r as

$$\int_{\partial\mathcal{V}_r} \langle \Delta u_i(\mathbf{r})\Delta\phi(\mathbf{r})^2\rangle \frac{r_i}{r} dS = -2\langle\chi\rangle \int_{\mathcal{V}_r} dV \ , \tag{4.48}$$

where $\mathbf{r}$/r is a unit vector normal to the surface of the sphere and dS represents an elemental area on the sphere with radius r. The term $\langle \Delta u_i(\mathbf{r})\Delta\phi(\mathbf{r})^2\rangle r_i/r$ is the longitudinal

component ($\langle \Delta u_L(r) \Delta\phi(r)^2 \rangle$) of the mixed structure function $\langle \Delta u_i(\mathbf{r}) \Delta\phi(\mathbf{r})^2 \rangle$. We define the spherical average of the longitudinal mixed structure function as

$$\langle \Delta u_L(r) \Delta\phi(r)^2 \rangle_{sp} = \frac{1}{4\pi r^2} \int_{\partial \mathcal{V}_r} \langle \Delta u_L(r) \Delta\phi(r)^2 \rangle dS \ . \tag{4.49}$$

Substituting Eq. 4.49 into Eq. 4.48 and noting that the integral on the right-hand-side is the volume of a sphere ($4\pi r^3/3$), we finally get

$$\langle \Delta u_L(r) (\Delta\phi(r))^2 \rangle_{sp} = -2\langle \chi \rangle r/3 \ , \tag{4.50}$$

Hence, in non-isotropic turbulence, Yaglom's relation is still valid with the proviso that the integral in Eq. 4.46 be replaced by a spherical average as given in Eq. 4.50. However, for the purpose of examining the linear scaling of the mixed velocity-scalar structure function with respect to spatial separation in Eq. 4.46, component averaging is still relevant. Figure 4.10 shows a direct test of Yaglom's relation, with a focus on more recent datasets at either higher Reynolds number or higher Schmidt number. It can be seen clearly that both an increase in Reynolds number and increase in Schmidt number produce improved agreement with Yaglom's relation, which is indicated by a plateau at height 2/3 at intermediate scale sizes.

Figure 4.11 shows a test of Yaglom's relation as computed from Eq. 4.46 for scalars with $Sc \leq 1$, which if valid, should lead to a plateau at height 2/3 in the inertial-convective range where the anisotropic large-scale effects and diffusion can be neglected. It is clear that as the Schmidt number is dropped to very low values, the data falls much lower than the level predicted by Yaglom's relation. This is perhaps not a surprise, since Yaglom's relation is in part based on the existence of a wide range of scales, which does not occur if the Schmidt number is very low, so low that even the Peclet number is small. In the limit of small r, using Taylor series approximation and assuming local isotropy holds for both the scalar and velocity fields, the mixed structure function can be written as (Yeung *et al.* 2002)

$$\frac{\langle \Delta_r u (\Delta_r \phi)^2 \rangle}{\langle \chi \rangle r} \approx \frac{S_{u\phi}}{6\sqrt{15}} \Big(\frac{r}{\eta_B} \Big)^2 \ , \tag{4.51}$$

where $S_{u\phi}$ is known as the mixed derivative skewness and is given as

$$S_{u\phi} = \left\langle \left(\frac{\partial u}{\partial x} \right) \left(\frac{\partial \phi}{\partial x} \right)^2 \right\rangle \Big/ \left\langle \left(\frac{\partial u}{\partial x} \right)^2 \right\rangle^{1/2} \left\langle \left(\frac{\partial \phi}{\partial x} \right)^2 \right\rangle \tag{4.52}$$

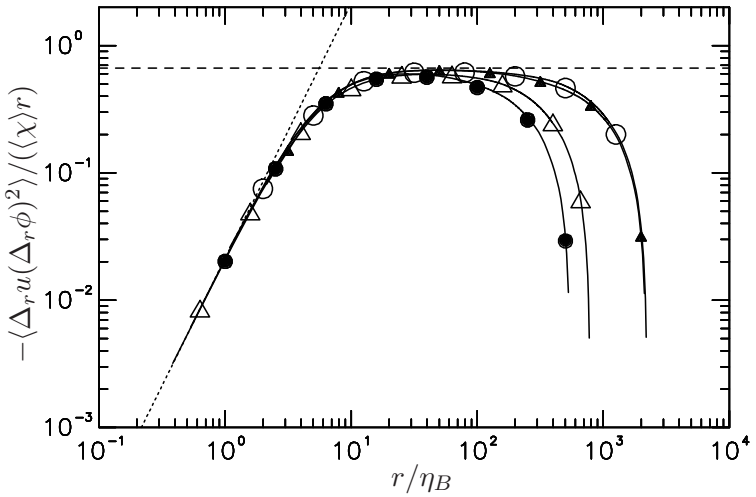

Figure 4.10: Test of Yaglom's relation based on the (component-averaged) mixed velocity-scalar third-order structure function, versus separation scaled by the Batchelor scale. Reynolds and Schmidt number combinations shown are: (i) $R_\lambda \approx 650$ and $Sc = 1/8$ ($\triangle$), (ii) $R_\lambda \approx 650$ and $Sc = 1$ ($\bigcirc$), (iii) $R_\lambda \approx 140$ and $Sc = 4$ ($\bullet$), (iv) $R_\lambda \approx 140$ and $Sc = 64$ ($\blacktriangle$). The dashed horizontal line is at height 2/3. The dotted line of slope 2 shows the small r limit assuming $\tilde{S}_{u\phi} = -0.5$ (see Table 4.6), which holds approximately for $Sc = O(1)$ or higher.

A match between the lines corresponding to the mixed velocity-scalar structure functions and dashed lines (with positive slope) corresponding to Eq. 4.51 in Fig. 4.11 indicates that the small scales are adequately resolved in the low Schmidt number limit. Past studies (Kerr 1985; Yeung *et al.* 2002) show that $S_{u\phi}$ converges approximately to -0.5 with increasing R_λ for $Sc \sim O(1)$. Table 4.6 shows the values of $S_{u\phi}$ as well as the corresponding quantities $S_{v\phi}$ and $S_{w\phi}$ in the other coordinate directions. Because the mean gradient is in the x direction, $S_{u\phi}$ is systematically larger in magnitude than the other two, indicating non-isotropy. However the difference between the three components decreases with decreasing Schmidt number, indicating that local isotropy may be a valid assumption at low enough Peclet numbers. The component averaged value $\tilde{S}_{u\phi}$ is also shown in the last line of this table. At each given Reynolds number, the mixed derivative skewness is seen to become much weaker in magnitude when the Schmidt number is made very small (with magnitude less than 0.5). It may be noted that the numerators in the definitions of $S_{u\phi}$, $S_{v\phi}$ and $S_{w\phi}$ appear, presumably as dominant terms, in the nonlinear amplification term in the scalar dissipation transport equation (Eq. 4.13). Accordingly, the decrease in mixed derivative skewness at low Sc is consistent with the weakening of alignment between scalar gradients

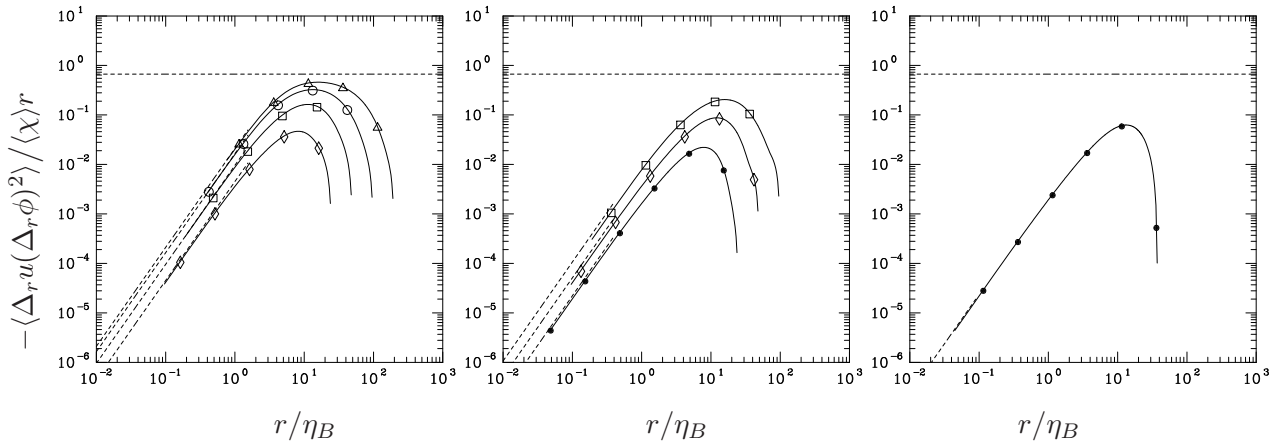

Figure 4.11: Scaling of mixed third-order velocity-scalar structure function for strongly diffusive scalars. Frames and symbols correspond to notation used in Fig. 4.5. The dashed horizontal line is at height 2/3 for comparison with Yaglom's result (Eq. 4.2). Dashed lines (with positive slope) is to check small-r behavior (Eq. 4.51)

and strain rate fluctuations observed earlier.

Table 4.6: Mixed gradient skewness along the three orthogonal directions, and their component average, $\tilde{S}_{u\phi} = (S_{u\phi} + S_{v\phi} + S_{w\phi})/3$.

N^3	512^3	512^3	512^3	512^3	1024^3	1024^3	1024^3	2048^3
R_λ	140	140	140	140	240	240	240	390
Sc	1/8	1/32	1/128	1/512	1/128	1/512	1/2048	1/2048
$S_{u\phi}$	-0.642	-0.528	-0.331	-0.178	-0.324	-0.174	-0.092	-0.083
$S_{v\phi}$	-0.404	-0.300	-0.162	-0.067	-0.208	-0.086	-0.030	-0.043
$S_{w\phi}$	-0.423	-0.322	-0.175	-0.069	-0.248	-0.091	-0.031	-0.040
$\tilde{S}_{u\phi}$	-0.490	-0.383	-0.223	-0.105	-0.260	-0.117	-0.051	-0.055

4.7 Scalar structure function budget - balance of terms

In this section we show results for the two-point scalar structure function budget (Eq. 4.23). The formulae used to compute the production, diffusion, longitudinal convection and transverse convection are given by Eqs. 4.29, 4.34, 4.40 and 4.42. The smallest spatial separation possible in the computations is restricted by the grid spacing while the largest spatial separation is limited by the length of the finite domain. We compare the results with the theoretical limits given in previous section. In view of the Yaglom's relation not holding in the low Schmidt number limit, we propose an alternative scaling law in the inertial-diffusive limit.

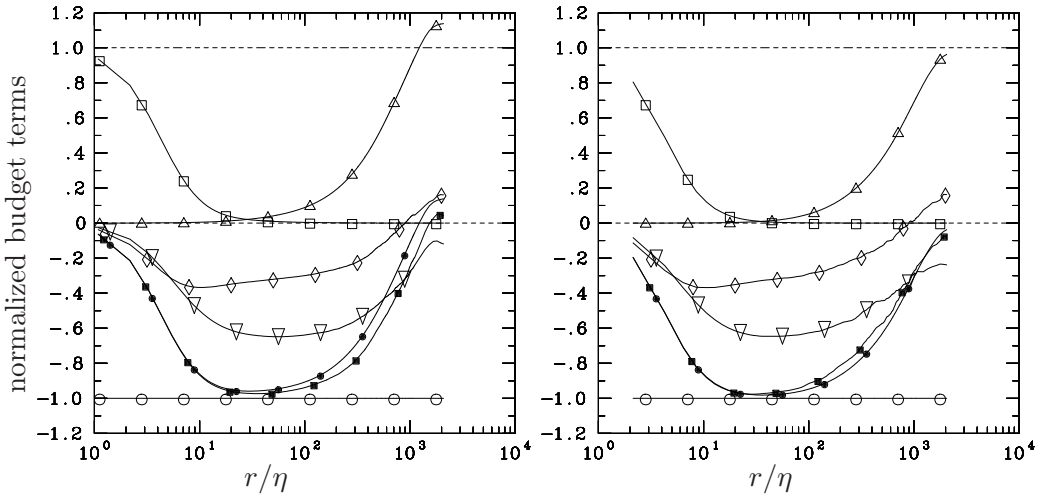

Figure 4.12: Balance of terms in the scalar structure function budget, for (left) R_λ=650, $Sc = 1$; (right) R_λ=140, $Sc = 64$. Each type of term in the equation is represented by a different symbol: longitudinal convection ($\diamond$), transverse convection (∇), production ($\triangle$), molecular diffusion ($\square$), dissipation ($\bigcirc$), all normalized by $2\langle\chi\rangle$. Closed circles and triangles represent the total of terms on the l.h.s and r.h.s respectively of Eq. 4.23.

Figure 4.12 shows the balance of terms for the cases of highest Reynolds number and highest Schmidt number presented earlier in Fig. 4.10. The sums of terms on each side of the budget equation are seen to follow each other closely, while some mild discrepancies are evident at the large scales. If the calculations are accurate then these discrepancies are an indication that the unsteady term in the structure function budget has not completely vanished through ensemble averaging performed over a number of instantaneous snapshots in the simulation. This effect is more substantial at the large scales since the large scales evolve slowly in time and only a limited number of samples of them exist in a finite solution domain at any given time. If a very long simulation with the scalar fluctuations in a stationary state were available this imperfection in the data can certainly be reduced. Nevertheless the present data in this figure show clearly that an intermediate scale range exists where both production and molecular diffusion terms are weak compared to advection and dissipation.

Figure 4.13 shows the budget plots for a range of Schmidt numbers and Reynolds numbers at low-to-moderate Peclet numbers. As Schmidt number decreases the longitudinal and transverse convective terms decrease in magnitude. At Schmidt number 1/2048 at

$R_\lambda \sim 240$ (bottom left in Fig. 4.13), the convective terms are vanishingly small at all separation distances. The inertial-convective range for this case is almost non-existent with the Obukhov-Corrsin scale being larger than the velocity integral scale ($L_1/\eta_{OC} \approx 0.76$). For the same Schmidt number at $R_\lambda \sim 390$ (bottom right in Fig. 4.13), the convective terms are greater in magnitude at intermediate-to-large spatial separations due to a wider range of scales ($L_1/\eta_{OC} \approx 1.47$). This is consistent with Fig. 4.11, which shows that the validity of Yaglom's relation decreases with decreasing Peclet number. Diffusion is dominant at the small scales and becomes negligible at the large scales. The range of scales at which diffusion is important appears to increase with decreasing Peclet number in Fig. 4.23 with diffusive effects prominent even at intermediate-to-large scales for low Schmidt numbers. On the other hand the production mechanism compensates for the dissipation at large scales and monotonically decreases to zero at the small scales. With decreasing Peclet number (Fig. 4.13), production appears to be significant even at the small-to-intermediate scale ranges. This is mainly due to the fact that the range of scales decreases with increasing diffusivity at a fixed Reynolds number. The production term represents large scale effects and is influenced by non-universal factors such as boundary conditions and initial conditions. Hence if the production mechanism was dominant in a given scale range, then universal scaling laws may not hold for such a range. In all cases, we see that both sides of the budget equation are well balanced, possibly except at the largest-r where finite box size and sampling effects can contaminate the results.

In order to contrast the Schmidt number dependence, we report the various terms involved in the structure function budget for different Schmidt numbers in Fig. 4.14. The convective terms decrease in magnitude with decreasing Schmidt number. The magnitude of the transverse convection term is larger than that of the longitudinal part, at moderately low Schmidt numbers. It is worth noting that it is the integration of the longitudinal part that potentially leads to Yaglom's relation (see Eq. 4.11). The decrease in the longitudinal convection term with Schmidt number is the primary reason for Yaglom's relation not holding in the very low Schmidt number regime. Production is increasingly prominent at smaller scales with decreasing Schmidt number as seen in Fig. 4.14. This is due to a decrease in the

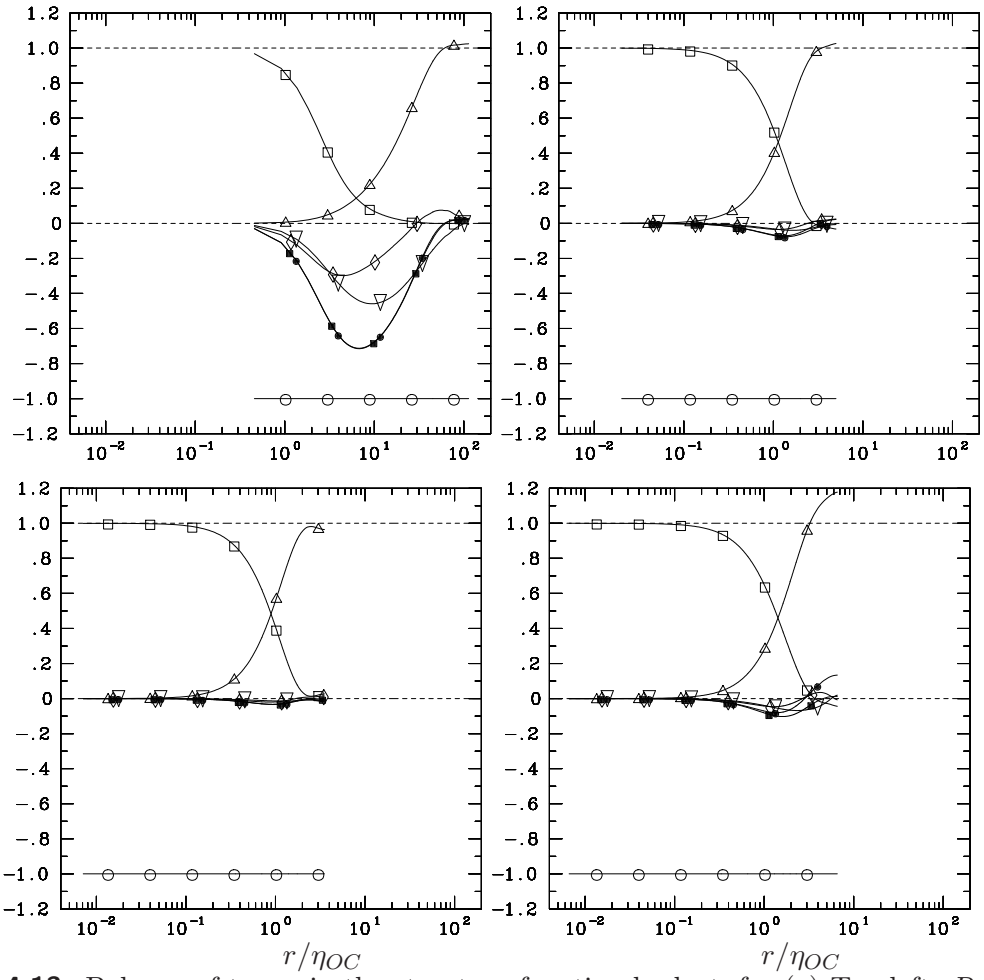

Figure 4.13: Balance of terms in the structure function budget, for (a) Top left: R_λ=140, $Sc = 1/8$; (b) Top right R_λ=140, $Sc = 1/512$; (c) Bottom left: R_λ=240, $Sc = 1/2048$; (d) Bottom right: R_λ=390, $Sc = 1/2048$. Each term is represented by a different symbol: production ($\triangle$), dissipation ($\bigcirc$), longitudinal convection ($\diamond$), transverse convection (∇), molecular diffusion ($\square$). Sum of the production, diffusion and dissipation terms (R.H.S of Eq. 4.23) by ($\bullet$), sum of longitudinal and transverse convection (L.H.S of Eq. 4.23) by ($\blacksquare$). All terms normalized by $2\langle\chi\rangle$.

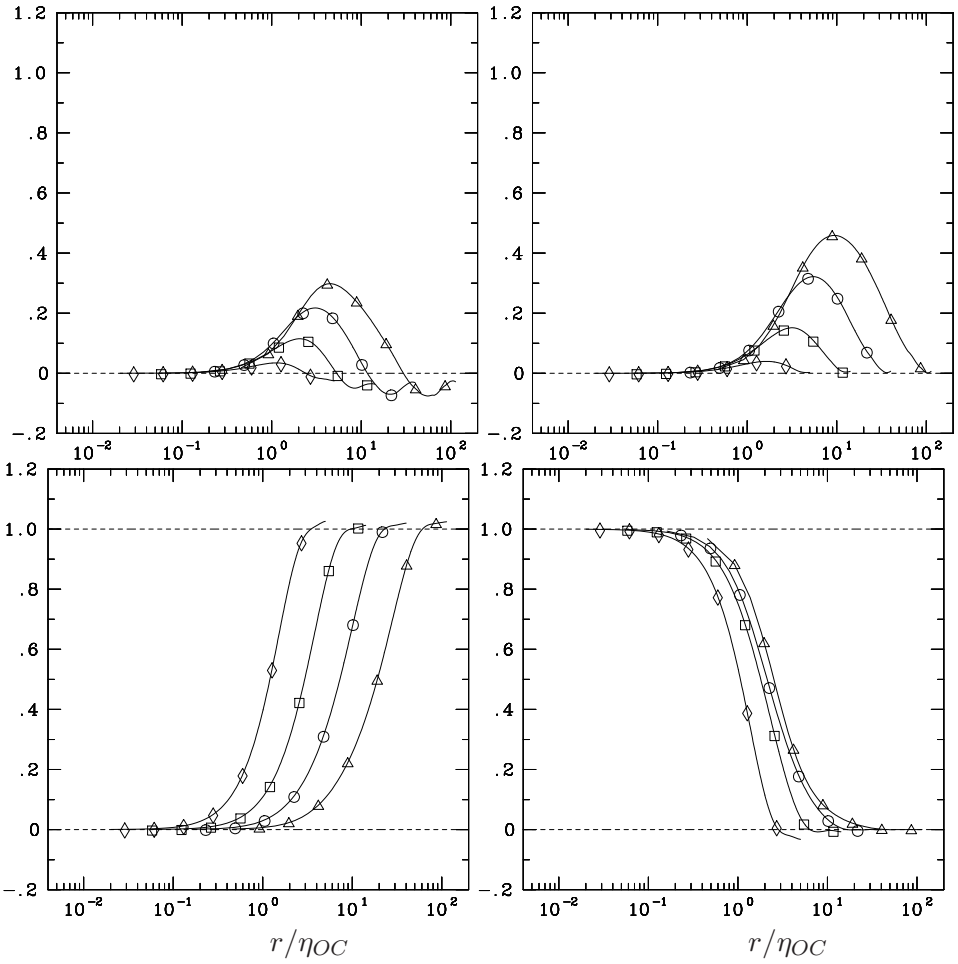

Figure 4.14: Schmidt number dependence of different terms in the scalar structure function budget, normalized by $2\langle\chi\rangle$, in the $R_\lambda \sim 140$ simulation, at Schmidt numbers 1/8 (△), 1/32 (◯), 1/128 (□), and 1/512 (◇). Upper-left: negative of advection (longitudinal); Upper-right: negative of advection (transverse); Lower-left: production; Lower-right: molecular diffusion.

range of scales with decreasing Schmidt number and is indicative of non-universal effects influencing the dynamics in the moderate-to-low Peclet number regime. The diffusive term increases in magnitude with decreasing Schmidt number. Diffusive effects increasingly dominate over a wider range of scales with decreasing Schmidt number and lead to a shrinking of the inertial convective range.

4.7.1 Inertial-Diffusive regime

As seen from the structure function budget plots, the diffusive effects become increasingly dominant over a wider range of scales with decreasing Schmidt number. The increasing diffusivity causes the inertial-convective range to shrink, where the Yaglom's relation is expected to hold. The growth of the small scales in the scalar field can lead to a inertial-diffusive regime if the Reynolds number is sufficiently high and the Schmidt number, sufficiently low. That is if, $\eta \ll r \ll \eta_{OC} \ll (L_1, L_\phi)$, then both inertial and diffusive effects are important, whereas the viscous and scalar production mechanisms are negligible in such a scale range. In particular, the mean scalar gradient term in Eq. 4.23 can be expected to be small, in the inertial-diffusive regime. Neglecting the production and unsteady terms in Eq. 4.23, assuming isotropy at the relevant scale range ($\eta \ll r \ll \eta_{OC}$) and integrating twice we get,

$$-\frac{\langle \Delta_r u (\Delta_r \phi)^2 \rangle}{\langle \chi \rangle r} + 2D_\phi \frac{\langle \Delta\phi(r)^2 \rangle}{\langle \chi \rangle r^2} = \frac{1}{3} \,, \tag{4.53}$$

where the first term represents the inertial effects while the second term corresponds to the diffusive effects. For convenience we use the following notation,

$$A(r) = -\frac{\langle \Delta_r u (\Delta_r \phi)^2 \rangle}{\langle \chi \rangle r} \,, \tag{4.54}$$

$$B(r) = 2D_\phi \frac{\langle \Delta\phi(r)^2 \rangle}{\langle \chi \rangle r^2} \,. \tag{4.55}$$

such that $A(r) + B(r) = 1/3$ in the inertial-diffusive range. If isotropy does not hold in the inertial-diffusive range, such as in the case of a non-zero mean gradient, the neglect of the production and unsteady terms in Eq. 4.23 leads to

$$-\frac{\partial \langle \Delta u_i(\mathbf{r}) \Delta\phi(\mathbf{r})^2 \rangle}{\partial r_i} + 2D_\phi \frac{\partial^2 \langle \Delta\phi(\mathbf{r})^2 \rangle}{\partial r_j \partial r_j} = 2\langle \chi \rangle \tag{4.56}$$

Consider a volume integral of above equation over a sphere of radius r. The first term on the left-hand-side of this equation is the divergence term. As before, we use Gauss's theorem to convert it into an integral over the sphere of the vector $\langle \Delta\mathbf{u}(\mathbf{r})\Delta\phi(\mathbf{r})^2\rangle$ projected onto $\mathbf{r}$, which is in the outward normal direction. The surface integral yields $\langle \Delta u_L(r)\Delta\phi(r)^2\rangle_{sp}$ as before. The second term on the left-hand-side is the Laplacian of the second-order scalar structure function whose volume integral can be evaluated using Gauss's theorem as follows,

$$\begin{aligned}
\int_{\mathcal{V}_r} \nabla\cdot\nabla\langle\Delta\phi(\mathbf{r})\rangle dV &= \int_{\partial\mathcal{V}_r} \nabla\langle\Delta\phi(\mathbf{r})\rangle\cdot\mathbf{dS} \\
&= \int_{\partial\mathcal{V}_r} \nabla\langle\Delta\phi(\mathbf{r})\rangle\cdot r^2\hat{\mathbf{r}}d\Omega \\
&= r^2\int_{\partial\mathcal{V}_r} \frac{\partial\langle\Delta\phi(r)^2\rangle}{\partial r}d\Omega \\
&= 4\pi r^2\frac{d}{dr}\frac{1}{4\pi}\int\langle\Delta\phi(r)^2\rangle d\Omega
\end{aligned} \tag{4.57}$$

Where the subscript $\partial\mathcal{V}_r$ denotes integration on the surface of a sphere of radius r and Ω is the solid angle subtended at the center of the sphere such that $\int d\Omega = 4\pi$. Defining the spherical average for the scalar structure function as

$$\langle\Delta\phi(r)^2\rangle_{sp} = \frac{1}{4\pi}\int\langle\Delta\phi(r)^2\rangle d\Omega\ , \tag{4.58}$$

substituting the spherical average definition into Eq. 4.57, using this result in the volume integration of Eq. 4.56, we get

$$-\langle\Delta u_L(r)\Delta\phi(r)^2\rangle_{sp} + 2D_\phi\frac{d}{dr}\langle\Delta\phi(r)^2\rangle_{sp} = \frac{2}{3}\langle\chi\rangle r \tag{4.59}$$

Integrating again with respect to r and rearranging, we get

$$-\frac{\langle\Delta u_L(r)\Delta\phi(r)^2\rangle_{sp}}{\langle\chi\rangle r} + 2D_\phi\frac{\langle\Delta\phi(r)^2\rangle_{sp}}{\langle\chi\rangle r^2} = \frac{1}{3} \tag{4.60}$$

The above equation is similar in form to the inertial-diffusive relation obtained using isotropy (Eq. 4.53).

Figure 4.15 shows Eq. 4.53 as a function of spatial separation for a range of Schmidt numbers considered at two different Reynolds numbers. At the higher Reynolds number, we see a tendency towards a plateau in the range $10\eta < r < 40\eta$. These simulations do not have a well-defined inertial diffusive regime because of a lack of sufficient scale separation.

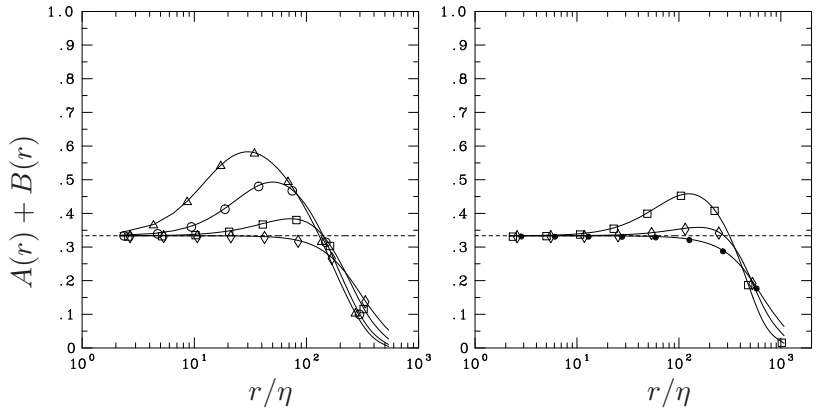

Figure 4.15: Inertial-diffusive scaling for (a) (left) $R_\lambda \sim 140$, (b) (right) $R_\lambda \sim 240$. Each choice of symbol on the curves denote a different value of Sc: 1/8 ($\triangle$), 1/32 ($\bigcirc$), 1/128 ($\square$) and 1/512 ($\diamond$) and 1/2048 ($\bullet$). Horizontal line at 1/3 for comparison with Eq. 4.53.

Nevertheless, we see the emergence of a plateau like behavior. for the higher Reynolds number case in Fig. 4.15.

Based on the above analysis, an inertial-diffusive result can be postulated for the asymptotic case of infinite Reynolds number and vanishing Schmidt number. The magnitude of the advection term, $A(r)$ increases with increasing spatial separation at small-r, as seen in Figs. 4.13 and 4.14. In contrast, the diffusion term $B(r)$ is largest at the smallest scales and monotonically decreases with increasing spatial separation. It follows that, in the inertial-diffusive range, the magnitudes of the advection and diffusion terms are roughly the same, that is $A(r) \approx B(r)$. Hence, in the limit, $R_\lambda \to \infty$ and $Sc \to 0$, Eq. 4.53 can be approximated as

$$-\frac{\langle \Delta_r u (\Delta_r \phi)^2 \rangle}{\langle \chi \rangle r} = \frac{1}{6} \tag{4.61}$$

The above equation is a scaling result in the inertial-diffusive range, $\eta \ll r \ll \eta_{OC} \ll (L_1, L_\phi)$. Due to the low-to-moderate Peclet number simulations in our DNS database, this equation has not been tested yet. A rigorous test of this scaling relationship may also require a DNS simulation without a mean scalar gradient to ensure that the largest scales and the dissipative scales in the scalar field are well separated in space.

4.7.2 Diffusive limit

For the smallest scales in the flow, $r \ll (\eta, \eta_{OC}) \ll (L_1, L_\phi)$, diffusive effects are important. If the Reynolds number is not sufficiently high and the Schmidt number is sufficiently low (such that the Peclet number is low), a sufficient scale separation may not exist for a well-defined inertial-diffusive range. In such a scenario, the diffusive effects dominate at the small scales such that $A(r) \ll B(r)$ for $r \ll \eta_{OC}$ (see Eqs. 4.53, 4.54 and 4.55). Thus in the low Peclet number limit, the diffusive range ($r \ll \eta_{OC}$) relation,

$$B(r) = 2D_\phi \frac{\langle \Delta\phi(r)^2 \rangle}{\langle \chi \rangle r^2} = \frac{1}{3} \tag{4.62}$$

is obtained. As before if isotropy does not hold then, $\langle \Delta\phi(r)^2 \rangle$ can be replaced with $\langle \Delta\phi(r)^2 \rangle_{sp}$. The diffusive range relation is consistent with Taylor series result in the ballistic limit $r \ll \eta_{OC}$ given in Eq. 4.5.

CHAPTER V

EXTRACTING ISOTROPIC STATISTICS IN TURBULENCE SIMULATIONS

Scaling laws in turbulence theory are important partly because they help overcome the closure problem, at least in the relevant scale-range. The Kolmogorov (1941b) theory gives an explicit relation for the isotropic component of the third-order velocity structure function in the intermediate scale range, in the infinite Reynolds number limit. The analogous relation for the scalar field is the Yaglom (1949) relation for the isotropic part of the third-order mixed velocity-scalar structure function in the infinite Peclet number limit. Such scaling laws are said to be "exact", which means that they hold in the asymptotic case of very large Reynolds and/or Schmidt number. Empirical validation of such laws is naturally curtailed by the finite Reynolds and Schmidt numbers that are possible in practice. Both experiments and direct numerical simulations(DNS) have striven to obtain isotropic statistics by time-averaging structure functions calculated along a few directions, over multiple eddy-turnover times. In the case of DNS, forcing schemes carefully constructed to generate isotropic data have been used (Eswaran & Pope 1988). However large scale effects still persist in the intermediate and small scale ranges to varying degrees both in experiments and simulations (Sreenivasan & Antonia 1997). Finite sampling further compounds the problem of flow field anisotropy. In the case of the scalar field, the presence of a non-zero mean gradient results in a systematic difference in statistics parallel and perpendicular to the mean gradient, making it even harder to extract isotropic statistics (Warhaft 2000). In such a scenario, it is difficult to objectively assess scaling laws involving isotropic statistics.

Theoretical work by Nie & Tanveer (1999) and Eyink (2003) show that the local isotropy requirements can be relaxed by spherically averaging structure functions over all directions. Such averages do not seem to require at least in theory, homogeneity or isotropy. In light of these theories, spherical averaging can be interpreted as a technique to extract the isotropic

component of a given quantity from anisotropic fields. Taylor *et al.* (2003) have given a numerical recipe that calculates such spherical averages by approximating them as angle averages. With the aim of reducing effects of anisotropy and finite sampling on isotropic statistics we have implemented the numerical procedure given in Taylor *et al.* (2003). We have developed a parallel algorithm to compute spherical averages, that is particularly suited for very large data sets. We have used the latest computational techniques in hybrid (MPI/OpenMP) and OpenACC programming to achieve scalability on different parallel machines.

We have calculated isotropic statistics in both the velocity and scalar fields using the angle averaging technique given in Taylor *et al.* (2003). We find that subject to Reynolds limitations, the spherical averaging technique gives a wider scaling range in the velocity field. Because of better statistical convergence, we also require fewer snapshots of data when using this method. We also test the local laws of Nie & Tanveer (1999) and Eyink (2003) using the spherical averaging algorithm on our DNS database. For the scalar field, we find that we require greater resolution when using the spherical averaging process. We conjecture that we require both high resolution and high Peclet numbers to realize the asymptotic scaling laws for passive scalar mixing.

5.1 Kolmogorov $4/5$*th law*

An exact result for the incompressible Navier-Stokes equations was derived by Kolmogorov (1941b) in the infinite Reynolds number limit under the conditions of statistical homogeneity and local isotropy and is given as

$$\begin{aligned} \langle [\Delta u_L(\mathbf{r})]^3 \rangle &= -\frac{4}{5}\langle \epsilon \rangle r \ , \\ \Delta u_L(\mathbf{r}, \mathbf{x}) &= [\mathbf{u}(\mathbf{x}+\mathbf{r}) - \mathbf{u}(\mathbf{x})] \cdot \hat{\mathbf{r}} \ , \\ \hat{\mathbf{r}} &= \mathbf{r}/r \ , \end{aligned} \tag{5.1}$$

where, $\langle \cdot \rangle$ denotes ensemble averaging. The left hand side of Eq. 5.1 is the third-order longitudinal structure function. The length scale r must lie in the inertial range $\eta \ll r \ll L$, sufficiently far from the large scales L and the dissipation scales given by the Kolmogorov

scale η. The mean energy dissipation rate per unit mass is given by $\langle\epsilon\rangle$. Equation 5.1 is commonly known as the "4/5 law" and is one of the few exact, nontrivial results in the theory of statistical hydrodynamics. The derivation of Eq. 5.1 has an unproven assumption that the energy dissipation rate has a strictly positive limit as viscosity tends to zero.

Using the incompressibility constraint (Monin & Yaglom 1975) Eq. 5.1 can be restated as the "4/15 law" and the "4/3 law". For completeness we provide these equations:

$$\langle \Delta u_L(r)(\Delta u_T(r))^2\rangle = -\frac{4}{15}\langle\epsilon\rangle r \ , \tag{5.2}$$

$$\langle \Delta u_L(r)|\Delta \mathbf{u}_T(r)|^2\rangle = -\frac{4}{3}\langle\epsilon\rangle r \ , \tag{5.3}$$

where $|\Delta\mathbf{u}|$ is the magnitude of the velocity difference across separation vector $\mathbf{r}$ and $\Delta u_T(r) = \hat{\mathbf{t}}\cdot\Delta\mathbf{u}(r)$ is a transverse velocity increment,with $\hat{\mathbf{t}}$ any unit vector orthogonal to $\mathbf{r}$. Equations 5.1, 5.2 and 5.3 are referred to hereafter as K41-3.

Both experiments and numerical simulations use the K41-3 results as a measure to assess the width of the inertial range. The validity of these equations also constitutes as an important test for the boundedness property of the dissipation rate (Frisch 1995). Experiments in high Reynolds number turbulence in general support the linear scaling of the third-order structure function in the inertial range (Yeung *et al.* 2005). However, the convergence to the asymptotic coefficient of 4/5 in Eq. 5.1 has been quite slow (Gotoh *et al.* 2002; Antonia & Burattini 2006). In both experiments and numerical simulations, large volumes of data extending over many eddy-turnover time and/or integral length scales is required to obtain these isotropic statistics.

A modified version of the 4/5 law which does not assume local isotropy exists. Nie & Tanveer (1999) proved that the 4/3 and hence the 4/5 law can be recovered in homogeneous, but not necessarily isotropic flows:

$$\langle(\Delta u_L(r))^3\rangle = \lim_{T\to\infty}\frac{1}{T}\int_0^T dt\int\frac{d\Omega}{4\pi}\int\frac{d\mathbf{x}}{L^3}[\Delta u_L(\mathbf{r},\mathbf{x},t)]^3 \tag{5.4}$$

$$= -\frac{4}{5}\langle\epsilon\rangle r \ . \tag{5.5}$$

The integration over $\mathbf{x}$ is over the entire volume $L^3 = \int d\mathbf{x}$. The angle integration in $\mathbf{r}$ is over the sphere of radius r with $\int d\Omega = 4\pi$ being the solid angle subtended at the center of

a unit sphere.For each point $\mathbf{x}$ the vector increment $\mathbf{r}$ is allowed to vary over all angles, and the resulting longitudinal moments are considered. The integration over time t extends over long times (T). The integration over long-times is consistent with the ensemble averages used in K41 theory (Eq. 5.1) due to ergodicity (Frisch 1995).

A local version of the 4/3 law was derived by Duchon & Robert (2000). Subsequently Eyink (2003) derived the corresponding versions of the 4/5 and 4/15 laws. Following Taylor *et al.* (2003), we state the local law as follows: Given any local region B of size R, for $r \ll R$, and in the limits $\nu \to 0$, then $r \to 0$, and finally $\delta \to 0$,

$$\begin{aligned} \langle (\Delta u_L(r))^3 \rangle_{\Omega,B} &= \lim_{\delta \to 0} \frac{1}{\delta} \int_t^{t+\delta} d\tau \int \frac{\Omega}{4\pi} \int_R \frac{d\mathbf{x}}{R^3} [\Delta u_L(\mathbf{r}; \mathbf{x}, \tau)]^3 \\ &= -\frac{4}{5} \langle \epsilon_B \rangle r \ , \end{aligned} \tag{5.6}$$

for almost every (Lebesgue) point t in time, where $\langle \epsilon_B \rangle$ is the instantaneous (in time) mean energy dissipation rate over the local region B. Equation 5.6 does not require stationarity, homogeneity, or isotropy of the flow. Long-time or ensemble averages are also not required. The Duchon & Robert (2000) and Eyink (2003) versions are truly local in space and time.

Our velocity fields are nominally isotropic. Individual snapshots are not necessarily isotropic, only the ensemble of all snapshots is guaranteed to be isotropic. Hence we rely on long-time averages of data spanning multiple eddy-turnover times to obtain isotropic statistics. This invariably leads to longer simulation times, which can be prohibitively expensive at higher Reynolds number. Furthermore, saving the velocity and scalar fields at every checkpoint and having to read the data again accrues significant I/O time. The finite length of the solution domain is also a limitation. Large scale isotropic forcing schemes such as Eswaran & Pope (1988) can also cause some anisotropy at the smaller scales. Increasing the length of the domain does improve statistical sampling, but at a considerable computational expense. Figure 5.1 shows the left hand side of Eq. 5.1 along the three orthogonal directions for the highest Reynolds number in this study. The structure functions differ at the intermediate scale range possibly due to anisotropy at this scale size.

We are motivated by the desire to extract isotropic statistics particularly in the intermediate flow ranges, where anisotropic effects can be significant. We wish to differentiate

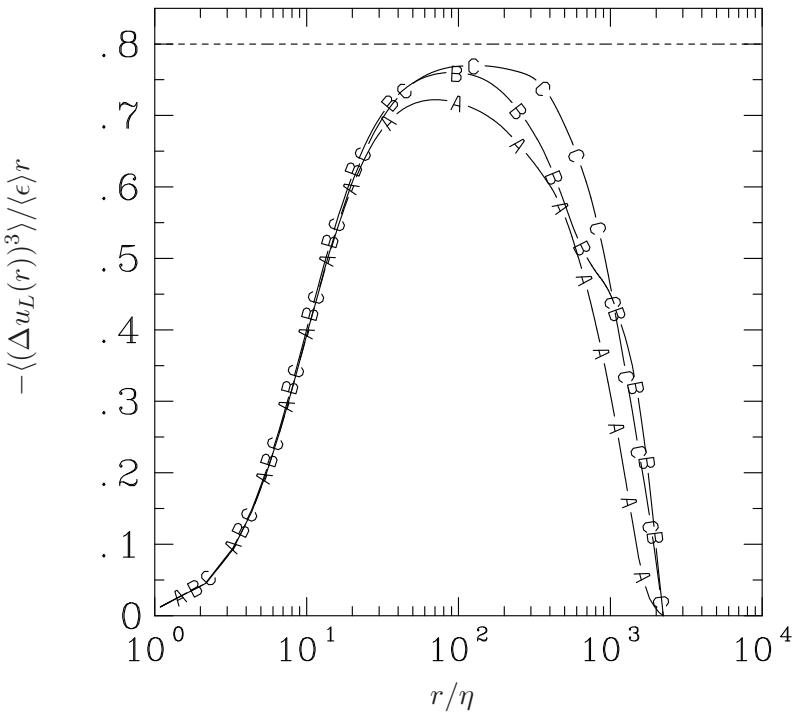

Figure 5.1: Third order longitudinal velocity structure function (Eq. 5.1) averaged over 7 snapshots for $R_\lambda \sim 650{,}4096^3$, simulation. Lines A,B,C correspond to the Cartesian directions $(1,0,0)$,$(0,1,0)$ and $(0,0,1)$ respectively. Horizontal line at 4/5 shows the asymptotic K41 constant (Eq. 5.1).

between the effects of finite Reynolds number and anisotropy on the width of the inertial range. This will enable a more systematic study of the Reynolds number dependence of the small-scale phenomenology.

The concept of averaging over a sphere to extract isotropic data has existed for some time. Present high-Reynolds number experiments only enable data measurements at a few spatial locations. Taylor's hypothesis is typically used to convert spatial derivatives into time derivatives. Such methods are not suitable to spherical averaging. Numerical methods such as direct numerical simulations provide complete space-time information of the flow. The new spherical- averaged and local laws (Nie & Tanveer 1999; Eyink 2003) provide us with the theoretical impetus to extract isotropic components of the flow in high Reynolds number turbulence. The interpolation of square grid data onto spherical shells is very expensive (Nie & Tanveer 1999). We use a method of taking averages over angles which avoids the expense of interpolating square-grid data over spherical shells.

Taylor *et al.* (2003) have proposed a numerical scheme to approximate a spherical average by averaging over a finite number of angles. This method has been shown to be a good approximation to a true spherical average. We have adapted the numerical scheme given in Taylor *et al.* (2003) with a different parallel implementation suited towards very large problem sizes.

5.2 Angle-averaging technique

We would like to extract the isotropic component by a suitable average of the of the two-point structure function $\mathbf{u}(\mathbf{x}+\mathbf{r})-\mathbf{u}(\mathbf{x})$, as defined by Eqs. 5.4 and 5.6. Following Taylor *et al.* (2003) we approximate the spherically averaged third-order longitudinal structure function by the following average over N_d directions:

$$\langle[\Delta u_L(r)]^3\rangle = \frac{1}{N_d}\frac{1}{N^3}\sum_{j=1}^{N_d}\sum_{i=1}^{N^3} w_j[\Delta u_L(\mathbf{r}_j;\mathbf{x}_i)]^3 \,, \tag{5.7}$$

where $\mathbf{x}_i$ denotes grid points, $\mathbf{r}_j$ denotes the increment vector in the jth direction, $r=|\mathbf{r}_j|$ is fixed and w_j are quadrature weights.

We choose vectors $\mathbf{r}_j$ $(j=1,\ldots,N_d)$ from among those that are natural to a square computational grid. We restrict ourselves to the set of all unique directions which can be expressed with integer components with length less than or equal to $\sqrt{11}$. Each vector $\mathbf{r}_j$ $(j=1,\ldots,N_d)$ is the minimum grid-point separation distance in the jth direction. This set is generated by vectors

$$\mathcal{B} = \Big\{(1,0,0),(1,1,0),(1,1,1),(2,1,0),(2,1,1),(2,2,1),(3,1,0),(3,1,1)\Big\}, \tag{5.8}$$

by taking all index and sign permutations of the three coordinates. Vectors that are positive or negative multiples of any other vector in this set are removed. This procedure generates a total of $N_d=73$ unique directions. For convenience, we consider the set $\widehat{\mathcal{B}}$ whose elements have the same direction as those in $\mathcal{B}$, but with unit magnitude,

$$\begin{aligned}\widehat{\mathcal{B}} \;=\; &\Big\{(1,0,0),(\tfrac{1}{\sqrt{2}},\tfrac{1}{\sqrt{2}},0),\Big(\tfrac{1}{\sqrt{3}},\tfrac{1}{\sqrt{3}},\tfrac{1}{\sqrt{3}}\Big),\Big(\tfrac{2}{\sqrt{5}},\tfrac{1}{\sqrt{5}},0\Big),\\ &\Big(\tfrac{2}{\sqrt{6}},\tfrac{1}{\sqrt{6}},\tfrac{1}{\sqrt{6}}\Big),\Big(\tfrac{2}{3},\tfrac{2}{3},\tfrac{1}{3}\Big),\Big(\tfrac{3}{\sqrt{10}},\tfrac{1}{\sqrt{10}},0\Big),\Big(\tfrac{3}{\sqrt{11}},\tfrac{1}{\sqrt{11}},\tfrac{1}{\sqrt{11}}\Big)\Big\}.\end{aligned} \tag{5.9}$$

The elements of the set $\widehat{\mathcal{B}}$ and the vectors formed by permuting the indices and sign of elements in $\widehat{\mathcal{B}}$, removing any linearly dependent redundant vectors, together are referred to as *generator points*. We get $N_d = 73$ generator points in total. We refer to the collection of all generator points as the *generator set* and introduce the following notation for later use:

$$\mathcal{G} = \{\hat{\mathbf{r}}_1, \hat{\mathbf{r}}_2, \ldots, \hat{\mathbf{r}}_{N_d}\}, \quad |\hat{\mathbf{r}}_i| = 1, \;\; i = 1, \ldots, N_d \; . \tag{5.10}$$

Figure 5.2 shows the elements $\mathcal{G}(\hat{\mathbf{r}}_j)$ $(j = 1, \ldots, N_d)$ of the generator set $\mathcal{G}$ on the unit sphere in a Cartesian grid. Both $\hat{\mathbf{r}}_j$ and $-\hat{\mathbf{r}}_j$, are shown in Fig. 5.2. But we do not consider $-\hat{\mathbf{r}}_j$ directions, since they give the same contributions as $\hat{\mathbf{r}}_j$ when averaged. To see this, let us denote the normalized third-order velocity structure function for a fixed $\mathbf{r}_j = r\hat{\mathbf{r}}_j$ as:

$$f(r) = \frac{\left\langle [u(x+r) - u(x)]^3 \right\rangle}{\langle\epsilon\rangle r} \tag{5.11}$$

Then,

$$f(-r) = -\frac{\left\langle [u(x-r) - u(x)]^3 \right\rangle}{\langle\epsilon\rangle r} \tag{5.12}$$

Using the translation $x - r = k$ and invoking homogeneity, we get

$$f(-r) = -\frac{\left\langle [u(k) - u(k+r)]^3 \right\rangle}{\langle\epsilon\rangle r} = f(r) \; . \tag{5.13}$$

Hence we do not consider the directions $-\hat{\mathbf{r}}_j$ (where $\hat{\mathbf{r}}_j \in \mathcal{G}$), in the following analysis.

The magnitude $|\mathbf{r}_j|$ of the minimum grid spacing in the jth direction for a N^3 domain of length L_0 is gives as

$$|\mathbf{r}_j| = \lambda_j \frac{L_0}{N}, \quad j = 1, \ldots, N_d, \tag{5.14}$$

where λ_j is the magnitude of a given element in the set $\mathcal{B}$. It can be seen from the above equation that the grid spacing along the non-orthogonal directions is greater than $\Delta x = L_0/N$. For instance, for the body diagonal vector $(1, 1, 1)$ we have $\lambda = \sqrt{3}$. This gives the minimum grid spacing along $(1, 1, 1)$ as $\sqrt{3}\Delta x$.

For each of the N_d directions we form a set of $l = 1, \ldots, N_r$ separation vectors $\mathbf{x}_i + l\mathbf{r}_j$. Since $\mathbf{r}_j$ is the minimum separation distance of grid points in the jth direction and l is an integer, all the $\mathbf{x}_i + l\mathbf{r}_j$ points lie on our computational grid. This is illustrated (in two

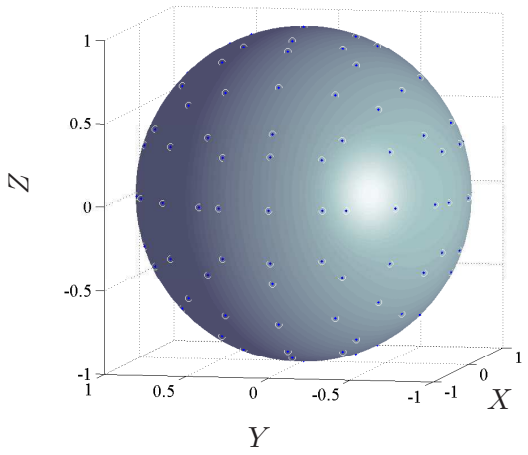

Figure 5.2: Unit sphere in Cartesian coordinate grid showing some of the $N_d = 73$ directions considered.

dimensions) in Fig. 5.3, where the black dots represent $\mathbf{x} + l\mathbf{r}_j$ for the case of $\mathbf{x}_i$ at the origin.

We can now compute structure functions along N_d directions at N_r separation distances as

$$\langle [\Delta u_L(l\mathbf{r}_j)]^3 \rangle = \frac{1}{N^3} \sum_{i=1}^{N^3} [\Delta u_L(l\mathbf{r}_j; \mathbf{x}_i)]^3. \tag{5.15}$$

For each direction, we get a one-dimensional curve as a function of $l\mathbf{r}_j$, where $l = 1, \ldots, N_r$.

For a N^3 grid with length L_0, the non-trivial separation distances expressed as a multiple of the grid spacing $\Delta x = L_0/N$ are $r = 1, \ldots, N/2$ because of periodicity. This implies that $N_r = N/2$ for any vector $\mathbf{r}_j$ $(j = 1, \ldots, N_d)$. In the forced simulations considered in this study, all Fourier modes with magnitude $|\mathbf{k}_F| \leq 2.1$ excluding $|\mathbf{k}_F| = 0$ are forced. This implies that all scales with magnitude $r \geq L_0/2.1$ are directly energized by the numerical forcing. The largest scales are also affected by finite box length and periodicity constraints to a greater extent than the smaller scales. These effects can often times have unpredictable behavior in the statistics of the largest scales in the flow. Figure 5.4 shows that the third order velocity structure function $\langle [\Delta u_L(r)]^3 \rangle$ becomes positive at the largest separation distances for the non-Cartesian direction. Scale separation along the non-Cartesian directions are larger than those along the Cartesian directions. Hence a greater range of scales along

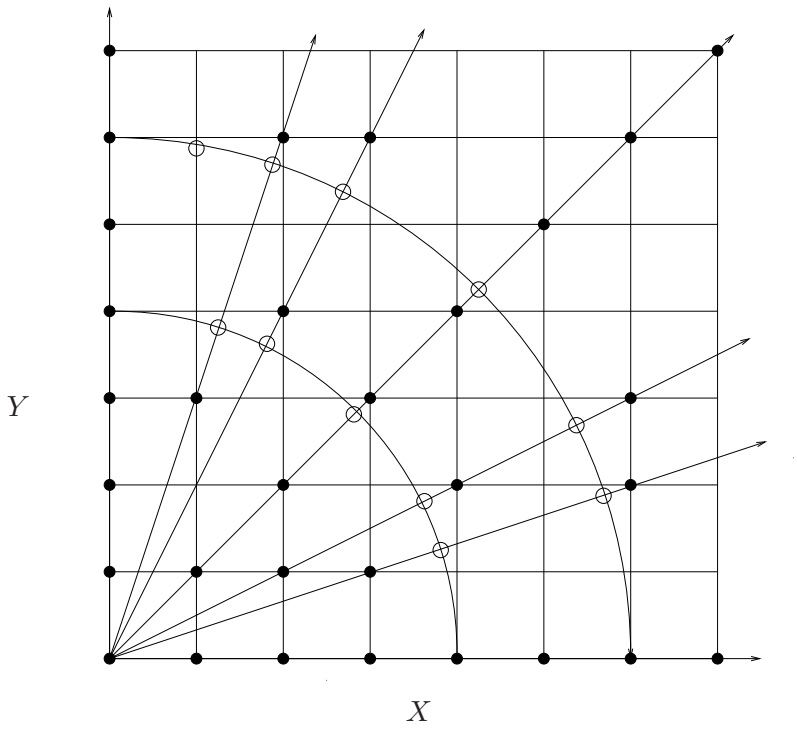

Figure 5.3: Two-dimensional example in the X-Y plane for the angle averaging procedure. Seven directions (in counter-clockwise direction) are shown: $(1,0)$, $(3,1)$, $(2,1)$, $(1,1)$, $(2,4)$, $(1,3)$ and $(0,1)$. Closed circles show grid points where structure functions for a particular direction can be computed with no interpolation. Structure functions at open circles (required for angle average) can then be obtained using interpolation.

these directions come under the direct influence of the numerical forcing. For computational efficiency we only consider those separation distances that are not directly impacted by forcing. Hence we force N_r to be a function of $|\mathbf{r}_j|$, and take $N_r = N_r(j)$ as

$$N_r(j) = \left[\frac{L_0/2}{|\mathbf{r}_j|}\right] = \left[\frac{N/2}{\lambda_j}\right] , \tag{5.16}$$

where $\Big[(\cdot)\Big]$ denotes the integer part of $(\cdot)$, $|\mathbf{r}_j|$ is given by Eq. 5.14 and λ_j is the magnitude of the vector in set $\mathcal{B}$ along the jth direction.

We compute the structure functions along the N_d directions at $N_r(j)$ $(j = 1, \ldots, N_d)$ separation distances for each direction:

$$\langle\langle[\Delta u_L(l\mathbf{r}_j)]^3\rangle = \frac{1}{N^3}\sum_{i=1}^{N^3}[\Delta u_L(l\mathbf{r}_j;\mathbf{x}_i)]^3 . \tag{5.17}$$

The evaluation of the right hand side of Eq. 5.17 scales as N^4 for each direction $\hat{\mathbf{r}}_j$ $(j =$

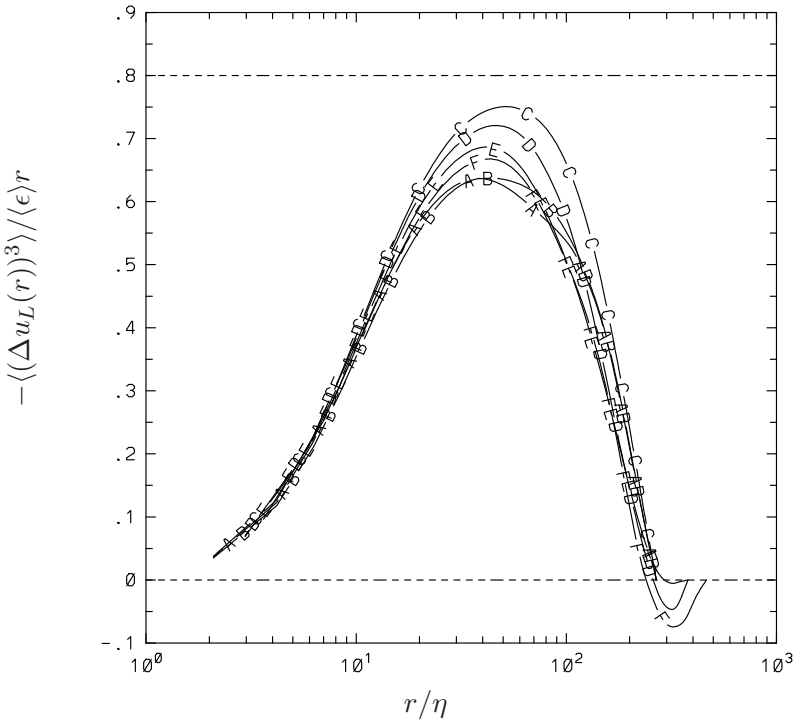

Figure 5.4: Third order velocity structure function computed at all non-trivial separation distances $N_r = 1, \ldots, N/2$ for 256^3,$R_\lambda \sim 140$. Curves A-F correspond to $(1,0,0), (0,1,0), (0,0,1), (1,1,0)$ and $(1,1,1)$ directions respectively. Horizontal dashed line (bottom) at zero to show unpredictable behavior of $\langle[\Delta u_L(r)]^3\rangle$ for scales that are forced in the DNS (see line F for $r > 200\eta$).

$1, \ldots, N_d$). Hence we use a parallel algorithm, the details of which are given in the next section.

The structure functions calculated from Eq. 5.17 cannot be angle-averaged directly, as they are computed at different spatial separations r, for different directions. We use cubic splines to interpolate the structure functions so that we obtain the structure functions at the same spatial separation for all N_d directions. To calculate the end slopes, we compute slopes of $f(r)$ (Eq. 5.11), at small and large separation distances. At small-r, using Taylor series expansion we can write:

$$f(r) \approx r^2 \frac{\left\langle \left(\partial u/\partial x\right)^3 \right\rangle}{\langle\epsilon\rangle} \tag{5.18}$$

Differentiating with respect to r we get (at small-r)

$$\frac{d}{dr} f(r) = f'(r) \approx 2r \frac{\left\langle \left(\partial u/\partial x\right)^3 \right\rangle}{\langle\epsilon\rangle} \tag{5.19}$$

Hence, we get $f'(0) = 0$. For the slope in the large-r limit, we note that $f(r) \to 0$ as $r \to L_0/2$ (L_0 is the length of the domain) because of periodicity and furthermore, this approach to zero is monotonic. Hence we have $f'(L_0/2) = 0$. We use these clamp conditions to interpolate the structure functions over all directions. Figure 5.5 shows only some of the interpolated curves for visual clarity. We see that the cubic spline interpolation is an excellent interpolant. The structure functions along different directions agree with Eq. 5.18 at small spatial separations, while they diverge from each other at intermediate and large separations.

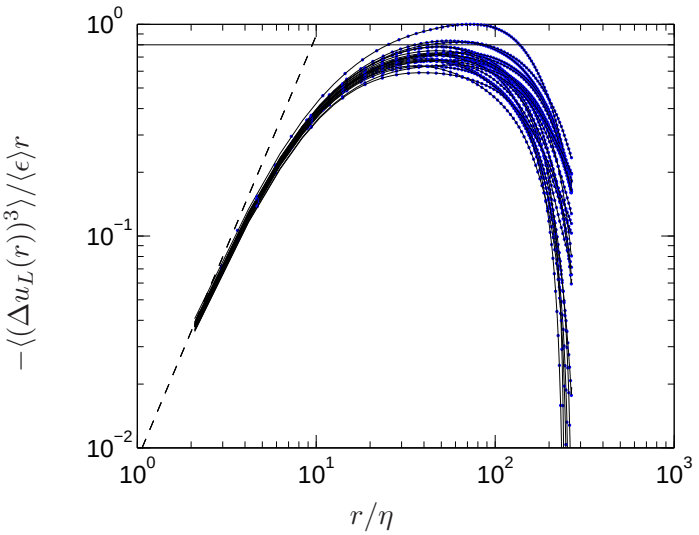

Figure 5.5: The third-order structure function nondimensionalized by $\langle\epsilon\rangle r$ for $R_\lambda \sim 140$, $k_{max}\eta \sim 1.4$. The dots indicate values of the structure function computed at various $l\mathbf{r}_j$. Each curve connecting the dots is the cubic-spline interpolation through all computed values of the structure function in a given direction. Only a few of the 73 curves are shown for clarity. The horizontal solid line indicates the asymptotic constant 4/5 (Eq. 5.1). The dashed line shows the small-r asymptote (Eq. 5.18).

Once the data for each structure function has been interpolated to a common separation distance, we can approximate the angle average over the N_d different directions as

$$\langle[\Delta u_L(r)]^3\rangle = \frac{1}{N_d^3}\sum_{j=1}^{N_d} w_j \langle[\Delta u_L(r\hat{\mathbf{r}}_j)]^3\rangle \; . \tag{5.20}$$

The quadrature weight w_j is the solid angle subtended by the spherical Voronoi polygon (Renka 1984) associated with the jth element $\mathcal{G}(\hat{\mathbf{r}}_j)$ of the generator set $\mathcal{G}$ ($j = 1, \ldots, N_d$). The spherical Voronoi polygon $V(\hat{\mathbf{r}}_i)$ associated with generator point $\hat{\mathbf{r}}_i$ on the unit sphere

S is defined as (Okabe *et al.* 1992)

$$V(\hat{\mathbf{r}}_i) = \left\{ \hat{\mathbf{r}} : d_{gc}(\hat{\mathbf{r}}, \hat{\mathbf{r}}_i) \leq d_{gc}(\hat{\mathbf{r}}, \hat{\mathbf{r}}_j), j \in \mathcal{G} \backslash \{i\}, \hat{\mathbf{r}} \in S \right\} \tag{5.21}$$

where $d_{gc}(\hat{\mathbf{r}}, \hat{\mathbf{r}}_i)$ is the shortest distance from $\hat{\mathbf{r}}$ to $\hat{\mathbf{r}}_i$ on S. Figure 5.6 shows the Voronoi tiling generated by the generator set $\mathcal{G}$ (see Eq. 5.10). We use the software package STRIPACK (Renka 1997) to compute the Voronoi quadrature weights. It is useful to note that

$$\sum_{j=1}^{N_d} w_j = 4\pi \ , \tag{5.22}$$

where, the right hand side corresponds to the solid angle subtended by a unit sphere.

The angle-averaging procedure described above has been implemented on parallel computers. In the following section we give details of the parallel implementation. Some scaling results are provided and performance assessments are made.

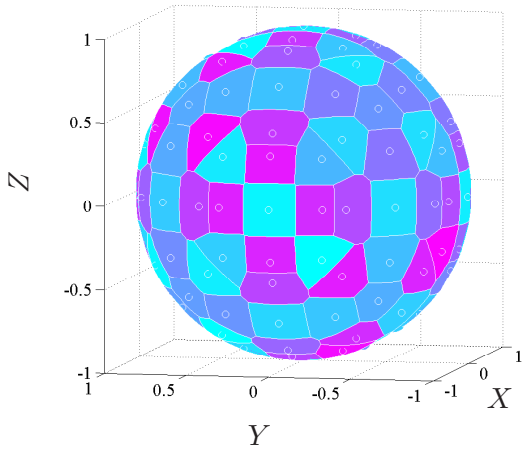

Figure 5.6: Voronoi polygons $V(\hat{\mathbf{r}}_i)$ generated by $\mathcal{G} = \bigcup_{i=1}^{i=N_d} \{\hat{\mathbf{r}}_i\}$ on the unit sphere. Open circles show the generator points $\hat{\mathbf{r}}_i$.

5.3 Parallel algorithm

The structure function calculation for a given spatial separation and direction, $\mathbf{r}_j = r\hat{\mathbf{r}}_j$ can be expressed as

$$\langle [\Delta u_L(r)]^3 \rangle = \langle [u_L(\mathbf{x} + r\hat{\mathbf{r}}_j) - u_L(\mathbf{x})]^3 \rangle \ , \tag{5.23}$$

where $u_L = u_i r_i / r$ is the longitudinal velocity component along $\hat{\mathbf{r}}_j$. For a N^3 grid, there are $N/2$ non-trivial spatial separations along any given direction due to periodicity. This means

that the calculation of Eq. 5.23 requires N^4 flops along a given direction. Furthermore, we want to compute the structure functions along $N_d = 73$ different directions to calculate an angle average (see Eq. 5.15).

As a simplification, we divide the N^3 grid among MPI processes in only one direction. We refer to this decomposition as (MPI) slab/$1D$ decomposition. In order to utilize more cores, we divide the orthogonal direction among OpenMP threads. In theory, this hybrid MPI/OpenMP method allows the use of $N \times k$ cores, where k is the number of cores in a given node. The number of cores in a given node varies typically from 12 to 32 cores. Figure 5.7 shows the mapping of the physical domain into the computational grid. Three different data configuration (a) X-Z slabs (b) X-Y slabs and (c) Y-Z slabs are used in the computations. The data are transposed using MPI_ALLTOALL in between the three configurations. In the hybrid MPI/OpenMP implementation, we use the thread-funneled approach, which means that only the main thread (the thread that initiates the MPI) does the communication. The structure functions in all the $N_d = 73$ directions are calculated in any one of the three slab configurations (Fig. 5.7) as explained in the following.

The slab decomposition makes it easy to calculate structure functions along directions in a given plane since all the required data is within a processor's memory. For instance the direction $(1, 1, 0)$ can readily be computed with data in X-Y slabs, whereas the X-Z configuration is utilized for the $(1, 0, 1)$ direction. Directions that do not lie on a single plane and hence are spread among the various MPI processes require a different strategy.

We give a detailed account of a sample calculation in the $(1, 1, 1)$ direction for a 16^3 box, divided equally among four MPI processes. Since we are using a $1D$ slab decomposition, the domain is divided into four slabs. Consider the data in the X-Z slab configuration. Since we require the longitudinal structure function, we consider the projection of the velocity vector in the $(1, 1, 1)$ direction. Let this quantity be denoted as V, with V_i being the velocity vector projection in the $(1, 1, 1)$ direction at the ith grid point. Figure 5.8 shows the initial setup for this calculation.

For convenience, we refer to spatial separation in terms of integer multiples of the grid-spacing, $\Delta x = L_0/N$. Thus $r = 1$ corresponds to $r/\Delta x = 1$ and so on. We require

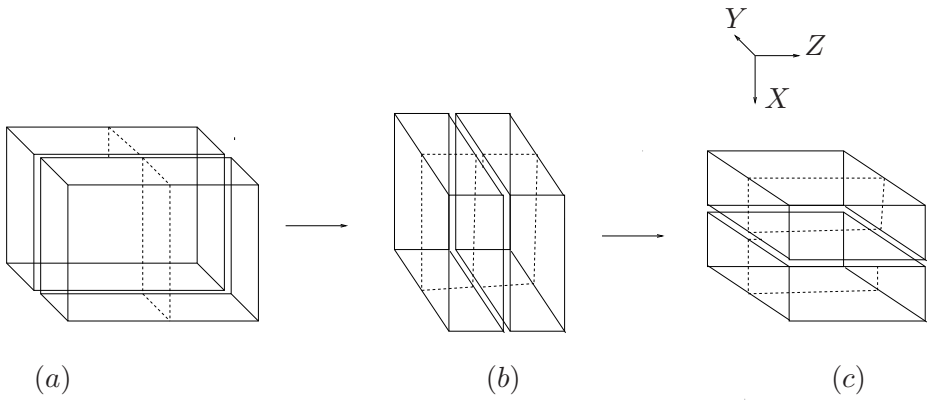

Figure 5.7: Schematic of the hybrid MPI/OpenMP decomposition for a N^3 box for the angle-averaging algorithm. The solid lines denote division among MPI processes, threaded lines represent OpenMP decomposition. (a) X-Z slabs (b) X-Y slabs and (c) Y-Z slabs. Global transposes between slabs are done using MPI_ALLTOALL.

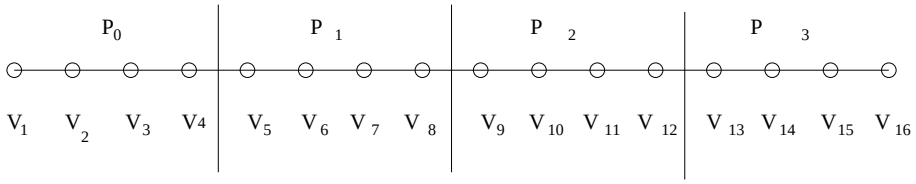

Figure 5.8: Initial setup for the structure function calculation in the $(1,1,1)$ direction for 16^3 grid divided into slabs by four MPI processes, P_0, P_1, P_2 and P_3. Here, V_i is the projection of the velocity vector along $(1,1,1)$ direction at the ith grid point. Note that the distance along adjacent grid points along $(1,1,1)$ is $\sqrt{3}\Delta x$ where $\Delta x = L_0/N$ is the grid spacing in the Cartesian directions.

the right hand side of Eq. 5.23 for $r = 1, 2, 3$ and so on at every point $\mathbf{x}$ in the domain. In the case of $r = 1$, each MPI process only requires the first element in the adjacent process to complete the calculation (see Fig. 5.8). The algorithm consists of three stages for every spatial separation r. In stage-1, we start the data transfer of the first element of each process to its "left neighbor". The initiation of the MPI communication is shown in Fig. 5.9(a). We define the computational grid to be periodic in the direction of the MPI decomposition. This identifies P_3 as the left neighbor of P_0. Thus P_0 sends data to P_3 and P_3 receives this data. The neighboring grid points at the boundaries is shown in Fig. 5.9(a) along with the communication pattern at the boundaries. We use non-blocking persistent MPI communication for the transfers. While the communication proceeds, we perform the computations local to each process as shown in stage-2 of Fig. 5.9. Once we are done with local computations, we complete the communication using MPI_WAIT. Finally we

perform the remaining computation using the transferred data, represented as closed circles in Fig. 5.9. Now, Eq. 5.23 can be computed for $r = 1$ in the $(1, 1, 1)$ direction at every $\mathbf{x}$. This procedure overlaps communication with computation,provided that the number of MPI processes be less than N, the box length.

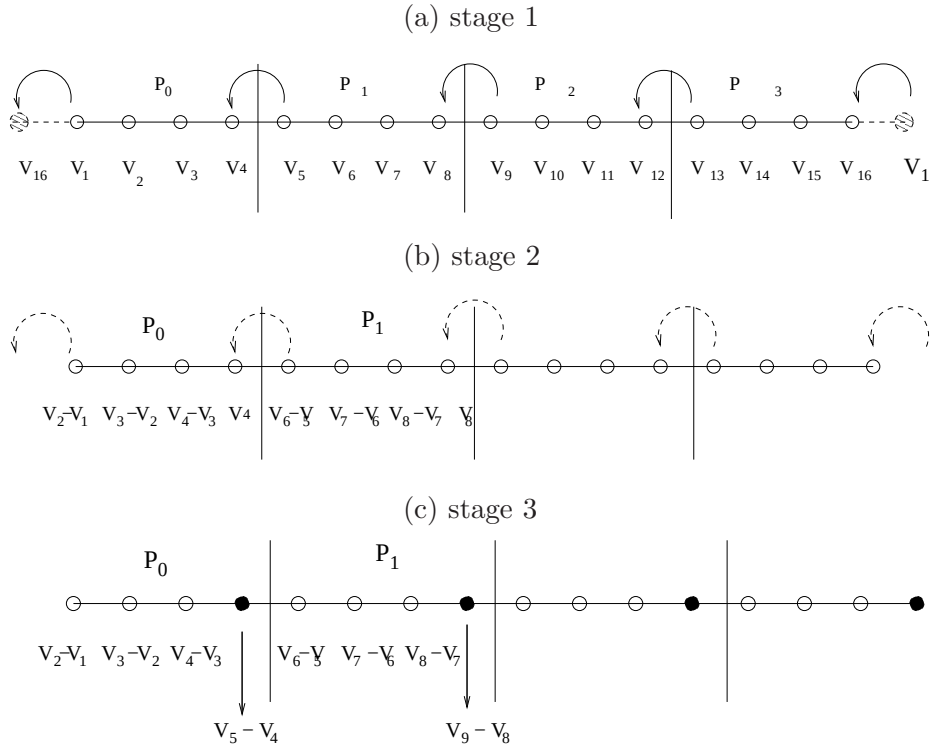

Figure 5.9: Schematic of the different stages in the parallel algorithm for computing Eq. 5.23 for $r = 1$ along $(1, 1, 1)$ direction. (a)(Top) Stage-1: Initiate the MPI transfer using non-blocking calls (b) (Center): Stage-2: perform local computations while the communication is taking place in order to overlap communication and computation, (c) (Bottom): Stage-3: after the communication is completed, compute velocity differences at grid point at the boundaries of each process (shown as solid closed circles).

For the $r = 2$ calculation, we require data points spaced two grid spacings away from each other. From Fig. 5.9(c), we see that the only missing data point for process P_0 is V_6 which is available at the first grid point in process P_1 at the end of $r = 1$ calculation. Similarly, process P_1 requires V_{10} which is available at the first grid point of P_2. Hence, transfer of data from the first grid point of the "right" MPI neighbor to its "left neighbor" is initiated in stage-1 of the $r = 2$ calculation. The rest of the procedure for $r = 2$ is identical to $r = 1$ calculation. Figure 5.10 shows the result at the end of stage-3 for the

$r = 2$, $r = 3$ and $r = 8$ calculations. A essential feature of this scheme is that the data transfer is always from the first grid point of any process to its "left neighbor". Thus each process is receiving data from the first grid point of its "right neighbor" and sending data again from its first grid point to its "left neighbor". This ensures a highly structured and localized communication pattern. Comparing Fig. 5.8 and 5.10 ($r = 8$) case, we see that V_9 has moved from process P_2 to P_0. Thus, information propagates "upstream" in this algorithm, while the communication remains localized in the network.

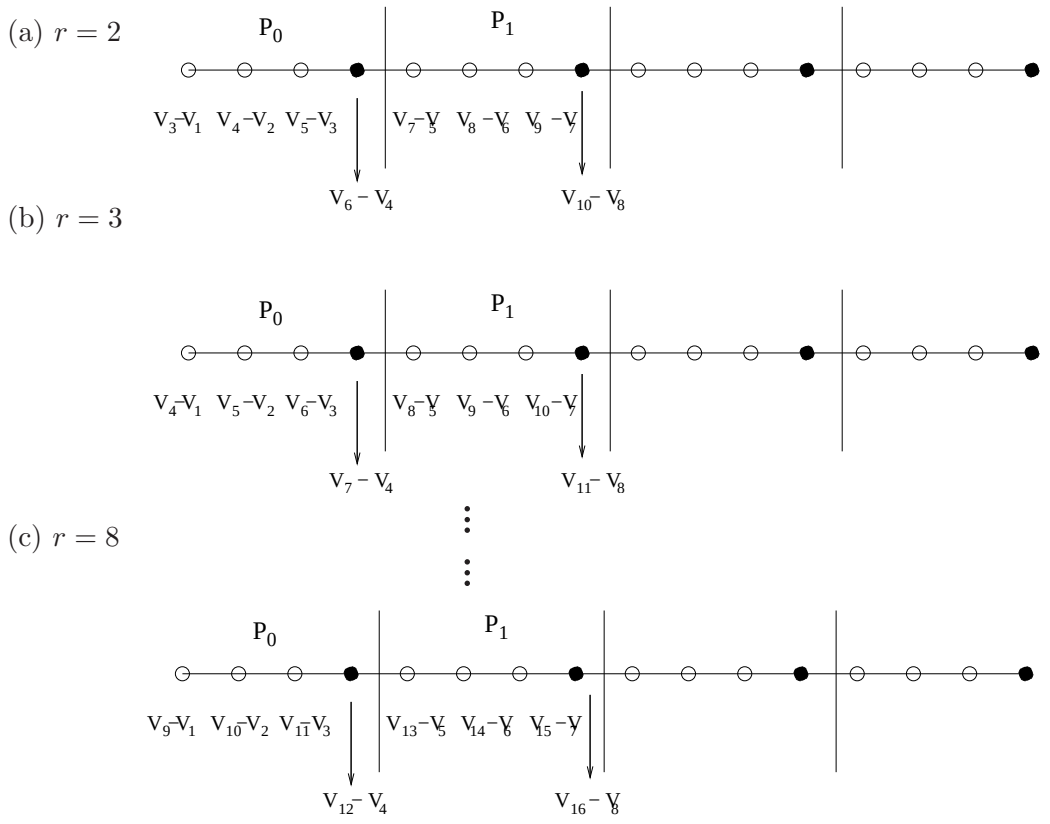

Figure 5.10: Result of the iterative structure function calculation (end of stage-3) for $r = 2, 3, 8$ for 16^3 grid, using four MPI processes. The solid grid points denote the data values calculated as a result of the communication process. Open circles represent grid point values calculated using only local computations. Comparing $r = 8$ case with Fig. 5.8 we see that V_9 which was local to process P_2 is now available in P_0. [13]

A limitation of this iterative approach is that the structure function calculation be carried out at consecutive r. This requires that the calculation be carried out at all $N/2$

spatial separations for every direction, which can be expensive. However, we only compute Eq. 5.23 for $r = 1, 2, \ldots, N_r(j)$ along $\hat{\mathbf{r}}_j$ (Eq. 5.16). This means that the number of spatial separations considered in non-planar directions is lesser than those that lie on a plane. For example, we need to compute Eq. 5.15 along $(3, 1, 1)$ at only one-third of the spatial separations considered for $(1, 0, 0)$ direction.

To efficiently mask the communication time, we perform local computations for multiple directions simultaneously wherever possible. Consider the data in the X-Z slabs configuration. For performing the $(1, 1, 1)$ calculation data is exchanged in the Y direction among the MPI processes as explained above. Now direction $(1, 1, -1)$ requires the same data in the Y-direction as the $(1, 1, 1)$ calculation. Hence we perform these two calculations together.

All the directions of the form $(\alpha, 1, \beta)$ where α,β are non-zero integers can be obtained from the data in X-Z slabs. For directions of the form $(\alpha, \beta, 1)$ we use the X-Y slab configuration, while for $(1, \alpha, \beta)$ we use Y-Z slabs. The data is transposed using MPI_ALLTOALL. Thus two global data transposes are required. The time for the ALLTOALL is insignificant compared to the time required for the structure function calculations.

5.4 Parallel performance

The parallel algorithm uses a 1D slab decomposition to divide the domain among multiple processors. For a N^3 grid, the 1D domain decomposition limits the number of MPI processes to N. This limitation on the number of MPI processors is partly overcome by dividing the orthogonal direction of the domain among OpenMP threads (see Fig. 5.7). Decreasing the MPI processes and increasing the OpenMP threads makes the domain decomposition more coarse grained. Since each process now has more data in the local node memory, the communication time is reduced. Furthermore, using less than N MPI processes allows (at least in principle) a part of the communication to be overlapped with the computation.

Figure 5.11 shows the effect of increasing the number of threads on Stampede (Texas Advanced Computing Center) for a fixed problem size and core count. Each Stampede node contains two Xeon Intel 8-Core 64-bit $E5$-processors (16 cores in all) on a single board, as an symmetric multi-processor unit. Hence, the maximum number of threads allowed is equal

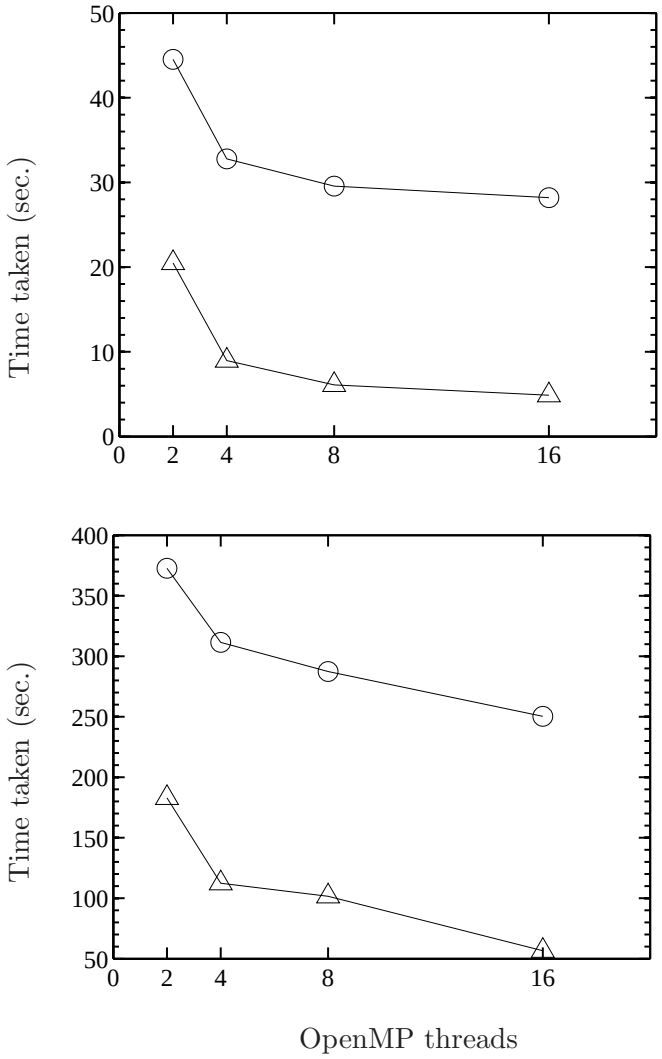

Figure 5.11: Scaling of the angle-averaging code with respect to number of OpenMP threads on Stampede (TACC) for (a) (top) 1024^3 using 2048 cores, (b) (bottom) 2048^3 using 4096 cores. Symbols (△) and (◯) correspond to the communication and total times respectively. The difference between the ordinates of the two curves in each frame gives the computation time which is roughly a constant.

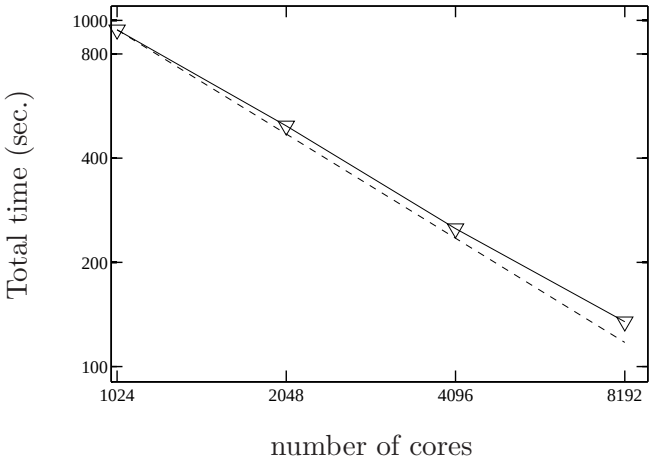

Figure 5.12: Strong scaling for 2048^3 on Stampede (TACC) on logarithmic scales. The number of OpenMP threads used is 16. The number of cores is given by the product of MPI processes and OpenMP threads. Symbol ($\bigtriangledown$) corresponds to total time, while dashed line shows perfect scaling result.

to 16. As the number of threads increase (accompanied by a decrease in MPI processes) for a fixed problem size and core count, the communication time and hence the total time taken decreases. In Fig. 5.11, the vertical distance between the curves corresponding to total and communication times gives the computation time. The computation time is approximately a constant since the workload per core remains the same. The computation time for the 2048^3 case increases slightly as the number of threads is increased (especially when going from 8 to 16 threads). This is possibility because of the penalty incurred when threads migrate across sockets to access data. Regardless, the overall time is the least when 16 threads are used on Stampede. Similar trends have been observed on other machines such as Blue Waters and Titan onto which the code was ported.

Figure 5.12 shows the strong scaling result for 2048^3 box on Stampede. The number of threads was fixed at 16. The strong scaling trend is seen to deviate from the perfect scaling result at larger core counts. This is due to the fact that the number of MPI processes increases with increasing core count for a fixed number of OpenMP threads. The domain decomposition among processes tends to become more fine grained with increase in MPI processes, leading to greater communication time.

5.5 Angle-averaging results

Figure 5.13 compares the angle-averaged third order velocity structure function with the component-averaged version at two different Reynolds numbers. Even at the lower Reynolds number ($R_\lambda \sim 140$) differences are seen between the angle-averaged and component averaged curves. The angle-averaged result appears to have a higher plateau than the component averaged result. At $R_\lambda \sim 650$ the angle-averaged result is seen to converge to the asymptotic $K41$ constant of 4/5. Whereas the component averaged curve has not yet reached the 4/5 plateau.

The structure function results in Fig. 5.13 have been averaged in time using seven instantaneous snap shots. Hence the 4/5 plateau for the 4096^3 case can also be taken as a numerical validation of Eq. 5.4.

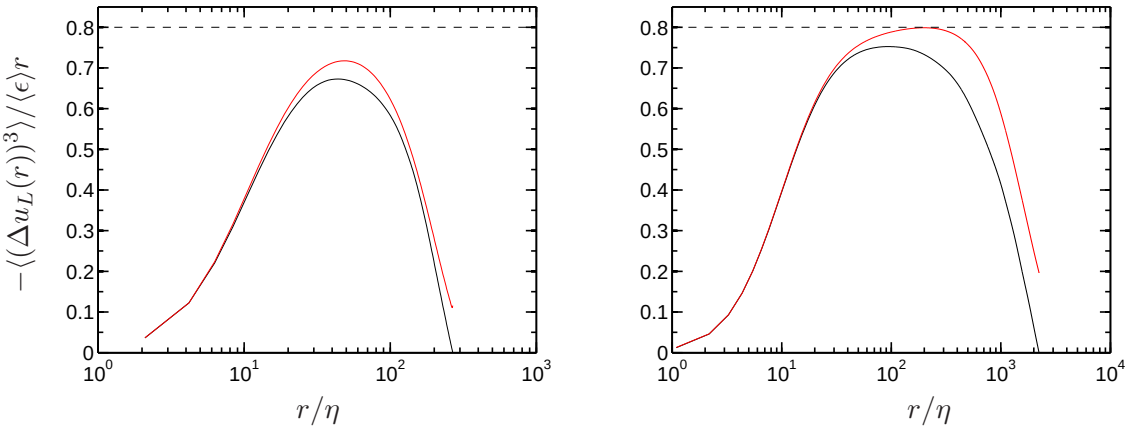

Figure 5.13: Angle-averaged third-order velocity structure function (red), component averaged third-order velocity structure function (black) for (a) (left) 256^3, $R_\lambda \sim 140$, (b) (right) 4096^3, $R_\lambda \sim 650$. Horizontal line at 4/5 shows the asymptotic K41 constant (Eq. 5.1). The structure functions have been averaged in time using seven instantaneous snap shots.

To examine Eq. 5.6 which is the time-local result of Eyink (2003), the maximum instantaneous values of the angle-averaged normalized third order structure function are plotted against time (see Fig. 5.14). The time-axis is normalized by the eddy-turnover time $T_E = l/u'$, where l is the integral scale and u' is the r.m.s velocity fluctuation. In our DNS simulations, $T_E \approx 0.8$ time units. The maximum value of the curves converge well to the constant of 0.8 with very little spread (if any). Velocity field snap shots both close to each

other in time and far apart in time have been checked to converge to the K41 constant. The mean of the data shown in Fig. 5.14 is equal to 0.7998 and its dispersion is almost zero.

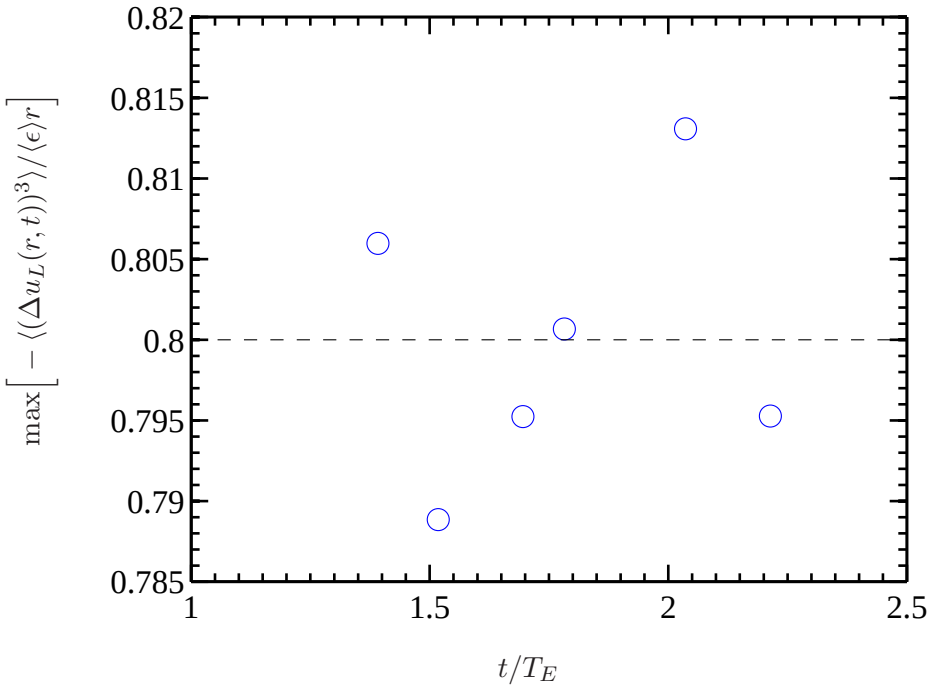

Figure 5.14: The peak of the angle-averaged non-dimensionalized third-order structure function as a function of non-dimensionalized time t/T_E, where $T_E = l/u'$ is the eddy-turnover time, l is the integral scale and u' is the r.m.s velocity fluctuation. The dashed line is the asymptotic constant of 0.8 (Eq. 5.1). The mean of the data is 0.7998 and variance is 7.5×10^{-5}.

CHAPTER VI

LOCAL AVERAGES AND THE REFINED SIMILARITY HYPOTHESIS

6.1 Introduction

A statistical description of the local flow structure in high-Reynolds number turbulence was given by Kolmogorov (1941a). In its simplified version, the first hypothesis relates the probability density function (PDF) of the longitudinal velocity increments $\Delta u(r) = [\mathbf{u}(\mathbf{x}+\mathbf{r}) - \mathbf{u}(\mathbf{x})].\mathbf{r}/|\mathbf{r}|$ to the mean energy dissipation rate $\langle\epsilon\rangle$ and the fluid viscosity (ν), for spatial separations $r = |\mathbf{r}| \ll L$, L being the integral scale of the turbulence. The second hypothesis is that if the Reynolds number is very large, there exists a range of scales (in the so-called inertial range) for which ν becomes irrelevant, so that the PDF of $\Delta u(r)$ depends only on $\langle\epsilon\rangle$, apart from r itself. An implicit assumption in the Kolmogorov (1941a) theory is that the rate of transfer of energy from the large to the small scales is a constant everywhere in the flow and is equal to $\langle\epsilon\rangle$. However, the energy dissipation rate per unit mass of a turbulent fluid is given by

$$\epsilon(\mathbf{x},t) = \frac{\nu}{2}\sum_{i,j}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)^2 , \tag{6.1}$$

which fluctuates in space and time and can be considered as a random function. The fluctuations of $\epsilon(\mathbf{x},t)$ may depend on large-scale properties which can be non-universal (Monin & Yaglom 1975).

Kolmogorov (1962) introduced the refined similarity theory, which introduced more restrictive alternatives and abandoned universality. It postulated that one of the most important factors determining the statistics of $\Delta u(r)$ for $r \ll L$, is the dissipation rate averaged over a local volume $\mathcal{V}(r)$ of linear dimension r, i.e., the quantity

$$\epsilon_r(\mathbf{x},t) = \frac{1}{r^3}\int_{\mathcal{V}(r)} \epsilon(\mathbf{x}+\mathbf{r}',t)d\mathbf{r}' . \tag{6.2}$$

Following Kolmogorov (1962), the quantities r and $\epsilon_r(\mathbf{x},t)$ can be used to construct a velocity scale at the point $(\mathbf{x},t)$ as $U_r = (r\epsilon_r)^{1/3}$, and a local Reynolds number can be formed as

$$Re_r = \frac{U_r r}{\nu} = \frac{(r\epsilon_r)^{1/3} r}{\nu} = \left(\frac{r}{\eta_r}\right)^{4/3}, \qquad \eta_r = \left(\frac{\nu^3}{\epsilon_r}\right)^{1/4}, \tag{6.3}$$

where η_r is the local Kolmogorov length scale. The first refined similarity hypothesis can be stated as the following (Stolovitzky *et al.* 1992), for $r \ll L$,

$$\Delta u(r) = V(r\epsilon_r)^{1/3}, \tag{6.4}$$

where V is a non-dimensionless stochastic variable whose PDF only depends on Re_r.

The second refined hypothesis states that if $Re_r \gg 1$, the PDF of V becomes independent of Re_r, i.e. it is universal.

The third hypothesis is that the PDF of ϵ_r is log-normal with a variance given by (Kolmogorov 1962)

$$\mathrm{Var}(\log \epsilon_r) = A(\mathbf{x},t) + \kappa log(L/r) \tag{6.5}$$

where κ is a universal constant and $A(\mathbf{x},t)$ depends on the flow macrostructure. Thus by knowing the PDF of ϵ_r and V, the PDF (and hence all other statistical moments) of $\Delta u(r)$ can be obtained in the appropriate scale range.

Previous work on this topic (Stolovitzky *et al.* 1992; Wang *et al.* 1996) have examined the refined similarity hypothesis using one-dimensional (1D) averages of the local dissipation rate $\hat{\epsilon_r}(\mathbf{x},t)$, averaged over a line $\mathcal{L}$ of length r as,

$$\hat{\epsilon}_r(\mathbf{x},t) = \frac{1}{r}\int_{\mathcal{L}} \epsilon(\mathbf{x}+\mathbf{r}',t)dr', \tag{6.6}$$

The use of 1D averages over 3D averages in experiments, is partly because of difficulties in obtaining measurements in 3D space, especially for small spatial separations $r \ll L$. Computationally, 1D averages are easier to calculate than $3D$ averages, since the latter require heavy communication between different processors in a parallel network, especially when the spatial separation r is large.

In this chapter a new algorithm to calculate local $3D$ averages is presented. The statistics of $3D$ averaged dissipation are compared and contrasted with those of the 1D averaged

dissipation rate. Significant differences in the scaling laws for the local 1D and 3D averaged dissipation are observed in the intermediate scale ranges. The $3D$ local averages of dissipation are used to examine the first and the second hypotheses of the refined similarity theory. Results for the $K62$ theory is presented for the 4096^3 simulation at $R_\lambda \sim 650$.

6.2 Numerical procedure

The local average of a scalar quantity such as the energy dissipation rate (ϵ) over a volume, $\mathcal{V} \sim O(r^3)$ is defined as

$$\epsilon_r(\mathbf{x}, t) = \frac{1}{r^3} \int_{\mathcal{V}} \epsilon(\mathbf{x} + \mathbf{r}, t) d\mathbf{r} \ . \tag{6.7}$$

A sample of ϵ_r is calculated at the point (x_1, y_1, z_1) over a cube with edge length of r grid spacings using the formula,

$$\epsilon_r(x_1, y_1, z_1) = \frac{1}{(r+1)^3} \sum_{z=z_1}^{z_{r+1}} \sum_{y=y_1}^{y_{r+1}} \sum_{x=x_1}^{x_{r+1}} \epsilon(x, y, z) \ . \tag{6.8}$$

In order to obtain adequately converged statistics of ϵ_r, Eq. 6.8 needs to be calculated at every point (x, y, z) in a N^3 grid. The examination of the statistics of ϵ_r at various scale sizes, requires Eq. 6.8 to be calculated at all non-trivial spatial separations, which spans $r = 1, 2, \ldots, N/2$ grid spacings. To save computer time, we typically compute Eq. 6.8 only at selected multiples of grid spacing, at the larger spatial separations. In spite of this simplification, the computational complexity of Eq. 6.8 can be taken as $\sim O(N^2)$ flops. Added to the computational complexity, the domain decomposition in a parallel code, requires significant inter-processor communication, especially when the averaging length r is large. The non-linear dependence of the computational complexity on N and the inter-processor communication required at every averaging length, makes the problem of computing local averages challenging.

The right-hand-side of Eq. 6.8 is calculated using three one-dimensional (1D) prefix sums (Blelloch 1990) for each spatial separation r, expressed as a multiple of the grid spacing. The algorithm that has been developed for computing Eq. 6.8 is referred to as the "$3D$ local averages" algorithm. The $3D$ local averages algorithm uses a two-dimensional (2D) domain decomposition (Donzis *et al.* 2008*a*) to map the 3D physical domain onto a 2D

computational grid. The use of a 2D computational grid allows the use of a maximum of N^2 processors in theory, for a N^3 domain with N points along each direction. The main advantages of $2D$ computational grids over 1D grids is that they decrease memory required per processor by enabling the use of higher core counts. A schematic of the mapping of the N^3 domain onto a 2D processor grid with $P = P_{row} \times P_{col}$ MPI processes is shown in Fig. 6.1.

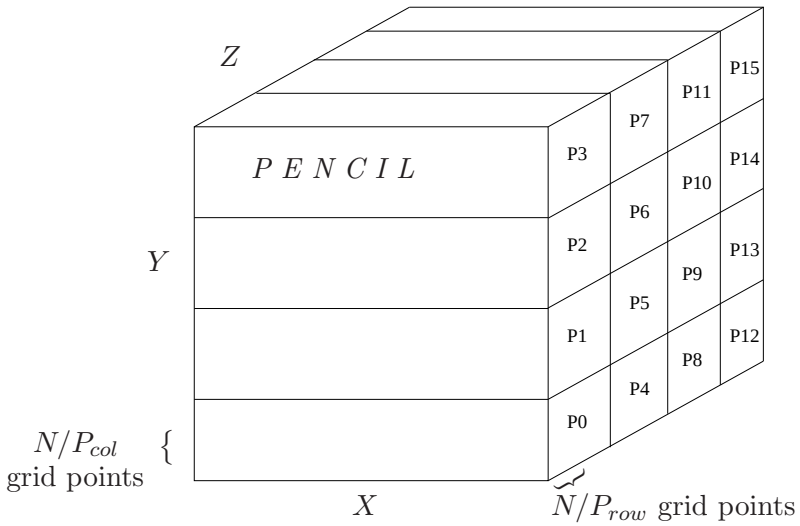

Figure 6.1: Mapping the N^3 domain into a 2D computational grid with $P = P_{row} \times P_{col}$ processors for the 3D prefix sum algorithm. Here $P_{row} = P_{col} = 4$.

The 3D local averages algorithm computes local sums for a given spatial separation along the three orthogonal directions, in successive fashion to calculate the right-hand-side of Eq. 6.8. In what follows, a 1D local sum calculation along the X-direction (see Fig. 6.1) which is local to each processor's memory, is presented. Apart from being used in the 3D local averages algorithm, the 1D local sum calculation is used to calculate 1D local averages. The extension of the 1D case to the 3D local averages algorithm is then discussed.

Consider a line segment of length (L_0) equally divided among N points (say) in the X direction. Assume, all the N points are contained within a single processor's memory. Let the value of a scalar (denoted by ϵ) be ϵ_k at the k^{th} grid point. Periodicity is assumed along the X-direction with a period of L_0, that is $\epsilon_{L_0+k} = \epsilon_k$ for $1 \leq k \leq N$. Figure 6.2 (top) shows the setup for the $N = 8$ case. The prefix sum σ_k at the k^{th} grid point is calculated

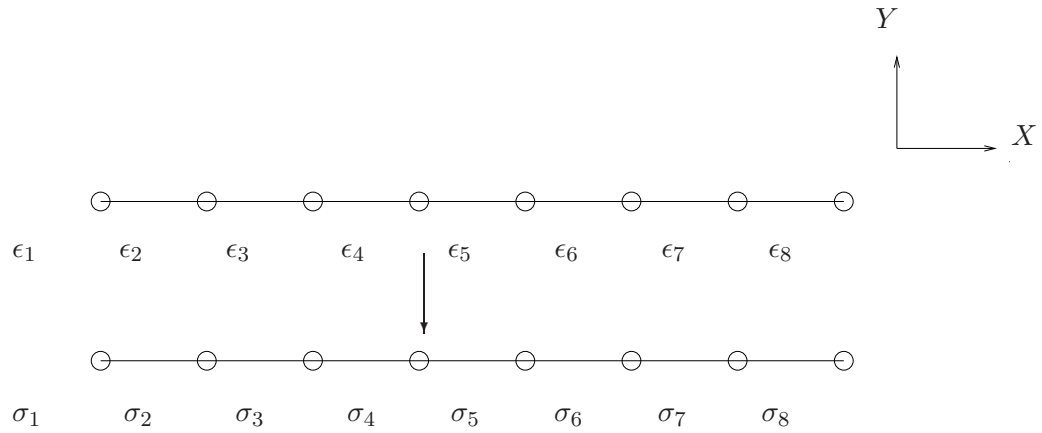

Figure 6.2: Schematic for the 1D local sums algorithm for $N = 8$ case. Here ϵ_k denotes the value of a scalar quantity at the k^{th} grid point, while σ_k denotes its prefix sum (see Eqs. 6.9 and 6.10). The former can be overwritten by the latter using Eqs. 6.9 and 6.10.

as

$$\sigma_1 = \epsilon_1 \tag{6.9}$$

$$\sigma_k = \sigma_{k-1} + \epsilon_k \ , \quad 2 \leq k \leq N \ . \tag{6.10}$$

The result is shown in Fig. 6.2 (bottom) for the $N = 8$ case. The computational complexity of calculating Eq. 6.9 and 6.10 is $O(N)$. The 1D local sum for separation distance of r grid spacings, denoted as $\Sigma_r(k)$ at the k^{th} grid point ($k = 1, 2, \ldots, N$) is then evaluated according to the following formula:

$$\Sigma_r(k) = \begin{cases} \sigma_{r+1} & \text{if } k = 1 \\ \sigma_{k+r} - \sigma_{k-1} & \text{if } 2 \leq k \leq N - r \\ \sigma_N + \sigma_{N-k-r} - \sigma_{k-1} & \text{if } N - r < k \leq N \end{cases} \tag{6.11}$$

In the above formulae, the last case of $N - r < k \leq N$ accounts for the periodicity along the X-direction. The computation of Eq. 6.11 in the single processor case considered here is trivial since at the most only two or three data points are required for a given spatial separation. Hence the computational complexity of the local 1D sum calculation is $\sim O(N)$, i.e. an $O(N^2)$ problem is now reduced to an $O(N)$ problem with the use of prefix sums. For the case when the N points are distributed among multiple processors, inter-processor communication may be required.

The 3D local averages algorithm uses the procedure explained above to calculate three successive 1D sums at each of the N^3 grid points for a given spatial separation r. The data are contained in X-pencils in each processor. The corresponding $2D$ computational grid is shown is Fig 6.3. Two orthogonal sub-communicators are created using MPI_COMM_CREATE since each process is only going to be communicating with processes in the same column or same row. For the case with $P_{row} = P_{col} = 4$ shown in Fig. 6.3, there are four non-overlapping column communicators and four non-overlapping row communicators.

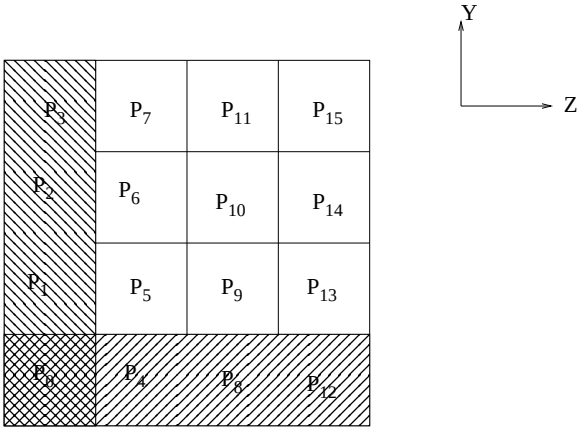

Figure 6.3: Mapping the N^3 domain into a 2D computational grid with $P = P_{row} \times P_{col}$ processors for the 3D prefix sum algorithm. Here $P_{row} = P_{col} = 4$. MPI_COMM_CREATE is used to divide MPI_COMM_WORLD into orthogonal sub-communicators. Processes P_0, P_1, P_2, P_3 form a column communicator; processes P_0, P_4, P_8, P_12 form a row communicator. The communication in the $3D$ local average algorithm is restricted to these sub-communicators.

The first local sum calculation is carried out in the Y-direction (see Fig. 6.3). The local prefix sums are calculated using Eqs. 6.9 and 6.10. Then, MPI_SCAN (Gropp *et al.* 1999) is used to form the global prefix sums. If $\epsilon_{p_0}, \epsilon_{p_1}$ etc. are the local prefix sums in the processes P_0, P_1 etc. which belong to the same column communicator (see Fig. 6.3), then an inclusive SCAN results in ϵ_{p_0} in the process P_0, $\epsilon_{p_0} + \epsilon_{p_1}$ in process P_1 etc. The SCAN is a blocking collective operation provided by MPI and works similarly as the reduction collectives (reduce/all-reduce etc.). Thus the result from the scan operation is used to form the global prefix sums along the Y direction.

The data are now in the form of σ_k (see Eq. 6.9,6.10 and Fig. 6.2), where $k = 1, 2, \ldots, N$ is a global index. Now Eq. 6.11 is used to calculate the 1D local sums in the Y direction. Since the data in Y direction are spread among P_{col} processes, a given processor may need to communicate with another processor in its column sub-communicator, to access the appropriate σ_k. For a given r, each process needs to communicate with at the most three other processes to access the required global sums. Whether the point that contains σ_{k+r} or σ_{N-k-r} etc that is required by point k (as per Eq. 6.11) is contained in a different pencil or resides in the same pencil as point k, depends on the values of r, N and the grid parameters P_{row} and P_{col}. Since for every point in the domain and for every spatial separation r, this relationship is fixed, we ascertain if global points $k + r$, $N - k - r$ etc. are contained in the same pencil as point k for every point k in the domain ($k = 1, 2, \ldots, N$ in the Y-direction) as a one-time initialization step. This one-time initialization procedure is not repeated again. Then, if a inter-processor communication is required, an MPI_SENDRECV is used to complete the communication. If the other point resides in the same pencil, then a MPI_PROC_NULL message is passed which means that the MPI call return immediately with no action. In this way, using Eq. 6.11 $\Sigma_r(k)$ for every point k is calculated in the Y-direction at a given spatial separation.

Now that the 1D local sums in the Y direction is calculated, we do a local 1D sum in the X-direction using Eqs. 6.9, 6.10 and 6.11 as explained above. Since all the data required, is contained within a processor's memory, no communication is required.

After the $2D$ sums are formed by summing in the Y and X direction, the final summation along the Z direction is performed. The 1D local sum in the Z-direction is carried out in the row communicator (see Fig. 6.3). As with the column communicator, a one-time initialization step to determine which k (now in the Z direction) requires which other points to form the 1D local sums is done at the beginning. This information is used to determine which processes need to communicate with each other. If the two points belong to the same process then the process sends a MPI_PROC_NULL message and returns immediately. This completes the $3D$ local sums for the given spatial separation of r grid spacings.

Finally the $3D$ local average for the averaging volume $\sim O(r^3)$ is calculated using Eq. 6.8.

The same process is repeated for every spatial separation for which the local averages are required. The advantage of this method is that the calculations at a given spatial separation is independent of results from any other spatial separation. This fact is used to skip over many spatial separations in the large-r regime to save computer time.

6.3 Local 1D and local 3D averages of dissipation

Consider the stochastic variable V defined as (Kolmogorov 1962)

$$V(r) = \frac{\Delta u(r)}{(r\epsilon_r)^{1/3}} \,, \tag{6.12}$$

where $\Delta u(r) = [\mathbf{u}(\mathbf{x}+\mathbf{r}) - \mathbf{u}(\mathbf{x})] \cdot \mathbf{r}/|\mathbf{r}|$ is the longitudinal velocity increment along separation vector $\mathbf{r}$ and ϵ_r is the three-dimensional (3D) local average of the dissipation rate over a volume $O(r^3)$. The 3D local dissipation rate $\epsilon_r(\mathbf{x}, t)$ is defined over a volume $\sim O(r^3)$ as

$$\epsilon_r(\mathbf{x}, t) = \frac{1}{r^3} \int_{\mathcal{V}} \epsilon(\mathbf{x} + \mathbf{r}', t) d\mathbf{r}' \,, \tag{6.13}$$

where $\mathcal{V}$ is the volume of integration. In Cartesian coordinates it is convenient to consider a cube of edge length r grid spacings as the volume over which dissipation is locally averaged. Figure 6.4 shows a sample cube of edge length r grid spacings that is considered in this work.

There is considerable ambiguity in the precise definition of V when 3D averages are considered. Since a cube such as that in Fig. 6.4 has twelve edges over which a longitudinal velocity increments can be defined, we can have twelve different samples of variable V. Presumably in isotropic homogeneous turbulence the statistics of V over these twelve directions can be considered as different samples which can then be averaged.

In this work we calculate $V(r)$ along the three edges marked in red in Fig. 6.4 as

$$V_1(r) = \frac{u_1^D - u_1^A}{(r\epsilon_r)^{1/3}} \,, \tag{6.14}$$

$$V_2(r) = \frac{u_2^E - u_2^A}{(r\epsilon_r)^{1/3}} \,, \tag{6.15}$$

$$V_3(r) = \frac{u_3^B - u_3^A}{(r\epsilon_r)^{1/3}} \,, \tag{6.16}$$

where (u_1^A, u_2^A, u_3^A) represent the Cartesian velocity components at point A. The statistics of V_α $(\alpha = 1, 2, 3)$ can be considered as different realizations of the statistics of V and can

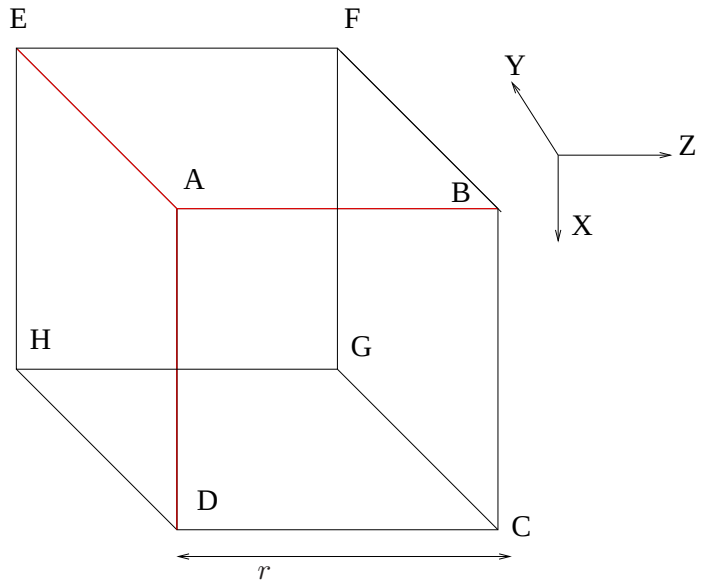

Figure 6.4: Cube of edge length equal to r grid spacings. Red lines show edges along which V (see Eqs. 6.12, 6.14, 6.15 and 6.16) is calculated.

be averaged accordingly. For example, the third moment of V is calculated as

$$\langle V^3 \rangle = \frac{1}{3}[\langle V_1^3 \rangle + \langle V_2^3 \rangle + \langle V_3^3 \rangle] \,, \tag{6.17}$$

where $\langle \cdot \rangle$ denote space averages over the entire domain.

As a preliminary step, some statistical information on the single point dissipation and the corresponding 3D and 1D local averages of dissipation is presented. The 1D local average of dissipation rate $\hat{\epsilon_r}$ is defined as

$$\hat{\epsilon}_r(\mathbf{x}, t) = \frac{1}{L} \int_L \epsilon(\mathbf{x} + \mathbf{r}', t) dr'. \tag{6.18}$$

In the small-r limit, we have

$$\lim_{r \to 0} \frac{\langle {\epsilon_r}^q \rangle}{\langle \epsilon_r \rangle^q} = \frac{\langle \epsilon^q \rangle}{\langle \epsilon \rangle^q} \,, \tag{6.19}$$

$$\lim_{r \to 0} \frac{\langle \hat{\epsilon}_r^q \rangle}{\langle \hat{\epsilon}_r \rangle^q} = \frac{\langle \epsilon^q \rangle}{\langle \epsilon \rangle^q} \,, \tag{6.20}$$

where $q = 1, 2, 3, \ldots$. At large spatial separations, we can write

$$\lim_{r \to \infty} \frac{\langle {\epsilon_r}^q \rangle}{\langle \epsilon_r \rangle^q} = 1 \,, \tag{6.21}$$

$$\lim_{r \to \infty} \frac{\langle \hat{\epsilon}_r^q \rangle}{\langle \hat{\epsilon}_r \rangle^q} = 1 \,. \tag{6.22}$$

Figure 6.5 shows the second moments of local 3D and 1D averaged dissipation as a function of spatial separation for two different resolutions at the same Reynolds numbers. It is clear that the 3D local averages are more resolution limited than the 1D averages. The intermediate slope range for the curves is different. The 1D averaged dissipation moments are greater than the 3D averaged dissipation moments at a given spatial separation. This is because 3D averages encompass more spatial information than the 1D averages, and are hence likely to be smaller. Whereas the 1D averages are closer to the point-wise dissipation and hence have better resolution at small separations. At the largest spatial separations, 3D averages satisfy Eq. 6.21 while the 1D case does not satisfy Eq. 6.22. This is because, for any order (greater than 1), 3D moments have a smaller dispersion than the corresponding 1D moments due to increased statistical stability. Periodic boundary conditions seem to have a decreased effect on 3D averages at large r due to increased spatial sampling.

Figure 6.6 shows the higher order moments for the dissipation. As the order of the moment increases, the resolution requirements for the 3D case seem to be more stricter than that for the 1D case. Consider local averages over a volume of characteristic length Δx, which is the grid spacing. A sample of the 1D average represents the dissipation at the midpoint of an edge in the cube. Whereas the 3D average represents the dissipation at the centroid of the cube, which is $\sqrt{3}$ times further away from the given grid point than the corresponding 1D average. Hence 1D averages are closer (spatially) to the point-wise averages than their $3D$ counterparts and hence are numerically closer. The moments of $3D$ averaged dissipation are smaller than that of 1D averaged dissipation, indicating that ϵ_r may be less intermittent than $\hat{\epsilon}_r$. For a given averaging length r, we can write

$$\langle {\epsilon_r}^q \rangle \leq \langle \hat{\epsilon}_r^q \rangle \leq \langle \epsilon^q \rangle \ , \quad q = 1, 2, 3, \ldots , \tag{6.23}$$

equality occurring when $q = 1$.

Figure 6.7 shows the probability density function for the single-point, 3D and 1D averaged dissipation at two different resolutions at the same Reynolds number. The tails of the PDF get wider with increased resolution as more scales in the large wavenumber regime contribute to the dissipation spectrum. The PDF of $\hat{\epsilon}_r$ has longer tails than that of ϵ_r (see

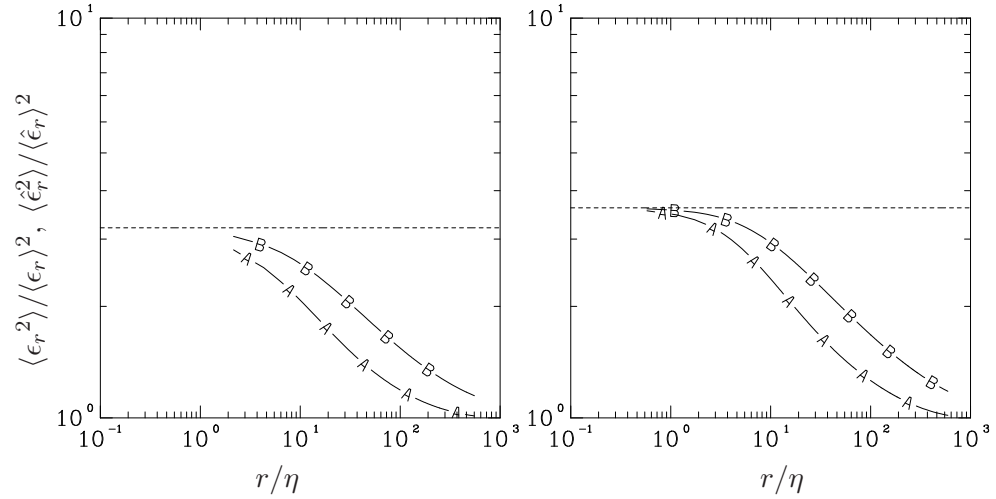

Figure 6.5: Second order moments of 3D local averages (line A) and 1D local averages (line B) of dissipation for (a) (left) $k_{max}\eta = 1.4$, (b) $k_{max}\eta = 5.7$, both at $R_\lambda \sim 240$. Dashed lines indicate second order moment of point-wise dissipation ($\langle \epsilon^2 \rangle / \langle \epsilon \rangle^2$).

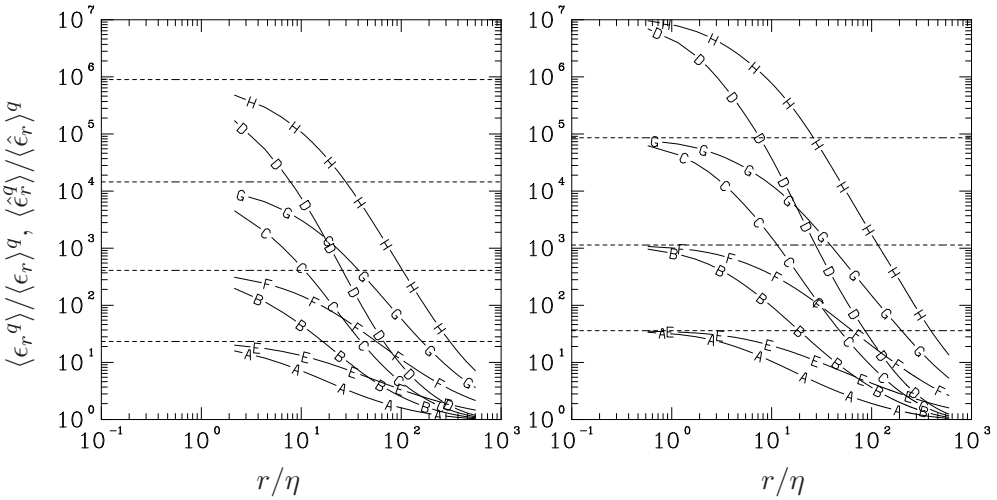

Figure 6.6: Higher order moments of 3D local averages (lines A-D correspond to $q = 3, 4, 5, 6$) and 1D local averages (lines E-H correspond to $q = 3, 4, 5, 6$) of dissipation for (a) (left) $k_{max}\eta = 1.4$, (b) $k_{max}\eta = 5.7$, both at $R_\lambda \sim 240$. Dashed lines indicate corresponding moment of point-wise dissipation ($\langle \epsilon^q \rangle / \langle \epsilon \rangle^q$, $q = 3, 4, 5, 6$).

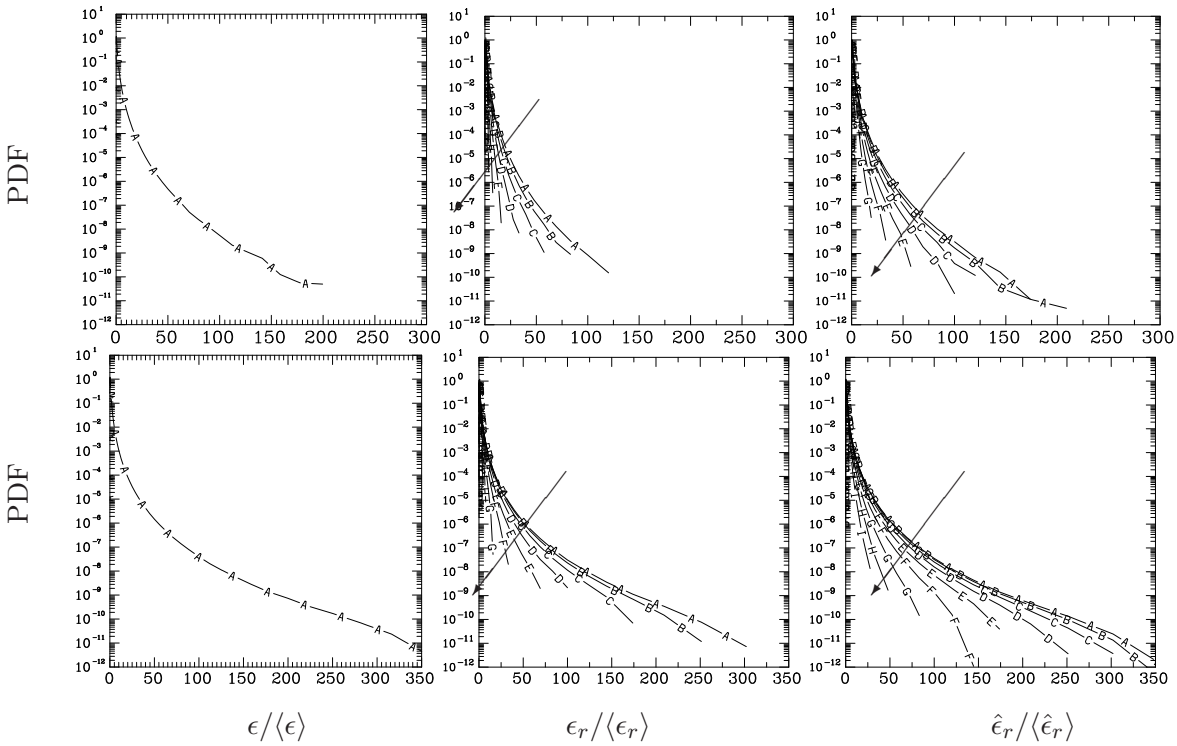

Figure 6.7: PDF of dissipation rate at $R_\lambda \sim 240$ for (left) single-point, (center) local 3D averages, (right) local 1D averages. Plots on the top row correspond to $k_{max}\eta = 1.4$, while bottom row is for $k_{max}\eta = 5.7$. Curves A, B, C etc. correspond to different averaging lengths $r/\Delta x = 1, 2, 4, \ldots$, increasing in the direction shown.

Fig. 6.7), which indicates that the likelihood of the 1D dissipation taking extreme values is greater than that of $3D$ dissipation. This is consistent with the trends for the absolute moments of dissipation shown in Fig. 6.6 and Eq. 6.23.

6.4 K62 related statistics

Following Stolovitzky *et al.* (1992), the correlation coefficient between velocity increments and $(r\epsilon_r)^{1/3}$ is plotted in Fig. 6.8 (line A). The two quantities appear poorly correlated at all r except possibly at largest r (in the $k_{max}\eta_B \approx 5.7$ case). The vanishing correlation between Δu and $(r\epsilon_r)^{1/3}$ is a consequence of homogeneity. At large r there seems to be some variation which may be due to effects of forcing and/or decreased sampling due to finite box considerations. Furthermore, as a means to save computer time, the computations at some spatial separations in the large-r regime are not performed. Neglecting some spatial separations at large-r might also be adding to the variability in this scale range.

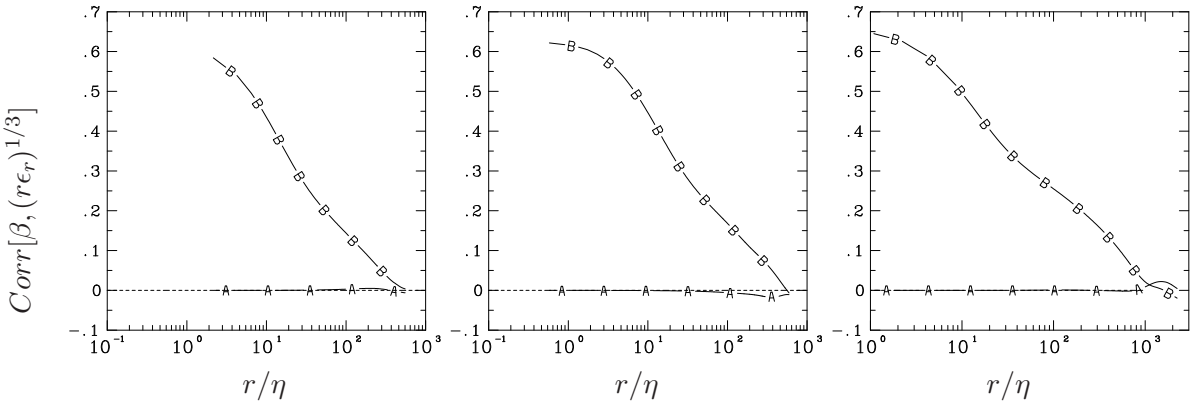

Figure 6.8: Correlation coefficient as a function of spatial separation between $(r\epsilon_r)^{1/3}$ and the quantity β where β is (a) $\Delta u(r)$ (line A) and (b) $|\Delta u(r)|$ (line B) for (left) $R_\lambda \sim 240$, $k_{max}\eta = 1.4$, (center) $R_\lambda \sim 240$, $k_{max}\eta = 5.7$, and (right) $R_\lambda \sim 650$, $k_{max}\eta = 2.7$

Line B shows that the correlation coefficient between $|\Delta u(r)|$ and $(r\epsilon_r)^{1/3}$ is stronger at the small and intermediate scale range. For both these plots, the correlation along the three Cartesian directions, that is the correlation between $\Delta u_1(r)$ and $(r\epsilon_r)^{1/3}$, $\Delta u_2(r)$ and $(r\epsilon_r)^{1/3}$ and $\Delta u_3(r)$ and , $(r\epsilon_r)^{1/3}$ are calculated. The component averaged results are then calculated by averaging over the three Cartesian directions. The correlation coefficients between $|\Delta u(r)|$ and $(r\epsilon_r)^{1/3}$ are calculated in a similar manner.

To investigate the effects of different choices of $\Delta u(r)$ that are possible in a cube (see Eqs. 6.14, 6.15 and 6.16), the correlation between $(|\Delta u(r)|, (r\epsilon_r)^{1/3})$ are plotted in the three orthogonal directions in Fig. 6.9. The correlations along X, Y and Z direction are similar to each other as can be expected in isotropic turbulence. The correlations are also in qualitative agreement with that in Fig. 2 of Stolovitzky *et al.* (1992), accounting for the fact that the Reynolds number in the present case in smaller than that in Stolovitzky *et al.* (1992).

Next the correlation coefficients of $V(r)$ (Eq. 6.12) with $(r\epsilon_r)^{1/3}$ are shown in Fig. 6.10 (line A). The correlation between V and $(r\epsilon_r)^{1/3}$ is almost zero at all spatial separations (except possibly at largest r). In contrast, $|V(r)|$ and $(r\epsilon_r)^{1/3}$ are positively (although weakly) correlated at $R_\lambda \sim 240$.

Figure 6.11 shows the correlation of $|V|$ with $(r\epsilon_r)^{1/3}$ along the three Cartesian directions. The correlation in the X, Y, Z directions are close to each other as can be expected

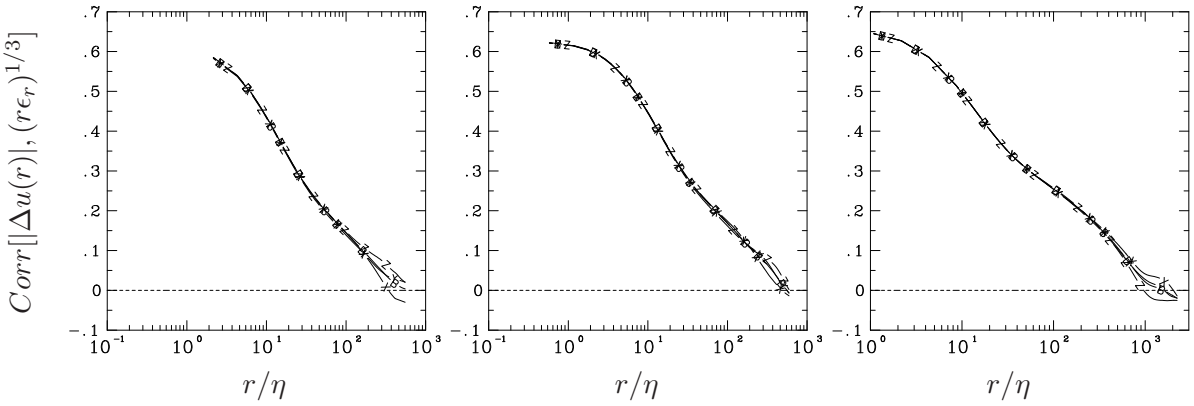

Figure 6.9: Correlation coefficient between $|\Delta u(r)|$ and $(r\epsilon_r)^{1/3}$ along three Cartesian directions (lines X,Y and Z) as a function of spatial separation for (left) $R_\lambda \sim 240$, $k_{max}\eta = 1.4$, (center) $R_\lambda \sim 240$, $k_{max}\eta = 5.7$, and (right) $R_\lambda \sim 650$, $k_{max}\eta = 2.7$. The component average is given by line B which is the same as line B in Fig. 6.8.

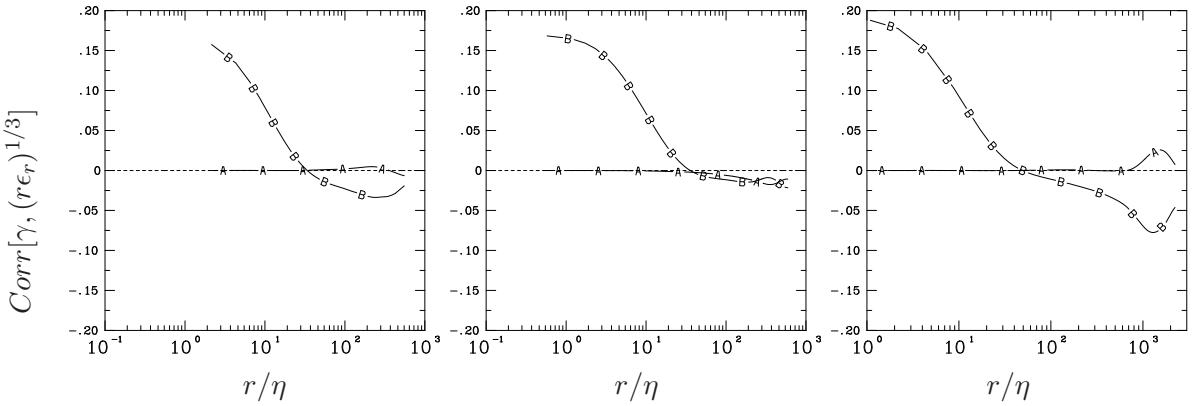

Figure 6.10: Correlation coefficient as a function of spatial separation between $(r\epsilon_r)^{1/3}$ and the quantity γ where γ is (a) $V(r)$ (line A) and (b) $|V(r)|$ (line B) for (left) $R_\lambda \sim 240$, $k_{max}\eta = 1.4$, (center) $R_\lambda \sim 240$, $k_{max}\eta = 5.7$, and (right) $R_\lambda \sim 650$, $k_{max}\eta = 2.7$

in isotropic turbulence.

Next, the third moment of $V(r)$ (see Eq. 6.12) is considered. Raising Eq. 6.12 to third power and averaging, we get

$$\langle (V(r))^3 \rangle = \left\langle \frac{(\Delta u(r))^3}{r\epsilon_r} \right\rangle \tag{6.24}$$

In the large-r limit, $\langle (V(r))^3 \rangle$ tends to zero. In the small r limit, Taylor series arguments can be used for $r \leq \eta$, to write,

$$\langle (V(r))^3 \rangle \approx \left\langle \frac{(\partial u_\alpha / \partial x_\alpha)^3}{\epsilon} \right\rangle r^2 \tag{6.25}$$

where $\partial u_\alpha / \partial x_\alpha$ is a longitudinal velocity component derivative ($\alpha = 1, 2, 3$), with no summation implied and ϵ is the point-wise dissipation. Figure 6.12 shows the variation of

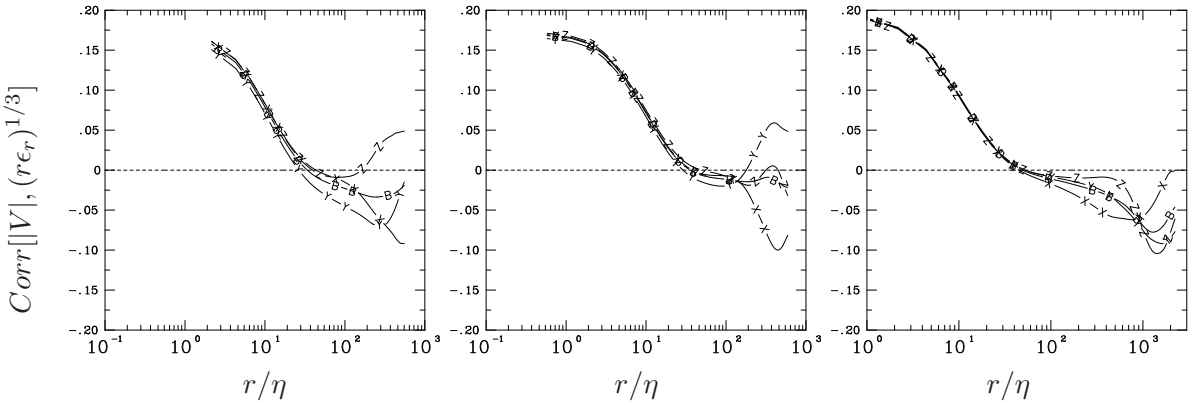

Figure 6.11: Correlation coefficient between $|V|$ and $(r\epsilon_r)^{1/3}$ along three Cartesian directions (lines X,Y and Z) as a function of spatial separation for (left) $R_\lambda \sim 240$, $k_{max}\eta = 1.4$, (center) $R_\lambda \sim 240$, $k_{max}\eta = 5.7$, and (right) $R_\lambda \sim 650$, $k_{max}\eta = 2.7$. The component average is given by line B which is the same as line B in Fig. 6.10.

$\langle (V(r))^3 \rangle$ with spatial separation on logarithmic axes. The dashed lines represent a r^2 behavior to check the small-r asymptote. All the four curves corresponding to the three Cartesian directions and the resulting component average satisfy Eq. 6.25 even for the lower resolution case. As expected, the small-r behavior is more pronounced for the higher resolution case. The third moment of $V(r)$ are plotted on linear scales to show the intermediate

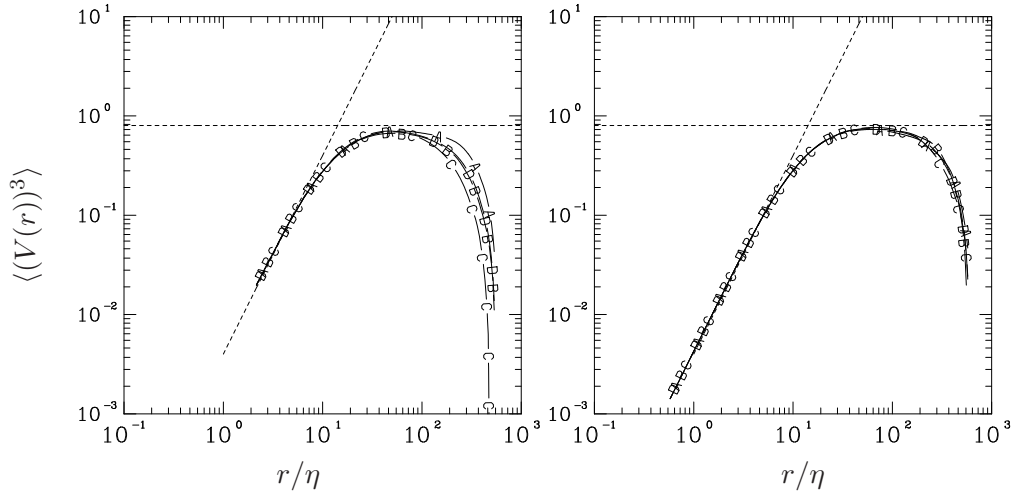

Figure 6.12: Third moment of V along three different Cartesian directions X,Y and Z (lines A,B and C) along with the component averaged plot (line D) at $R_\lambda \sim 240$ for (a) (left) $k_{max}\eta = 1.4$, (b) $k_{max}\eta = 5.7$. Dashed lines (positive slope) to check the small-r slope (see Eq. 6.25) while dashed line (zero slope) at 0.8 shows the $K41$ plateau.

range behavior. Figure 6.13 shows the variation of $\langle (V(r))^3 \rangle$ with spatial separation. The third moment $\langle (V(r))^3 \rangle$ is expected to reach the $K41$ plateau of 4/5 at a sufficiently high

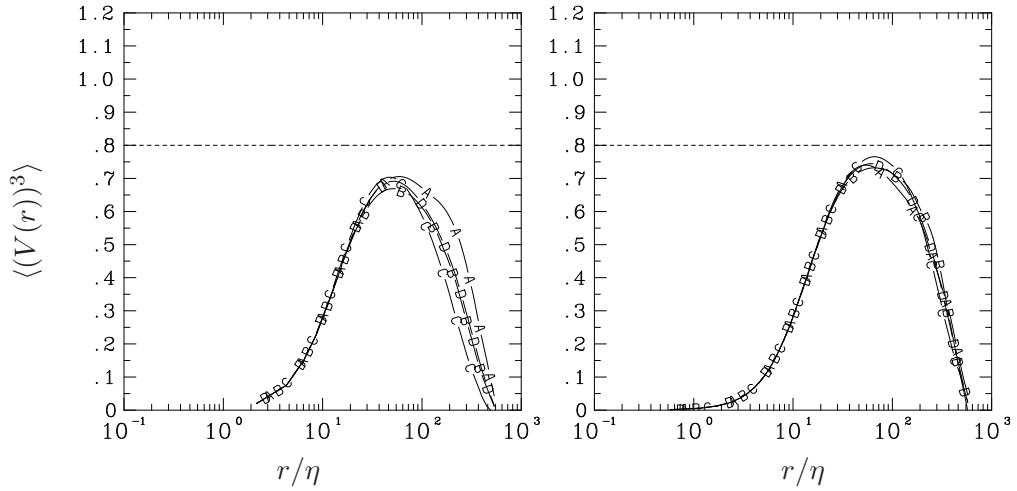

Figure 6.13: Third moment of V along three different Cartesian directions X,Y and Z (lines A,B and C) along with the component averaged plot (line D) at $R_\lambda \sim 240$ for (a) (left) $k_{max}\eta = 1.4$, (b) $k_{max}\eta = 5.7$. Dashed lines (positive slope) to check the small-r slope (see Eq. 6.25) while dashed line (zero slope) at 0.8 shows the $K41$ plateau.

Reynolds number (Stolovitzky *et al.* 1992).

6.5 K62 results

In the following, the first and the second postulates of the refined similarity theory (Kolmogorov 1962) are examined at a Reynolds number $R_\lambda \sim 650$, 4096^3. This is currently the highest Reynolds number in our DNS database at the resolution of $k_{max}\eta \sim 2.7$. As a prelude to the $K62$ results some statistical information on the $3D$ averaged dissipation for the $R_\lambda \sim 650$ is provided.

Figure 6.14 shows the moments of $3D$ averaged dissipation rate. The results are qualitatively similar to the $3D$ dissipation results for $R_\lambda \sim 240$ case (Figs. 6.5 and 6.6), noting that the resolution for the 4096^3 case is midway between those for $R_\lambda \sim 240$ cases. Figure 6.15 shows the PDF of point-wise dissipation and $3D$ averaged dissipation at $R_\lambda \sim 650$, with a focus on the tail behavior. At such a high Reynolds number, the dissipation is as high as 10000 times its mean value. The tail behavior of the $3D$ averaged dissipation is similar to that of the point-wise dissipation even when the averaging length is over four times the Kolmogorov length scale.

Figure 6.16 shows the normalized third-order longitudinal velocity structure function at $R_\lambda \sim 650$. The structure functions along the three Cartesian directions along with their

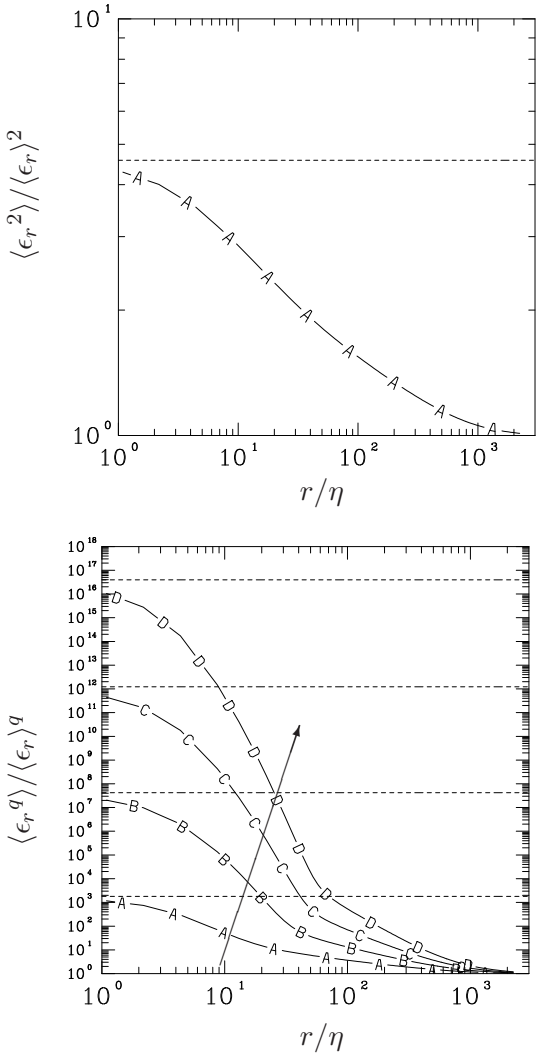

Figure 6.14: (Top) Second order moments of 3D local average of dissipation and (Bottom) higher order moments ($q = 3, 4, 5, 6$, increasing in the direction shown) of 3D local average of dissipation for $R_\lambda \sim 650$, 4096^3. Dashed lines indicate corresponding point-wise dissipation moments.

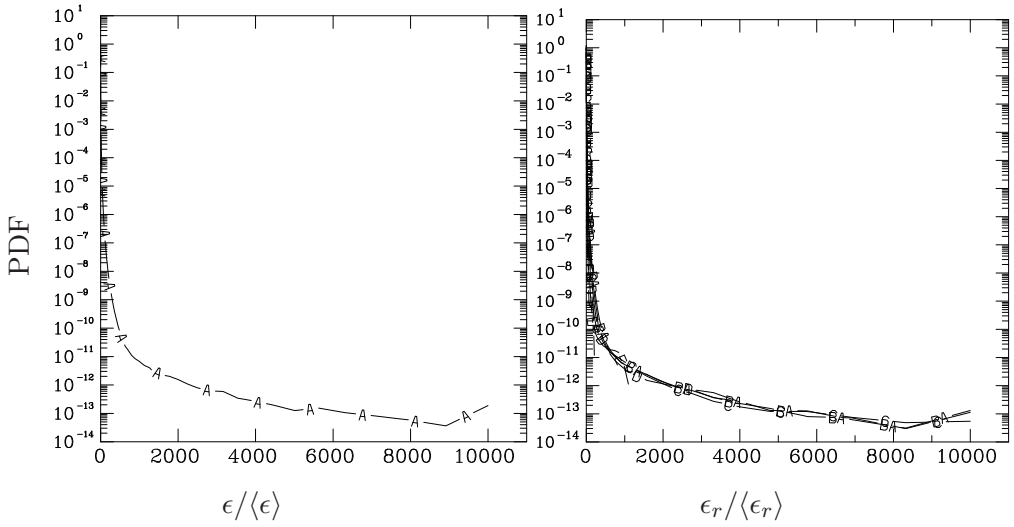

Figure 6.15: PDF of dissipation rate at $R_\lambda \sim 650$ for (left) single-point, (right) local 3D averages. Curves A, B, C etc. correspond to different averaging lengths $r/\eta = 1.1, 2.2, 4.4$ etc. The PDF for $r/\eta = 1.1, 2.2, 4.4$ seem to collapse onto each other.

average is given. The component averaged quantity is used to determined the extent of the inertial range in the following analysis.

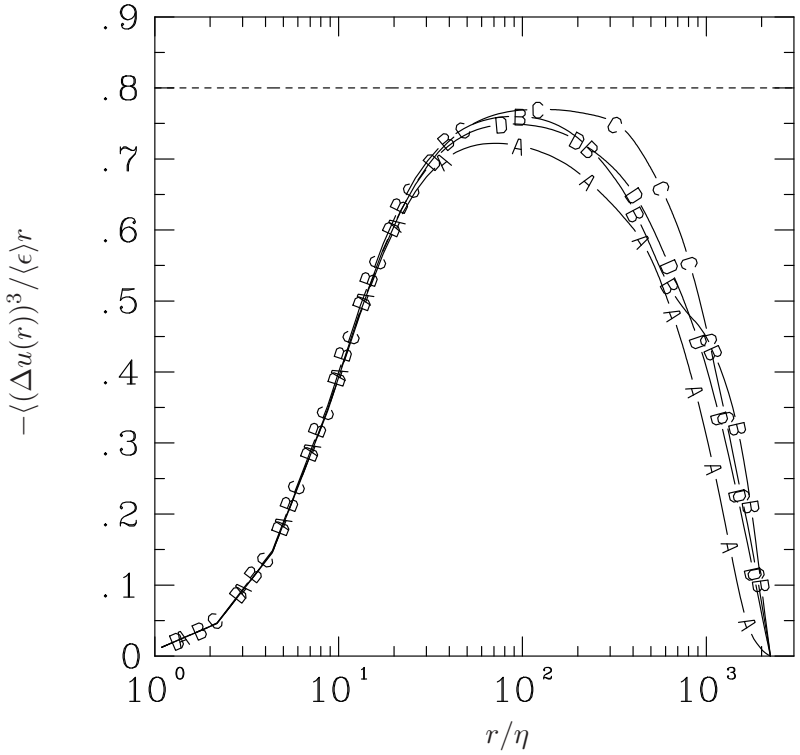

Figure 6.16: Normalized third-order velocity structure function for 4096^3, $R_\lambda \sim 650$. Curves A,B and C are correspond to the three Cartesian directions. Curves D is the component averaged result.

6.5.1 K62: second postulate

The second K62 hypothesis is that if the local Reynolds number $Re_r \gg 1$, the PDF of $V(r)$ for $\eta \ll r \ll L$, becomes independent of Re_r and hence universal. In our simulations, the viscosity is a constant. Hence it suffices to check for the dependence of V only on the local velocity scale $(r\epsilon_r)^{1/3}$.

Figure 6.17 shows the PDFs of V conditioned on $(r\epsilon_r)^{1/3}$ for $r/\eta \approx 70$ and 139, both in the inertial range (see Fig. 6.16). The PDFs coalesce for a range of values of $(r\epsilon_r)^{1/3}$ and the shape is preserved in going from $r/\eta = 70$ to 139, indicating an approximate independence of V from ϵ_r and r.

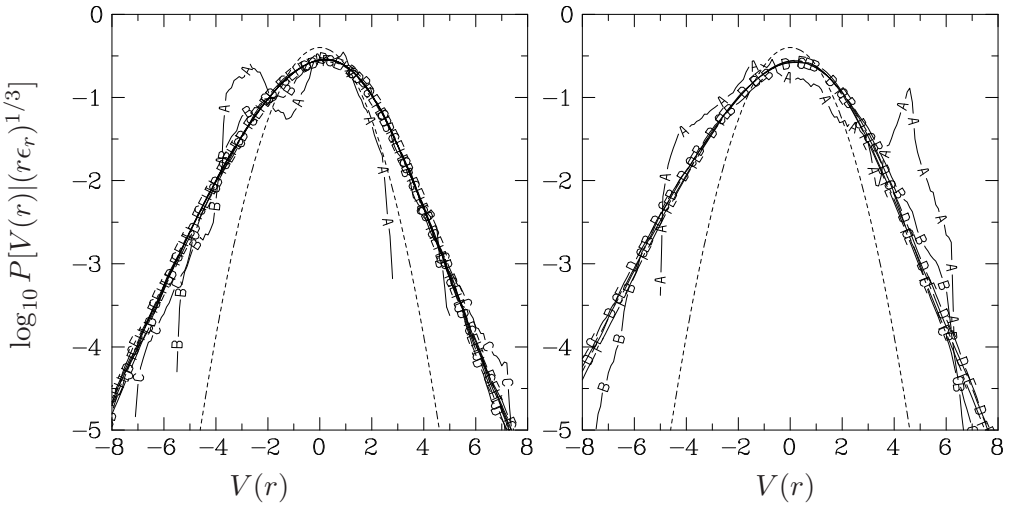

Figure 6.17: Conditional PDF of $V(r)$ in the inertial range for a given spatial separation. The separation distance r (fixed for each frame), the number of curves and the minimum and maximum values of Re_r are as follows: (a) (Left): $r/\eta = 70$, 10 curves with Re_r ranging from 71.4 (curve A) to 247.4 (curve J). (b) (Right): $r/\eta = 139$, 6 curves, with Re_r ranging from 248.0 (curve A) to 494.9 (curve F). In each frame, lines $A, B, C, \ldots$ are in increasing order of Re_r (Eq. 6.3). For example, curve A in each frame corresponds to smallest Re_r, followed by curve B and so on. Dashed curve is the Gaussian distribution with zero mean and unity variance.

In order to further test the correspondence of the data with the second postulate, consider the following equation,

$$\Delta u(r) = V(r\epsilon_r)^{1/3} \tag{6.26}$$

Following Stolovitzky *et al.* (1992), the conditional mean of the magnitude of the velocity

increment can be written as,

$$\langle |\Delta u(r)| \big| (r\epsilon_r)^{1/3} \rangle = \langle |V| (r\epsilon_r)^{1/3} \big| (r\epsilon_r)^{1/3} \rangle \tag{6.27}$$

$$= (r\epsilon_r)^{1/3} \langle |V| \big| (r\epsilon_r)^{1/3} \rangle \tag{6.28}$$

Now if r is in the inertial range, Fig. 6.17 suggests that V may be independent of $(r\epsilon_r)^{1/3}$, in which case the above equation becomes

$$\langle |\Delta u(r)| \big| (r\epsilon_r)^{1/3} \rangle = (r\epsilon_r)^{1/3} \langle |V| \rangle \tag{6.29}$$

Hence, if the second postulate holds then, $\langle |\Delta u(r)| \big| (r\epsilon_r)^{1/3} \rangle$ is a linear function of $(r\epsilon_r)^{1/3}$ in the inertial range. Figure 6.18 shows the logarithm of the conditional mean $\langle |\Delta u(r)| \big| (r\epsilon_r)^{1/3} \rangle$ as a function of the logarithm of $(r\epsilon_r)^{1/3}$. The curves corresponding to different spatial separations (in the inertial range) collapse onto a line with slope 1 in Fig. 6.18, except possibly at the tails where the sampling uncertainty can be large. This confirms that Eq. 6.29 approximately holds in the inertial range, and thereby provides further confirmation that at $R_\lambda \sim 650$, the second K62 postulate is approximately valid.

Since it has been shown the stochastic variable V, in the inertial range is independent of r and ϵ_r it is approximately universal in the inertial range. It then follows from Eq. 6.4 that the mth-order structure function is given by

$$\langle [\Delta u(r)]^m \rangle = \langle V^m \rangle \langle (r\epsilon_r)^{(m/3)} \rangle \, . \tag{6.30}$$

In particular, $\langle V(r) \rangle = 0$ in the inertial range. The second and third moments of V in the inertial range are as follows,

$$\langle V^2 \rangle = \frac{\langle (\Delta u(r))^2 \rangle}{r^{2/3} \langle {\epsilon_r}^{2/3} \rangle} \, , \tag{6.31}$$

$$\langle V^3 \rangle = \frac{\langle (\Delta u(r))^3 \rangle}{r \langle \epsilon_r \rangle} \tag{6.32}$$

Figure 6.19 shows the second moment of V (Eq. 6.31) as a function of spatial separation. In the inertial range, $\langle V^2 \rangle \approx 2.1$ (see left panel in Fig. 6.31), which is comparable to the accepted estimates of the Kolmogorov constant in second-order structure functions (Saddoughi & Veeravalli 1994). In the small-r limit, a Taylor expansion shows that $\langle V^2 \rangle$ varies as $r^{4/3}$. This is also confirmed in Fig. 6.19 (right panel).

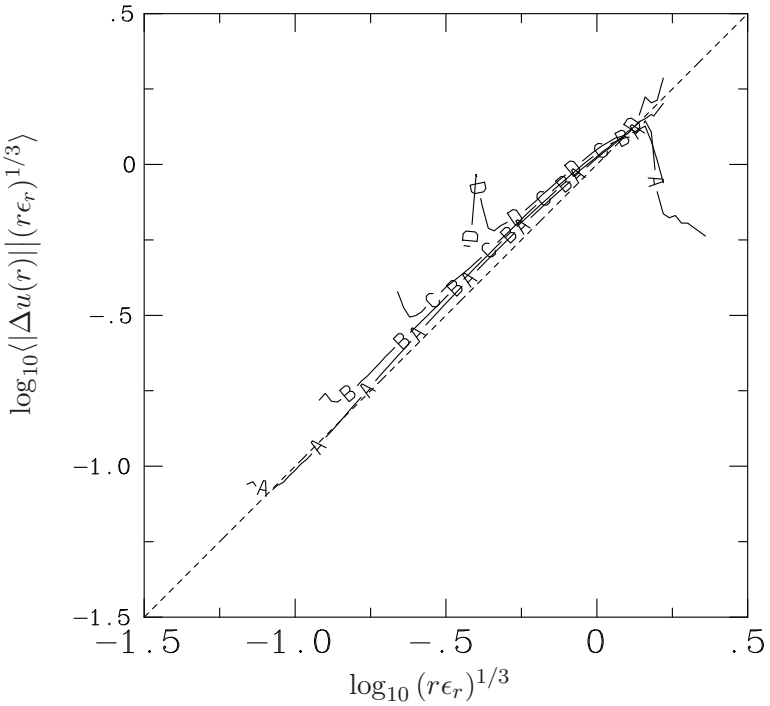

Figure 6.18: Logarithm of the mean of $|\Delta u(r)|$ conditioned on $(r\epsilon_r)^{1/3}$ as a function of the logarithm of $(r\epsilon_r)^{1/3}$. The lines A, B, C, D correspond to $r/\eta = 35, 70, 139$ and 279 respectively. Dashed line has slope 1.

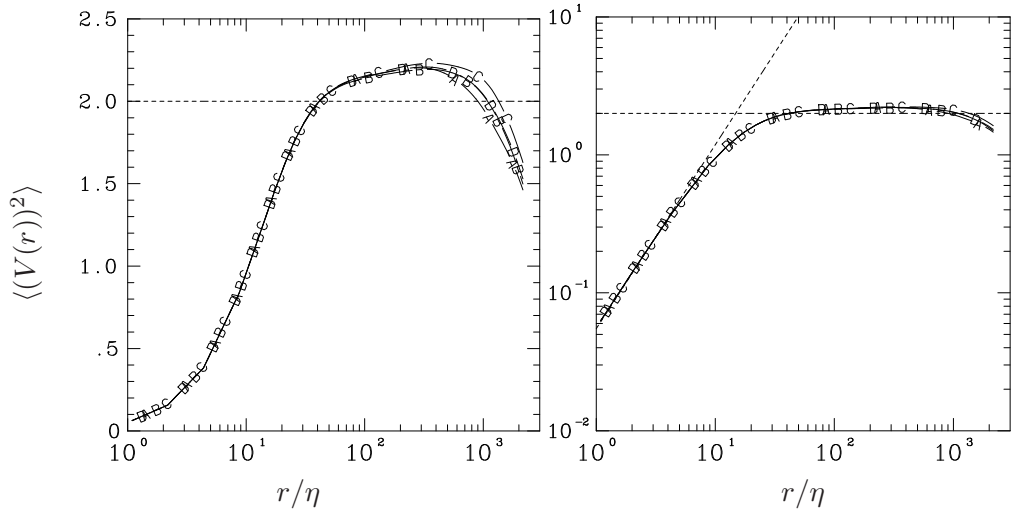

Figure 6.19: Second moment of V along three different Cartesian directions X,Y and Z (lines A,B and C) along with the component averaged plot (line D), for 4096^3, $R_\lambda \sim 650$. (a) (left) log-linear scales to check inertial range behavior (b) (right) log-log scales to check small-r behavior. Dashed lines (positive slope) corresponds to $r^{4/3}$ to check small-r slope. Dashed line (zero slope) at 2.0 for comparison with corresponding $K41$ result.

Figure 6.20 shows the third moment (Eq. 6.32) at $R_\lambda \sim 650$. It can be seen from the left panel of Fig. 6.20 that the skewness of V in the three Cartesian directions differ slightly in

the inertial range. The component averaged skewness (line D) in Fig. 6.20 gives $\langle V^3 \rangle = 0.8$ in the inertial range which is consistent with the corresponding $K41$ result. The right panel in Fig. 6.20 confirms that the slope of $\langle V^3 \rangle$ at small-r follows Eq. 6.25. A comparison of Fig. 6.20 (left panel) with the lower Reynolds number case of Fig. 6.13, shows that $\langle V^3 \rangle$ indeed approaches the 4/5th plateau with increasing Reynolds number.

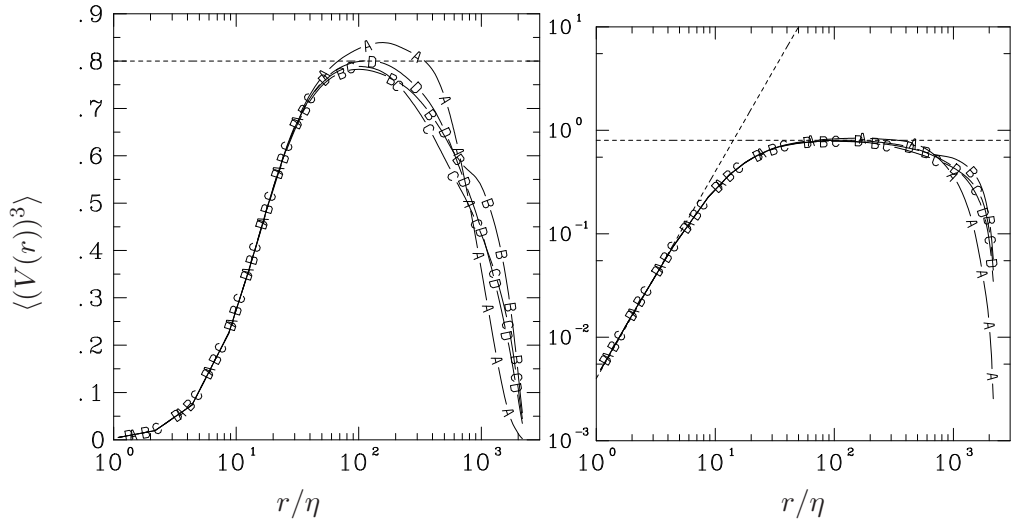

Figure 6.20: Third moment of V along three different Cartesian directions X,Y and Z (lines A,B and C) along with the component averaged plot (line D), for 4096^3, $R_\lambda \sim 650$. (a) (left) log-linear scales to check inertial range behavior (b) (right) log-log scales to check small-r behavior. Dashed lines (positive slope) to check the small-r slope (see Eq. 6.25) while dashed line (zero slope) at 0.8 shows the $K41$ plateau.

6.5.2 K62: first postulate

The statement of the first K62 postulate is that for $r \ll L$, the PDF of V depends only on Re_r. For the constant viscosity case, it is enough to check the dependence of V on $r(r\epsilon_r)^{1/3}$.

Figure 6.21 presents the PDF of V for $r/\eta = 4$ and 9, respectively. Each curve corresponds to a different value of $(r\epsilon_r)^{1/3}$. The uncertainty in the data is appreciably higher than in the corresponding inertial range PDFs (Fig. 6.17), because the averaging intervals are smaller. Regardless, the conclusion from Fig. 6.21 is that the PDF of V depends on $(r\epsilon_r)^{1/3}$ and r.

Figure 6.22 reports the PDFs of V conditioned on $(r\epsilon_r)^{1/3}$ for $Re_r \approx 27$ (left panel) and 39 (right panel), each at three different spatial separations $r/\eta = 4, 8, 17$. The PDFs

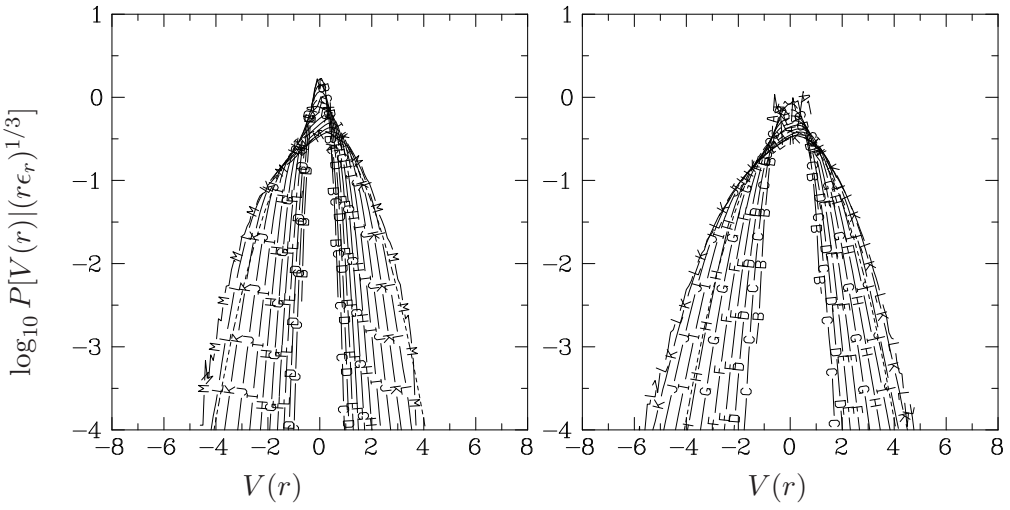

Figure 6.21: Conditional PDF of $V(r)$ in the small-scale range for a given spatial separation. The separation distance r (fixed for each frame), the number of curves and the minimum and maximum values of Re_r are as follows: (a) (Left): $r/\eta = 4$, 13 curves with Re_r ranging from 0.5 (curve A) to 38.8 (curve M). (b) (Right): $r/\eta = 9$, 12 curves, with Re_r ranging from 1.4 (curve A) to 77.7 (curve L). In each frame, lines $A, B, C, \ldots$ are in increasing order of Re_r (Eq. 6.3). For example, curve A in each frame corresponds to smallest Re_r, followed by curve B and so on. Dashed curve is the Gaussian distribution with zero mean and unity variance.

in each frame correspond to different spatial separations and $(r\epsilon_r)^{1/3}$ such that the local Reynolds number Re_r is approximately the same. Since ϵ_r is a random variable, exact correspondence in the values of Re_r is difficult, hence Re_r that are within 12% of each other are considered as approximately equal in this analysis. The PDFs collapse onto each other with some differences at the tails which can be attributed to the differences in the exact values of Re_r for the curves. From Fig. 6.22, it can be concluded that the PDFs of V for $r \ll L$, only depend on the local Reynolds number Re_r.

Figure 6.23 shows the logarithm of the conditional mean of $|V|$, conditioned on the local velocity $(r\epsilon_r)^{1/3}$ for the small-r range. Evidently the curves for different spatial separations (at small-r) do not collapse indicating a dependence on $(r\epsilon_r)^{1/3}$ and r.

Consider the conditional mean of $r|\Delta u(r)|$, conditioned on $r(r\epsilon_r)^{1/3}$,

$$\langle r|\Delta u(r)|\big|r(r\epsilon_r)^{1/3}\rangle = r(r\epsilon_r)^{1/3}\langle |V|\big|r(r\epsilon_r)^{1/3}\rangle \tag{6.33}$$

If V is only a function of Re_r (or $r(r\epsilon_r)^{1/3}$) for $r \ll L$, then $\langle |V|\big|r(r\epsilon_r)^{1/3}\rangle = |V|$, and the above equation becomes,

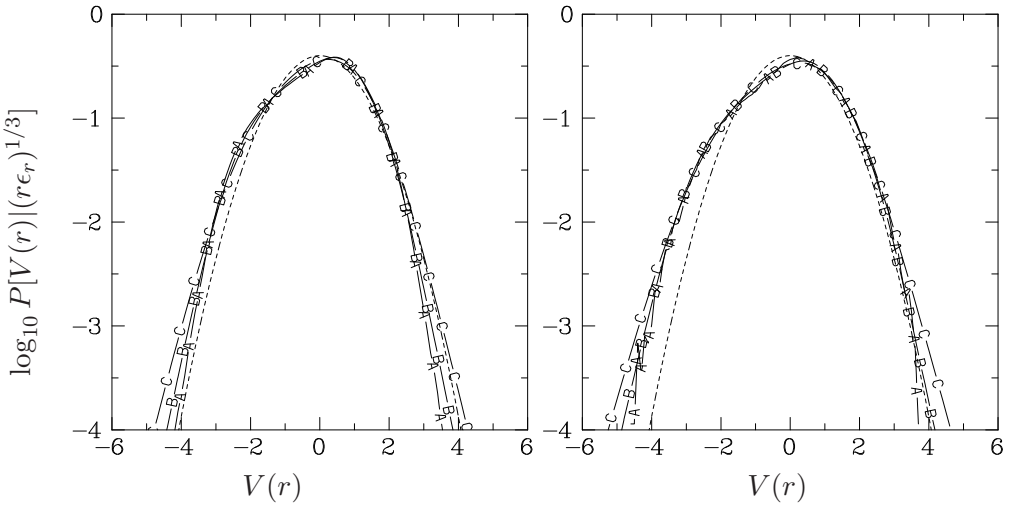

Figure 6.22: Conditional PDF of $V(r)$ in the small-scale range for a given local Reynolds number Re_r, for different spatial separations. (a) (Left): $Re_r \approx 27$, curves A, B, C correspond to spatial separations $r/\eta = 4, 8$ and 17 respectively. (b) (Right): $Re_r \approx 39$, curves A, B, C correspond to spatial separations $r/\eta = 4, 8$ and 17 respectively. The exact values of Re_r are within 12% of each other in each panel. Dashed curve is the Gaussian distribution with zero mean and unity variance.

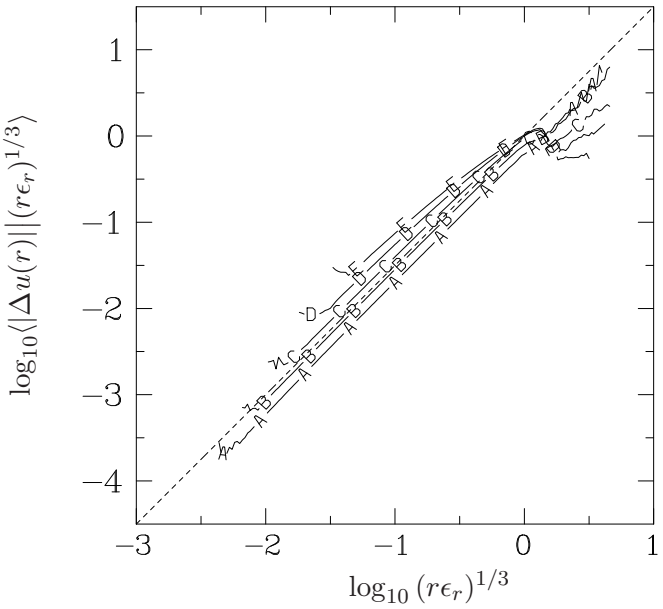

Figure 6.23: Expectation of the logarithm of $|\Delta u(r)|$ conditioned on $(r\epsilon_r)^{1/3}$, for different spatial separations in the small-r range as a function of the logarithm of $(r\epsilon_r)^{1/3}$. Curves A,B,C,D and E correspond to spatial separations $r/\eta = 1, 2, 4, 8$ and 16 respectively. Dashed line has slope of 1.5.

$$\langle r|\Delta u(r)|\big|r(r\epsilon_r)^{1/3}\rangle = |V|r(r\epsilon_r)^{1/3} \tag{6.34}$$

It follows that the left-hand-side of the Eq. 6.34 is only a function of $r(r\epsilon_r)^{1/3}$ if the first postulate is valid. Figure 6.24 shows the logarithm of the left-hand-side of 6.34 plotted against the logarithm of $r(r\epsilon_r)^{1/3}$ for different spatial separations in the small-r range. The curves collapse on to one another except possibly at the tails where statistical uncertainty can be large, indicating that the first hypothesis is approximately valid at $R_\lambda \sim 650$.

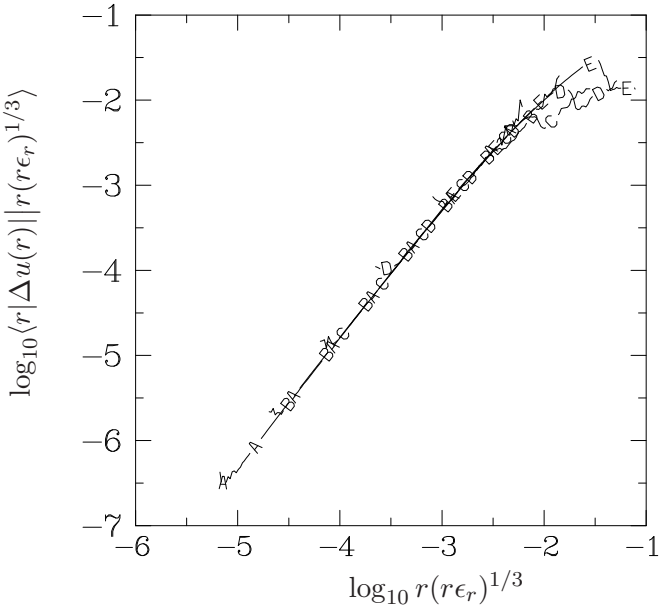

Figure 6.24: Expectation of logarithm of $r|\Delta u(r)|$ in the small-r range conditioned on $(r\epsilon_r)^{1/3}$ for different spatial separations as a function of the logarithm of $r(r\epsilon_r)^{1/3}$. Curves A,B,C,D and E correspond to spatial separations $r/\eta = 1, 2, 4, 8$ and 16 respectively.

6.6 Summary

A new algorithm to calculate local 1D and 3D averages is presented. At small separation distances, local $3D$ averages appear to have a stricter resolution requirement than local 1D averages. At large-r, boundary conditions seem to influence 1D averages to a greater extent. Local 3D averages of dissipation have smaller moments than the corresponding local 1D averages of dissipation for orders two and above. The scaling in the intermediate scale-range also appears to be different for local 3D and 1D averages of dissipation. The tails of PDFs of 3D dissipation are also narrower than those of 1D dissipation for a given averaging length. In general, the data suggest that the local 3D averages of dissipation are less intermittent than the local 1D averages of dissipation.

The local 3D averaged dissipation has been used to examine the first and second postulates of K62 for a 4096^3 data set at $R_\lambda \sim 650$. The variance and skewness of V in the inertial range is shown to be consistent with corresponding $K41$ predictions. The basic tenets of the first and second K62 postulates have been found to be approximately true at this Reynolds number.

CHAPTER VII

CONCLUSIONS

Direct Numerical Simulations (DNS) have been used to study passive scalar mixing and turbulence structure over a range of Reynolds and/or Schmidt numbers. The DNS algorithm uses three-dimensional ($3D$) Fast Fourier Transforms (FFT) which are known to scale poorly with increasing problem size. A hybrid (MPI/OpenMP) FFT algorithm has been developed which is shown to be faster than the pure MPI version at very large problem sizes. Parallel algorithms have been developed to calculate spherical averages and $3D$ local averages at Reynolds numbers as high as 650. These statistics have provided deeper insights into the small-scale phenomenological picture of turbulence. In light of recent studies (Yakhot & Sreenivasan 2005; Donzis *et al.* 2008*b*; Donzis & Yeung 2010), adequate attention has been given to small-scale resolution wherever appropriate. In the following, the conclusions for the main topics undertaken in this thesis are summarized. This is followed by some remarks about possible future work.

7.1 Summary of the main conclusions

7.1.1 Transpose-based 3D FFT

Parallel $3D$ FFT algorithms are critical to pseudo-spectral simulations in a number of fields including turbulence. Their performance is closely linked to the communication characteristics of the network topology of the interconnect used in the parallel system. As end users it is not always possible to have control on the job placements and traffic in the network topology. In such a scenario it is important to maximize the algorithmic efficiency in scheduling the computations and communications in a synergistic manner. A hybrid MPI/OpenMP model has been developed where the OpenMP threads can communicate sequentially with other threads. The communication and computation has been divided among the threads using a pipeline approach to mask some of the communication behind

the computation. Overlapping communication and computation has been shown to decrease the communication time by as much as 30% for a 4096^3 problem on the Blue Waters machine (NCSA).

The all-to-all communication is shown to consume as much as 80% of the total time for a 4096^3 box using 32768 cores. An important point of consideration is how this trend is going to extrapolate to larger problem sizes for exascale computing (10^{18} floating point operations per second). Greater attention has to be given to the network bisection bandwidth, as the algorithm becomes more communication dominated. New and improved communication protocols have to be developed to alleviate the communication characteristics of the 3D FFT. Of special interest is the use of asynchronous communication, which allows (partial) data that is sent across the network to be used as soon as it reaches the receiving process.

7.1.2 Low Schmidt number turbulent mixing

Chapter 4 presents numerical results for scaling properties of structure functions of passive scalar mixing in turbulence. The emphasis in this work has been on the highly diffusive (very low Schmidt number) regime. Data from DNS at Schmidt numbers as low as 1/2048 show that as Schmidt number decreases for a fixed Reynolds number, the range of scales decreases with the distribution of the small-scales approaching a Gaussian character. This is consistent with the fact that intermittency becomes weaker for strongly diffusive scalars. However, local isotropy is not recovered in the low Schmidt number limit and substantial differences in the statistics along different directions exist. Agreement with the Yaglom's relation is seen to be substantially closer with an increase in Reynolds number and/or Schmidt number due to a wider range of scales, but clear departures have been observed in the low Schmidt number regime. Examination of the structure function budget shows that an extended diffusive regime which is characteristic of low Schmidt number scalars tends to limit the inertial-convective behavior resulting in a deviation from the Yaglom's 2/3 plateau. A new scaling law for the inertial-diffusive range has been proposed for the low Schmidt number regime. However due to limitations in Reynolds numbers and Schmidt number choices presently available, the inertial-diffusive scaling behavior has not been examined.

7.1.3 Assessing K41 laws for finite Reynolds number simulations

One of the main benchmarks in DNS of three-dimensional turbulence is the Kolmogorov prediction for third-order structure function, involving isotropic statistics in the infinite Reynolds number limit. Typically, DNS techniques to obtain isotropic statistics have relied on time-averaging structure functions in a few directions over many eddy-turnover times, using isotropic forcing schemes. As the Reynolds number and the problem size increase, long simulation times, and the I/O time incurred in saving the Petabytes of data generated can become expensive. Motivated by recent theoretical and numerical work (Eyink 2003; Taylor *et al.* 2003), a new parallel algorithm to compute spherical averages has been developed. The parallel algorithm avoids the expensive ALLTOALL type communication, which has been previously used to calculate such quantities. Consequently, the parallel scheme developed in this thesis has been shown to scale well up to 16384 cores on Stampede (TACC). An extended inertial range is obtained by using the angle-averaging technique which extracts the isotropic component of the data by isolating the flow anisotropy. The local laws proposed by Nie & Tanveer (1999) and Eyink (2003) have also been examined using DNS data. In particular, the time local result of Eyink (2003) has been shown to hold well. The angle averaging technique has also been used to test the Yaglom's relation in the scalar field. The scalar field in our simulations is forced by the presence of a non-zero mean gradient. The non-zero mean gradient in one direction renders the scalar field strongly anisotropic, resulting in systematic differences in statistics parallel and perpendicular to it. The hope is that using angle averaging techniques, we can isolate the anisotropy to recover isotropic statistics as was done for the velocity field. However, greater care is required for the scalar field, since a high Reynolds number and a high Schmidt number are required, while sufficiently resolving the small and intermediate scales. More generator points on the Voronoi sphere may also be required to adequately resolve the small and intermediate scales in the scalar field. This is still an ongoing work and results will be reported elsewhere.

7.1.4 Local 3D averages and the Refined Similarity Hypothesis

Locally averaged quantities are less intermittent than the single-point quantities. The 3D local average of dissipation was used by Kolmogorov in the refined similarity theory (Kolmogorov 1962) to account for intermittency. However stringent testing of this theory using computer simulations has eluded researchers so far, partly due to difficulties in calculating 3D local averages. An efficient algorithm to calculate 3D local averages has been developed. The algorithm computes three 1D local sums in the three Cartesian directions by constructing prefix sums in each of the three directions. This code is seen to scale well up to 32768 cores on Kraken. Using the 3D averaged local dissipation, the first two postulates of the Refined similarity theory have been examined. Numerical results at a high Reynolds number ($R_\lambda \sim 650$) have been reported. Results show that the first and the second hypothesis (Kolmogorov 1962) are approximately valid.

7.2 Future work

The study of high Schmidt number passive scalar mixing is important not only from a theoretical perspective, but also from an engineering point of view. Pseudo-spectral schemes which are highly efficient in capturing the relevant flow physics are very expensive at large Peclet numbers. There has been considerable interest in the DNS community regarding hybrid numerical methods (Gotoh *et al.* 2012) involving a spectral scheme for the velocity field and a compact difference scheme for the passive scalar evolution. Such hybrid schemes do not exhaustively depend on global ALLTOALL type data exchanges and hence are more suited to run on Graphical Processing Units (GPU). Using GPU based codes may be not only more affordable, but also faster than regular CPU computations.

The spherical averaging technique presented in this work can be used to isolate the anisotropic contributions (Biferale *et al.* 2002) themselves by subtracting from the full structure function, its angle-averaged value. Individual moments in a spherical harmonics expansion of structure functions can be computed by using basis functions of interest to the angle-averaging integrand. Dominant scaling in anisotropic regions can then be determined which is important to determine the rate of return to isotropy at small scales.

To conclude, this thesis has developed parallel numerical algorithms to calculate statistics such as spherical averages and local averages at Reynolds numbers, that previously have not been undertaken. Passive scalar mixing for a range of Schmidt numbers has been examined to study appropriate scaling behavior in physical space. A thread-serialized hybrid MPI/OpenMP FFT algorithm has been developed which is shown to perform better at very large problem sizes. All this has only been made possible due to increased interaction between experts in computational sciences and those interested in physics. The quest for reaching higher Reynolds numbers may be a never ending one. But even at the Reynolds numbers currently feasible in computer simulations, detailed statistics can be extracted leading to deeper insights into the physics of turbulence. It is important to develop theories and numerical techniques that are suited to finite Reynolds number flows which are almost surely limited by finite sampling and finite resolution considerations. This thesis is one such attempt in this direction.

Bibliography

Antonia, R. A. & Burattini, P 2006 Approach to the 4/5 law in homogeneous isotropic turbulence. *J. Fluid Mech.* **550**, 175–184.

Ashurst, W. T., Kerstein, A. R., Kerr, R. M. & Gibson, C. H. 1987 Alignment of vorticity and scalar gradient with strain rate in simulated navier-stokes turbulence. *Phys. Fluids* **30**, 2343–2353.

Batchelor, G. K. 1959 Small-scale variation of convected quantities like temperature in turbulent fluid Part 1. General discussion and the case of small conductivity. *J. Fluid Mech.* **5**, 113–139.

Batchelor, G. K., Howells, I. D. & Townsend, A. A. 1959 Small-scale variation of convected quantities like temperature in turbulent fluid. part 2. The case of large conductivity. *J. Fluid Mech.* **5**, 134–139.

Biferale, L, Lohse, D, Mazzitelli, I & Toschi, F 2002 Probing structures in channel flow through SO(3) and SO(2) decomposition. *J. Fluid Mech.* **452**, 39–59.

Bilger, R. W. 2004 Some aspects of scalar dissipation. *Flow, Turb. & Combust.* **72**, 93–114.

Blelloch, G. 1990 Prefix sums and their applications. *Tech. Rep.* CMU-CS-90-190. School of Computer Science, Carnegie Mellon University.

Canuto, C., Hussaini, M. Y., Quarteroni, A. & Zang, T. A. 1987 *Spectral Methods in Fluid Dynamics*. Springer-Verlag.

Chan, A., Balaji, P., Gropp, W. & Thakur, R. 2008 Communication analysis of parallel 3D FFT for flat Cartesian meshes on large Blue Gene systems. In *15th IEEE Int. Conf. on High Perf. Comp.*, pp. 422–429.

Chapman, B., Jost, G. & Pas, R. V. 2007 *Using OpenMP: Portable Shared Memory Parallel Programming (Scientific and Engineering Computation)*. The MIT Press.

Cooley, J. W. & Tukey, J. W. 1965 An algorithm for the machine calculation of complex Fourier series. *Math. Comput.* pp. 297–301.

Corrsin, S. 1951 On the spectrum of isotropic temperature fluctuation. *J. Appl. Phys.* **22**, 469–473.

Czechowski, T., McClanahan, C., Battaglino, C., Iyer, K. P., Yeung, P. K. & R., Vuduc 2012 On the communication complexity of 3D FFTs and its implications for exascale. In *Proc. ACM Int'l. Conf. Supercomputing (ICS)*. San Servolo Island, Venice, Italy.

Dimotakis, P. E. 2005 Turbulent mixing. *Annu. Rev. Fluid Mech.* **37**, 329–356.

Doi, J. & Negishi, Y. 2010 Overlapping methods of all-to-all communication and FFT algorithms for torus-connected massively parallel supercomputers. In *Proc. of the 2010 ACM/IEEE Int. Conf. for High Performance Computing, Networking, Storage and Analysis*, *SC '10* , pp. 1–9. Washington, DC, USA: IEEE Computer Society.

Domaradzki, J. A. & Rogallo, R. S. 1990 Local energy-transfer and nonlocal interactions in homogeneous, isotropic turbulence. *Phys. Fluids* **2**, 413–426.

Donzis, D. A., Sreenivasan, K. R. & Yeung, P. K. 2005 Scalar dissipation rate and dissipative anomaly in isotropic turbulence. *J. Fluid Mech.* **532**, 199–216.

Donzis, D. A., Sreenivasan, K. R. & Yeung, P. K. 2010 The Batchelor Spectrum for Mixing of Passive Scalars in Isotropic Turbulence. *Flow Turbul. & Combust.* **85**, 549–566.

Donzis, D. A. & Yeung, P. K. 2010 Resolution effects and scaling in numerical simulations of passive scalar mixing in turbulence. *Physica D* **239**, 1278–1287.

Donzis, D. A., Yeung, P. K. & Pekurovsky, D. 2008*a* Turbulence simulations on $O(10^4)$ processors. In *Proc. TeraGrid '08 Conf.*.

Donzis, D. A., Yeung, P. K. & Sreenivasan, K. R. 2008*b* Dissipation and enstrophy in isotropic turbulence: Resolution effects and scaling in direct numerical simulations. *Phys. Fluids* **20**, 045108.

Dubey, A. & Tessera, D. 2001 Redistribution strategies for portable parallel fft: a case study. *Concurrency and Computation: Practice and Experience* **13**, 209–220.

Duchon, J & Robert, R 2000 Inertial energy dissipation for weak solutions of incompressible Euler and Navier-Stokes equations. *Nonlinearity* **13**, 249–255.

Eswaran, V. & Pope, S. B. 1988 An examination of forcing in direct numerical simulations of turbulence. *Comput. Fluids* **16**, 257–278.

Eyink, G 2003 Local 4/5-law and energy dissipation anomaly in turbulence. *Nonlinearity* **16**, 137–145.

Foster, T. & Worley, H. 1997 Parallel algorithms for the spectral transform method. *SIAM J. Sci. Comput.* **18**, 806–837.

Frigo, M. & Johnson, S. G. 2005 The design and implementation of FFTW3. *Proc. IEEE* pp. 216–231, special issue on "Program Generation, Optimization, and Platform Adaptation".

Frisch, U. 1995 *Turbulence*. Cambridge University Press.

Galanti, B. & Tsinober, A. 2004 Is turbulence ergodic? *Phys. Lett. A* **330**, 173 – 180.

Gotoh, G., Fukayama, D. & Nakano, T. 2002 Velocity field statistics in homogeneous steady turbulence obtained using a high-resolution direct numerical simulation. *Phys. Fluids* **14**, 1065–1081.

Gotoh, T., Hatanaka, S. & Miura, H. 2012 Spectral compact difference hybrid computation of passive scalar in isotropic turbulence. *J. Comp. Phys.* **231**, 7398–7414.

Gotoh, T. & Yeung, P. K. 2013 Passive scalar transport in turbulence: A computational perpsective. In *Ten Chapters in Turbulence* (ed. P. A. Davidson, Y. Kaneda & K. R. Sreenivasan). Cambridge University Press.

Grant, H. L., Stewart, R. W. & Moilliet, A. 1962 Turbulence spectra from a tidal channel. *J. Fluid Mech.* **12**, 241–268.

Gropp, W., Lusk, E. & Thakur, R. 1999 *Using MPI-2*. Cambridge University Press.

Gropp, W. & Thakur, R. 2006 Thread-safety in MPI implementation: Requirement and analysis. *Parallel Computing* pp. 595–604.

Hill, R 2002 Structure-function equations for scalars. *Phys. Fluids* **14**, 1745–1756.

Holzer, M. & Siggia, E. D. 1994 Turbulent mixing of a passive scalar. *Phys. Fluids* **6**, 1820–1837.

Ishihara, T., Gotoh, T. & Kaneda, Y. 2009 Study of high-Reynolds number isotropic turbulence by direct numerical simulation. *Annu. Rev. Fluid Mech.* **41**, 165–180.

Ishihara, T., Kaneda, Y., Yokokawa, M., Itakura, K. & Uno, A. 2007 Small-scale statistics in high-resolution direct numerical simulation of turbulence: Reynolds number dependence of one-point velocity gradient statistics. *J. Fluid Mech.* **592**, 335–366.

Iyer, K. P. & Yeung, P. K. 2014 Structure functions and applicability of Yaglom's relation in passive-scalar turbulent mixing at low schmidt numbers with uniform mean gradient. *Phys. Fluids* **submitted**.

Jackson, D. & Launder, B. 2007 Osborne Reynolds and the publication of his papers on turbulent flow. *Annu. Rev. Fluid Mech.* **39**, 19–35.

Kadoch, B., Iyer, K., Donzis, D., Schneider, K., Farge, M. & Yeung, P. K. 2011 On the role of vortical structures for turbulent mixing using direct numerical simulation and wavelet-based coherent vorticity extraction. *J. Turb.* **12**, N20.

Kaneda, Y., Ishihara, T., Yokokawa, M., Itakura, K. & Uno, A. 2003 Energy dissipation rate and energy spectrum in high resolution direct numerical simulations of turbulence in a periodic box. *Phys. Fluids* **15**, L21–L24.

Kerr, R. M. 1985 Higher-order derivative correlations and the alignment of small-scale structures in isotropic numerical turbulence. *J. Fluid Mech.* **153**, 31–58.

Kirk, D. & Hwu, W. M. 2010 *Programming Massively Parallel Processors: A Hands-on Approach*, 1st edn. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Kolmogorov, A. N. 1941a Local structure of turbulence in an incompressible fluid for very large reynolds numbers. *Dokl. Akad. Nauk. SSSR* **30**, 299–303.

Kolmogorov, A. N. 1941b Dissipation of energy in locally isotropic turbulence. *Dokl. Akad. Nauk. SSSR* **434**, 16–18.

Kolmogorov, A. N. 1962 A refinement of previous hypothesis concerning the local structure of turbulence in viscous incompressible fluid at high reynolds number. *J. Fluid Mech.* **13**, 82–85.

Kraichnan, R. H. 1994 Anomalous scaling of a randomly advected passive scalar. *Phys. Rev. Lett.* **72**, 1016–1019.

Kumar, S., Sabharwal, Y & Heidelberger, P. 2008 Optimization of all-to-all communication on the blue gene/l supercomputer. *International Conference on Parallel Processing* pp. 320–329.

Lepreti, F., Carbone, V., Spolaore, M., Antoni, V., Cavazzana, R., Martines, E., Serianni, G., Veltri, P., Vianello, N. & Zuin, M. 2009 Yaglom law for electrostatic turbulence in laboratory magnetized plasmas. *Europhys. Lett.* **86**, 1–5.

Moin, P. & Mahesh, K. 1998 Direct numerical simulation: A tool in turbulence research. *Annu. Rev. Fluid Mech.* **30**, 539–578.

Monin, A. S. & Yaglom, A. M. 1975 *Statistical Fluid Mechanics*, , vol. 2. MIT Press.

Moore, G. E. 1965 Cramming more components onto integrated circuits. *Electronics* .

Nie, Q & Tanveer, S 1999 A note on third-order structure functions in turbulence. *Proc. Royal Soc. A-Mathematical physical and Engineering Sciences* **455**, 1615–1635.

Oboukhov, A. M. 1962 Some specific features of atmospheric turbulence. *J. Fluid Mech.* **13**, 77–81.

Obukhov, A. M. 1949 The structure of the temperature field in a turbulent flow. *Dokl. Akad. Nauk. SSSR* **39**, 391.

Okabe, A, Boots, B & Sugihara, K 1992 *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. New York, NY, USA: John Wiley & Sons, Inc.

Orszag, S. A. & Patterson, G. S. 1972 Numerical simulation of 3-dimensional homogeneous isotropic turbulence. *Phys. Rev. Lett.* **28**, 76–&.

Overholt, M. R. & Pope, S. B. 1996 Direct numerical simulation of a passive scalar with imposed mean gradient in isotropic turbulence. *Phys. Fluids* **8**, 3128–3148.

Pekurovsky, D. 2012 P3DFFT: A framework for parallel computations of Fourier transforms in three dimensions. *SIAM J. Sci. comp.* pp. C192–C209.

Peyret, R. & Taylor, T. D. 1983 *Computational Methods for Fluid Flow*. Springer-Verlag.

Pope, S. B. 1994 Lagrangian PDF methods for turbulent flows. *Annu. Rev. Fluid Mech.* **26**, 23–63.

Pope, S. B. 2000 *Turbulent Flows*. Cambridge University Press.

Prasad, R. R., Meneveau, C. & Sreenivasan, K. R. 1988 Multifractal nature of the dissipation field of passive scalars in fully turbulent flows. *Phys. Rev. Lett.* **61**, 74–77.

Pumir, A. 1994 A numerical study of the mixing of a passive scalar in three dimensions in the presence of a mean gradient. *Phys. Fluids* **6**, 2118–2132.

Renka, R 1984 Interpolation of data on the surface of a sphere. *ACM Trans. Math. Softw.* **10**, 417–436.

Renka, R. 1997 Algorithm 772: Stripack: Delaunay triangulation and voronoi diagram on the surface of a sphere. *ACM Trans. Math. Softw.* **23**, 416–434.

Richardson, L. F. 1922 *Weather Prediction by Numerical Process*. Cambridge University Press.

Rogallo, R. S. 1981 Numerical experiments in homogeneous turbulence. *NASA Tech. Memo* .

Saddoughi, S. G. & Veeravalli, S. V. 1994 Local isotropy in turbulent boundary-layers at high Reynolds-number. *J. Fluid Mech.* **268**, 333–372.

Sanders, J. & Kandrot, E. 2010 *CUDA by Example: An Introduction to General-Purpose GPU Programming*, 1st edn. Addison-Wesley Professional.

Shraiman, B & Siggia, E 2000 Scalar turbulence. *Nature* **405**, 639–646.

Sorriso-Valvo, L., Marino, R., Carbone, V., Noullez, A., Lepreti, F., Veltri, P., Bruno, R., Bavassano, B. & Pietropaolo, E. 2007 Observation of inertial energy cascade in interplanetary space plasma. *Phys. Rev. Lett.* **99** (11).

Speziale, C. G. 1991 Analytical methods for the development of Reynolds-Stress closures in turbulence. *Annu. Rev. Fluid Mech.* **23**, 107–157.

Sreenivasan, K. R. 1991 Do scalar fluctuations in turbulent shear flows posses local universality? *Physica* **51**, 567–568.

Sreenivasan, K. R. 1996 The passive scalar spectrum and the obukhov-corrsin constant. *Phys. Fluids* **8**, 189–196.

Sreenivasan, K. R. 1998 An update on the energy dissipation rate in isotropic turbulence. *Phys. Fluids* **10**, 528–529.

Sreenivasan, K. R. & Antonia, R. A. 1997 The phenomenology of small-scale turbulence. *Annu. Rev. Fluid Mech.* **29**, 435–472.

Sreenivasan, K. R. & Kailasnath, P. 1993 An update on the intermittency exponent in turbulence. *Phys. Fluids* **5**, 512–514.

Sreenivasan, K. R. & Tavoularis, S. 1980 On the skewness of the temperature derivative in turbulent flows. *J. Fluid Mech.* **101**, 783–795.

Stolovitzky, G., Kailasnath, P. & Sreenivasan, K. R. 1992 Kolmogorov's refined similarity hypotheses. *Phys. Rev. Lett.* **69**, 1178–1181.

Stolovitzky, G., Kailasnath, P. & Sreenivasan, K. R. 1995 Refined similarity hypotheses for passive scalars mixed by turbulence. *J. Fluid Mech.* **297**, 275–291.

Taylor, G. I. 1935 Statistical theory of turbulence. *Proc. of the Royal Society of London. Series A - Mathematical and Physical Sciences* **151**, 421–444.

Taylor, G. I. 1938 The spectrum of turbulence. *Proc. of the Royal Society of London. Series A - Mathematical and Physical Sciences* **164**, 476–490.

Taylor, M. A., Kurien, S. & Eyink, G. L. 2003 Recovering isotropic statistics in turbulence simulations: The kolmogorov 4/5th law. *Phys. Rev. E* p. 026310.

Tsuji, M. & Sato, M. 2009 Performance evaluation of OpenMP and MPI hybrid programs on a large scale multi-core multi-socket cluster, T2K Open Supercomputer. In *Int. Conf. on Parallel Process. Workshops (ICPPW 2009)*, pp. 206–213. IEEE.

Vedula, P. & Yeung, P. K. 1999 Similarity scaling of acceleration and pressure statistics in numerical simulations of turbulence. *Phys. Fluids* **11**, 1208–1221.

Vedula, P., Yeung, P. K. & Fox, R. O. 2001 Dynamics of scalar dissipation in isotropic turbulence: a numerical andmodelling study. *J. Fluid Mech.* **433**, 29–60.

Wang, L. P., Chen, S. & Brasseur, J. G. 1999 Examination of hypotheses in the Kolmogorov refined turbulence theory through high-resolution simulations. Part 2. Passive scalar field. *J. Fluid Mech.* **400**, 163–197.

Wang, L. P., Chen, S., Brasseur, J. G. & Wyngaard, J. C. 1996 Examination of hypotheses in the Kolmogorov refined turbulence theory through high-resolution simulations. Part 1. Velocity field. *J. Fluid Mech.* **309**, 113–156.

Warhaft, Z. 2000 Passive scalars in turbulent flows. *Annu. Rev. Fluid Mech.* **32**, 203–240.

Warhaft, Z. 2009 Why we need experiments at high Reynolds numbers. *Fluid Dyn. Res.* **41**, 021401.

Yaglom, A. M. 1949 On the local structure of a temperature field in a turbulent flow. *Dokl. Akad. Nauk NAUK SSSR* **69**, 743–746.

Yakhot, V. & Sreenivasan, K. R. 2005 Anomalous scaling of structure functions and dynamic constraints on turbulence simulations. *J. Stat. Phys.* **121**, 823–841.

Yeung, P. K., Donzis, D. A., & Sreenivasan, K. R. 2012 Dissipation, enstrophy and pressure statistics in turbulence simulations at high reynolds numbers. *J. Fluid Mech.* pp. 5–15.

Yeung, P. K., Donzis, D. A. & Sreenivasan, K. R. 2005 High-Reynolds-number simulation of turbulent mixing. *Phys. Fluids* **17**, 081703.

Yeung, P. K. & Pope, S. B. 1989 Lagrangian statistics from direct numerical simulations of isotropic turbulence. *J. Fluid Mech.* **207**, 531–586.

Yeung, P. K. & Sreenivasan, K. R. 2013 Spectrum of passive scalars of high molecular diffusivity in turbulent mixing. *J. Fluid Mech.* **716**, R14.

Yeung, P. K. & Sreenivasan, K. R. 2014 Direct numerical simulation of turbulent mixing at very low Schmidt number with a uniform mean gradient. *Phys. Fluids* **26**, 015107.

Yeung, P. K., Xu, S. & Sreenivasan, K. R. 2002 Schmidt number effects on turbulent transport with uniform mean scalar gradient. *Phys. Fluids* **14**, 4178–4191.

Yeung, P. K. & Zhou, Y. 1997 On the universality of the kolmogorov constant in numerical simulations of turbulence. *Phys. Rev. E* **56**, 1746–1752.