Particle method: an efficient tool for direct numerical simulations of a high Schmidt number passive scalar in turbulent flow

By J. B. Lagaert†, G. Balarac†, G.-H. Cottet‡ AND P. Bégou†

In this work, an efficient way to predict the dynamics of a scalar at high Schmidt numbers, advected by a turbulent flow, is presented. For high Schmidt numbers, the spatial resolution required for the scalar field has to be finer than the resolution required for the velocity field, leading to a significant computational cost due to the Courant-FriedrichsLewy (CFL) constraint. We propose here a remeshed particle method coupled to a spectral flow solver to overcome this computational cost limitation. This allows us to perform a systematic analysis of flows over a wide range of Reynolds and Schmidt numbers. For high enough Reynolds and Schmidt numbers, the results presented here recover the spectral behavior predicted by theory. First, the classic $k^{-5/3}$ law (where k is the wave number) is found for the inertial-convective range. At intermediate scales, the viscous-convective range exhibits a k^{-1} law for Schmidt numbers higher than unity. Finally, the numerical results indicate that the dissipation range agree well with the Kraichnan model for high Schmidt numbers.

1. Introduction

The prediction of the dynamics of a scalar advected by a turbulent flow is an important challenge in many applications. The scalar field can be used to describe the temperature field, the concentration of chemical species or a level-set function to capture the interface in multiphase flows. For a passive scalar, Z, the transport equation is an advection-diffusion equation,

$$\frac{\partial Z}{\partial t} + \mathbf{u} \cdot \nabla Z = \nabla \cdot (\kappa \nabla Z) \tag{1.1}$$

where κ is the molecular scalar diffusivity and $\mathbf{u}$ the turbulent velocity field. Similar to the Kolmogorov scale, η, which describes the smallest scale of turbulence motions before viscosity dominates, the Batchelor scale, η_B, describes the smallest length scale of fluctuations in scalar concentration that can exist before being dominated by molecular diffusion. The phenomenology of passive scalar diffusion is dependent on the molecular Schmidt numbers, the viscosity-to-diffusivity ratio, $Sc = \nu/\kappa$. Thus, for high Schmidt number, the Batchelor scale is smaller than the Kolmogorov scale. This means that scalar dynamics can occur at scales smaller than the smallest eddy. An important theoretical result is the influence of the Schmidt number on the scalar variance spectrum behavior. For a Schmidt number larger than one, Batchelor (1959) explained thus that the classical Corrsin-Obukhov cascade associated with a $k^{-5/3}$ law (where k is the wave number) for the scalar variance spectrum (Corrsin 1951; Obukhov 1949) is followed by a viscous-convective range with a k^{-1} power law. This viscous-convective range is followed by the

† Grenoble-INP/CNRS/UJF-Grenoble 1, LEGI UMR 5519, Grenoble, F-38041, France
‡ UJF-Grenoble 1/Grenoble-INP/CNRS, LJK UMR 5224, Grenoble, F-38041, France

dissipation range, where various theoretical scalings have been proposed for the spectrum (Batchelor 1959; Kraichnan 1968). Some recent results from Donzis *et al.* (2010) seem to show that the $k^{-5/3}$ law appears only for very high Reynolds numbers. Thus, additional work is needed to systematically study the $k^{-5/3}$-k^{-1} transition in function of the Reynolds and Schmidt numbers. Moreover, direct numerical simulations (DNS) databases of scalar mixing will be useful to better understand the Schmidt number influence on subgrid-scale models in the context of large-eddy simulation (LES) of passive scalars (You & Moin 2009; Brun *et al.* 2008).

For numerical simulations, the dependance of the Batchelor scale on the Schmidt number suggests that the prediction of scalar dynamics for high Schmidt numbers is more demanding in terms of spatial resolution than the prediction of momentum. Classical advection schemes for the scalar equation impose a CFL condition which requires adapting the time step to the finest scales, adding to the computational cost. To overcome this limitation, we propose to use a remeshed particle method in order to capture all scalar scales while keeping a reasonable computational cost. In particular, we take advantage of the fact that particle methods are not constrained by CFL conditions. This technique has been introduced in Cottet *et al.* (2009) in the context of LES simulations of homogeneous isotropic turbulence (HIT) where both the momentum and scalar equations were solved by particle methods at different resolutions. Following this work, the purpose of the present study is the development and application of an efficient parallel implementation of a hybrid particle/spectral solver to perform DNS of the scalar and momentum transports on different grid resolutions reflecting the different scalar and momentum spectra.

An outline of this paper is as follows. In section 2, we first describe the remeshed particle method used for the simulation of the scalar, the coupling approach to combine a Navier-Stokes spectral solver with a particle scalar solver, and the parallel implementation of the hybrid particle/spectral method. We then show and discuss in section 3 numerical results for a wide range of Reynolds and Schmidt numbers, exhibiting spectra decay in the large, intermediate and small scales as predicted by the theory. Section 4 is devoted to concluded remarks and directions for future work.

2. Numerical method

2.1. *Remeshed particles methods*

Particle methods have long been used for the simulation of advection driven phenomena. The idea is to concentrate advected quantities on a set of particles which are advected by the local velocity field. The method is conservative by nature, and does not suffer the usual stability constraints of grid-based methods: for constant velocity fields, the solution provided by particle methods is exact, independently of the the time-step and spacing between particles.

Quantitative information can be recovered from particles by interpolating particle strengths on a grid or by using regularized version of particles. Practical and numerical analyses show that to obtain accurate results it is necessary that regularized particles satisfy an overlapping condition. This overlapping is difficult to enforce in case of large flow distortion, unless very large regularization parameters are used, something which leads to excess numerical cost.

To overcome this shortcoming of totally grid-free particle methods, one possible strategy is to remesh particles frequently on regular grid. Remeshing is done through high

order interpolation formula which conserve the moments of the particle distribution up to a desired level. Conserving up to the third moment ensures that no net dissipation results from particle remeshing. The time-scale on which particles have to be remeshed is given by the strain of the flow. A similar time-scale is used in practice to choose the time step to advance particles. This often leads to remeshing particles at every time-step, leading to forward versions of semi-Lagrangian methods. With this strategy, particle methods have been used with success in a number of cases. Comparisons of remeshed particle methods with spectral methods for the simulation of the Navier-Stokes equation in vorticity formulation have been performed to assess their accuracy (Cottet *et al.* 2002; van Rees *et al.* 2011).

A series of recent works has allowed us to interpret remeshed particle methods in terms of high order finite-difference methods. In particular the analysis in Magni & Cottet (2012) allow us to derive new remeshing formulas and give, for these formula, rigorous bounds for the time-step, in terms of the velocity strain, to obtain a desired order of convergence. It also suggests a time-splitting implementation of particle methods, where particles are successively advected and remeshed in each direction. One advantage of this approach is that it allows the use of high order remeshing formulas, with large stencils, at a reduced cost compared to traditional multi-dimensional remeshing schemes. For a stencil of 6 points in each direction, like the one we are using in this paper, the cost of remeshing is $O(18N)$ in a splitting approach instead of $O(216N)$ for a 3D remeshing formula.

The remeshing schemes and analyzed in Magni & Cottet (2012) are based on minimal stencils (5 points for a 4th order formula giving a 3rd order particle method) with local corrections to ensure consistency in zones where the local CFL numbers at particle locations cross integer values. In the present work we use instead a formula with a larger stencil of 6 points for a 4th order remeshing (Bergdorf & Koumoutsakos 2006). This formula leads to a 4th order method for constant velocity field, locally 3rd order for non-constant velocity field. We have checked in a number of experiments that it gives results similar to those obtained for the 4th order formula analyzed in Magni & Cottet (2012), for a simpler implementation and a comparable computational cost. This remeshing scheme is given by the following formula:

$$Z_i = \sum_p v_p Z_p \Lambda(x_i - x_p),$$

where $\mathbf{x}_p$ (resp $\mathbf{x}_i$) denote the particle locations (resp grid points), Z_p (resp Z_i) the scalar weights (scalar grid value) and Λ is given by

$$\Lambda(x) = \begin{cases} \frac{1}{12}\left(1-|x|\right)\left(25|x|^4 - 38|x|^3 - 3|x|^2 + 12|x| + 12\right) & \text{if } 0 < |x| < 1 \\ \frac{1}{24}\left(|x|-1\right)\left(|x|-2\right)\left(25|x|^3 - 114|x|^2 + 153|x| - 48\right) & \text{if } 1 < |x| < 2 \\ \frac{1}{24}\left(3-|x|\right)^3\left(5|x|-8\right)\left(|x|-2\right) & \text{if } 2 < |x| < 3 \\ 0 & \text{if } 3 < |x|. \end{cases} \tag{2.1}$$

The maximal time-step to be used for the remeshed particle algorithm using this remeshing kernel is given by

$$\Delta t = \frac{1}{4(\max|\nabla \mathbf{u}|)} \tag{2.2}$$

where $\max|\nabla \mathbf{u}|$ is evaluated by finite differences on the grid of the velocity components along the corresponding directions.

2.2. *Coupling a particle method with a pseudo-spectral scheme*

Remeshed particle methods easily lend themselves to coupling strategies. In our case we wish to couple the advection of a scalar by a remeshed particle method with a pseudo-spectral method to solve the Navier-Stokes equations. Both particle and spectral solvers are run in parallel, with different grid resolutions to account for the different diffusion coefficients for the scalar and momentum. Thus, the velocity field is solved by a classic pseudo-spectral solver with a second-order explicit Runge-Kutta time-advancement scheme. A classic 3/2 rule is used for dealiasing the non-linear convection term and the viscous terms are treated exactly. At the beginning of every time-step the velocity obtained from the Navier-Sokes solver is interpolated in spectral space to obtain particle velocities. These particle velocities are used to push particles along each direction (see Algorithm 1 below).

In practice the Navier-Stokes solver uses a coarser grid resolution Δx^u than the particle solver. The time step used for both the particle and spectral solver is given by the CFL condition related to the grid spacing Δx^u to ensure a CFL number not exceeding 0.5. It can be checked that this time step satisfies the condition (2.2). Depending on the ratio between Δx^u and Δx^Z this time step can be significantly larger than what would be needed if a spectral solver was used for the scalar advection.

2.3. *Parallel efficient implementation*

The challenge is to perform direct numerical simulations of high Schmidt number passive scalars advected by a fully turbulent flow. A very fine spatial resolution is therefore required. Such a resolution increases both the CPU cost and the required memory. It is mandatory to develop a parallel algorithm. It allows us to benefit from the speed-up provided by our approach and to reach higher Schmidt numbers. The solver library is designed for runs from 100 to 1000 CPUs thanks to the MPI tools. Running simulations on so many CPUs decreases of course the computational cost of the simulations and additionally increases the total amount of distributed memory. But there is still risk of memory saturation for the finest spatial resolutions. Some other memory optimizations (fine memory management, data compression, etc.) are added to facilitate the study of higher Schmidt numbers.

2.3.1. *Parallel design of the particle method*

If the remeshing facilitates the coupling of the particle method with the pseudo-spectral solver (as fields are discretized on mesh points at the start and at the end of the advection solvers), some difficulties specific to Lagrangian description have to be overcome in the parallelization work. There are two different ways to design parallel strategy. On the one hand, each processor unit could be attached to a given geometric subdomain. A particle which moves from one subdomain to another is then exchanged between the two related MPI-processes. On the other hand, the unit could be associated with a set of particles and follow them during a complete time step. A subdivision of the 3D geometry is still used to define this particle set: an MPI-process is associated with particles depending on their initial positions. With the first choice, the number of particles contained by each process could change during the advection step, while on the opposite this number is strictly constant with the second choice. As the computational cost of the particle method is directly related to the number of particles, the second choice ensures a better computation balance. At present, implementation follows this last strategy.

The simplest parallel algorithm that could be used to solve advection along one direction is described below (see algorithm 1). By using dimensional splitting it provides a

numerical solver for the 3D transport term. Of course, many optimizations are performed

Algorithm 1 Advection of a particle line.

1: **for** each (1D-) line along the current direction **do**
2: Create a particle on each mesh point and initialize its position: $p_n^i = i \cdot dt$.
3: Compute the middle positions for each particle: $p_{n+1/2}^i = p_n^i + \frac{dt}{2} v_n^i$.
4: Compute the integer truncation l_n^i of $\left(p_{n+1/2}^i\right)/dx$.
5: **for** $m = 0$ to 1 **do**
6: Compute the rank $R_m(i,n)$ of the process which stored the velocity on node $I_n^i + m$.
7: **if** my MPI-rank is different from $R_m(i,n)$ **then**
8: Send a MPI message to process R_m to ask for the velocity on node $I_n^i + m$.
9: Receive the velocity on node $I_n^i + m$ from process $R_m(i,n)$.
10: **end if**
11: **end for**
12: Interpolate the velocity on position $p_{n+1/2}^i$ from velocity on node I_n^i and $(I_n^i + 1)$ $\{\text{as } p_{n+1/2}^i \in \left[I_n^i; I_n^i + 1\right]\}$.
13: Compute the final particles positions: $p_{n+1}^i = p_n^i + \frac{dt}{2} v_n(p_{n+1/2}^i)$.
14: Remesh all the line inside a buffer and redistribute it to the concerned processes.
15: **end for**

in order to improve performance and the parallel scalability. It is well known that it is generally interesting to limit communication numbers by gathering them (if possible). Also, the final algorithm does not include the "for" loop of line 1 but computes particle position and remeshing directly on a group of such lines. In this way, the needed communications for each line are performed together. In the same spirit, the communications performed on line 8 (respectively on line 9) inside the loop on m are gathered. Finally, each time step requires only four communications for each MPI-process. Two communications are performed to interpolate the velocity: a map of the velocity field to exchange between processes (see line 8 of the algorithm) and the exchange of the velocity field itself (line 9). Two communications are done to remesh particles (see line 14), one to map the buffer redistribution and another to send part of this buffer.

The amount of data corresponding to the two mapping processes is also reduced thanks to mathematical results (considering stability conditions, particles remain in the same order and follow each other) and using compression tools (a naive description of the geometry described by the map involves many redundancies). Finally some tricks allow us to achieve these four communications as four *point-to-point* communications.There is no need of any synchronization and it allows us to recover communication by computation thanks to non-blocking protocol. Since MPI provides a better management of its internal buffer for the *point-to-point* communication, memory needs are also diminished.

2.3.2. *Parallel design of the pseudo-spectral solver and of the coupling*

The pseudo-spectral solver uses a classical parallel design based on two-dimensional subdivisons. The 3D fast Fourier transform (FFT) is split in three mono-dimensional fast Fourier transforms. These 1D FFT are based on 1D summations. Thus the parallelization is done by subdividing the 3D geometry along only 2 directions: along Y and Z in the real space and along X and Y on the spectral space. For instance, consider a run on n^2 MPI-processes of a simulation performed with a resolution of N_x, N_y and N_z grid points along

Number of CPUs	Time per iteration on Intel	Time/iteration on Blue-Gene/P
128	55s	-
256	27.5s	56.4s
512	14.4s	30.0s
1024	-	15.5s

TABLE 1. Setup and iteration time for different number of CPUs

X-axis, Y-axis and Z-axis, respectively. Then the subdomain of each process contains N_x, N_Y/n and N_z/n grid points along X, Y and Z. The 3D FFT is performed by algorithm 2. A similar algorithm is used to perform the inverse transform. Note that computations

Algorithm 2 Parallel fast Fourier transform.

1: Compute 1D FFT along X {as there is not domain subdivision along X, no communication is needed}.
2: Perform communication in order to re-organize data in order to only have domain subdivision along X and Z.
3: Perform 1D FFT along Y {no communication is needed}.
4: Perform communication in order to re-organize data in order to only have domain subdivision along X and Y.
5: Perform 1D FFT along Z {no communication is needed}.

on real and spectral spaces do not require any communication. All the communications are performed during the FFT and the inverse-FFT and the key point is to obtain an efficient algorithm for these three-dimensional transforms. Mono-dimensionnal FFT are performed thanks to the classical *FFTW* library. In order to achieve the coupling between the two solvers, the domain subdivision used for the particle method implementation is the same the one used inside the pseudo-spectral library in the real space. As the particle solver deals only with real fields, data-reorganization (and thus communication) are avoided in the interface of the two solvers. This should enable easier coupling of the particle solver with general flow solvers based on various numerical methods.

The parallel scalability has been tested on different architectures. Table 1 presents results on a Blue-Gene/P cluster ("Babel" from the French institute IDRIS) and on Intel-based "Certainty" cluster (quad-cores with Intel Westmere-EP processors). The scalar grid size is 1024^3 points and the velocity is discretized on a 128^3 grid in both tests.

3. Results

3.1. *Flow configuration and numerical set-up*

In order to numerically observe the Schmidt and Reynolds number influence on the mixing scalar dynamics, various simulations have been performed in the context of homogenous isotropic turbulence (HIT), in a 3D periodic box with a length 2π. The velocity field is solved by using a classic pseudo-spectral code, whereas the scalar field is solved using particles method, as already described. A statistical steady flow is achieved by using a

Nx^u	R_λ	$\Delta^z_{\text{part}}t$	Δ^u_t	Nx^z	Sc	$\Delta^z_{\text{spec}}t$
128	80	2.0e-1	6.2e-2	128	0.7	6.2e-2
				512	8	1.6e-2
				512	16	1.1e-2
256	130	4.3e-2	1.2e-2	256	0.7	1.2e-2
				768	4	5.8e-3
				1024	8	4.0e-3
512	160	1.0e-2	3.0e-3	512	0.7	6.2e-2
				1024	4	3.1e-2

TABLE 2. Setup and time step for different simulations performed

forcing term (Alvelius 1999). The scalar field is initialized between 0 and 1 according to the procedure proposed by Eswaran & Pope (1988). To achieve a steady state for the scalar, a forcing scheme is also applied to low wave number modes in Fourier space. For the velocity field, resolutions from 128^3 to 512^3 grid points are used to allow simulations with Taylor-scale Reynolds number, R_λ, varying from 80 to 160. For the scalar field, resolutions up to 1024^3 grid points are used to obtain results for a Schmidt number until 16. All the simulations parameters are reported in Table 2. Further simulations were performed with refinement strategies to ensure that the simulations reported in this work are indeed converged results. This table also reports the imposed time-step to solve the velocity field with the classic pseudo-spectral approach, Δ^u_t, and the imposed time-step to solve the scalar field with particles method, $\Delta^z_{\text{part}}t$. The theoretical time-step for a scalar field solved by spectral method, $\Delta^z_{\text{spec}}t$, is also reported for comparison. As expected, the time-step for the particle method is much larger than for the spectral method at the same resolution. This efficiency has allowed the performance of all these simulations during the CTR program on the "Certainty" cluster (quad-core with Intel Westmere-EP processors).

3.2. *Influence of Reynolds and Schmidt numbers on the scalar variance spectrum*

The behaviors of the scalar variance spectrum are now studied at large, intermediate and small scales from this DNS database and compared with theoretical predictions (for a review see Lesieur 2008). First, at the largest scales, the classical Corrsin-Oboukhov cascade is expected to characterize the inertial-convective range. Similarly to the inertial range of the kinetic energy spectrum, it is expected that this cascade follows a $k^{-5/3}$ law, with k the wave number (Obukhov 1949; Corrsin 1951). Figure 1 (a) confirms this first cascade. In this figure the spectra are compensated by $k^{5/3}$ and the wave numbers are multiplied by the Taylor scale, λ. As expected, the results show that this range is independent of the Schmidt number, with a self-similarity of this part of the spectra for all simulations at the same Reynolds number. This range also occurs for the kinetic energy (not shown) and increases with the Reynolds number. At $R_\lambda = 80$, this range does not appear clearly because the flow is not fully turbulent. Note that the Taylor scale appears as a good boundary for the end of this range. After this range, for Schmidt

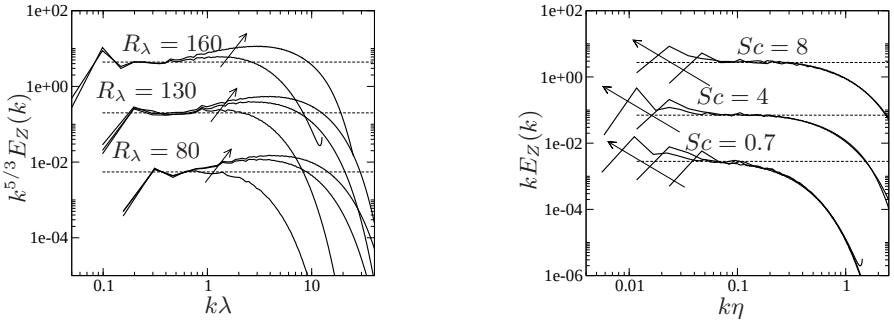

FIGURE 1. Spectrum shape at large and intermediate scales. (a) Scalar variance spectrum premultiplied by $k^{5/3}$. The wave numbers are premultiplied by the Taylor micro-scale, λ. Arrows show direction of the increasing Schmidt number, Sc. (b) Scalar variance spectrum premultiplied by k. The wave numbers are premultiplied by the Kolmogorov scale, η. Arrows show direction of the increasing Reynolds number, R_λ. Spectra are shifted for easier reading.

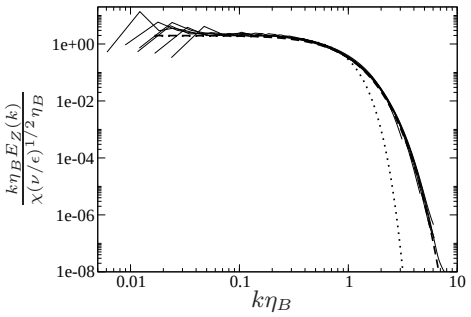

FIGURE 2. Normalized scalar variance spectra for all runs with a Schmidt number larger than unity. The results are compared with the scalings coming from Batchelor (1959) (dotted line) and from Kraichnan (1968) (dashed line).

numbers higher than unity, Batchelor (1959) explained the development of the viscous-convective range with a k^{-1} law. This scaling is due to the velocity small scales strain effect on the scalar field. Figure 1 (b) confirms this behavior. In this figure the spectra are compensated by k and the wave numbers are multiplied by the Kolmogorov scale, η. As expected, the k^{-1} law does not appear for Schmidt numbers smaller than 1, whereas the viscous-convective range increases with the Schmidt number. Note that this wave range appears as independent of the Reynolds number, as expected from theory. These results are consistent with the theoretical prediction of the scalar variance spectrum. Other previous numerical works (Donzis *et al.* 2010) seem to find the $k^{-5/3}$ Corrsin-Oboukhov cascade only at a very high Reynolds number. The scalar forcing schemes can explain the discrepancy of these results and future works will be devoted to better understand the forcing influence on the scalar variance spectrum.

Finally, the form of the scalar variance spectrum in the dissipation range (following

the viscous-convective range) is studied. Two distinct theoretical behaviors have been proposed. First, Batchelor (1959) proposed the following form:

$$\frac{k\eta_B E_Z(k)}{\chi(\nu/\epsilon)^{1/2}\eta_B} = \alpha \exp\left(-\alpha(k\eta_B)^2\right),$$

where α is a coefficient, χ the scalar variance dissipation rate and ϵ the kinetic energy dissipation rate. From Kraichnan's work (Kraichnan 1968), the following alternative form has been proposed:

$$\frac{k\eta_B E_Z(k)}{\chi(\nu/\epsilon)^{1/2}\eta_B} = \alpha\left(1+\beta k\eta_B\right)\exp\left(-\beta k\eta_B\right),$$

where β is another coefficient (see Bogucki *et al.* (1997) for details). Note that this question has some practical implications for better understanding energy transfer between ocean and atmosphere. Indeed, the turbulent kinetic energy dissipation rate is needed to evaluate turbulent fluxes controlling this transfer. This dissipation rate can be obtained from the temperature dissipation spectra by using a given form for the scalar variance spectrum in the dissipation range (Bogucki *et al.* 2012). Figure 2 shows the normalized scalar variance spectra for all runs with a Schmidt number larger than unity. Both proposed spectrum forms are also shown. These DNS results clearly show a good agreement with the Kraichnan form and confirm previous works (Bogucki *et al.* 1997, 2012). The systematic deviations from the Batchelor form imply that erroneous results can be expected if this form is used to compute the turbulent kinetic energy dissipation rate from temperature dissipation spectra in oceanography applications.

4. Conclusion

Direct numerical simulation (DNS) of high Schmidt number passive scalars in highly turbulent flows are challenging. Indeed, high Schmidt number scalars develop mixing scales smaller to the Kolmogorov scales. This means that a finer resolution has to be used to solve scalar transport equation than to solve Navier-Stokes equation. An approach coupling CFL-free remeshed particle solver for scalars with pseudo-spectral solver for velocity fields has been developed. This approach is found to be an efficient way to perform DNS at a reasonable cost. Extensive DNS have thus been performed to study the scalar variance spectrum behavior as a function of Schmidt and Reynolds numbers. Theoretical results have been recovered with a transition from the $k^{-5/3}$ law to the k^{-1} law for sufficiently high Schmidt numbers. These DNS results are also in good agreement with the Kraichnan spectrum shape in the dissipation region. Future works will be devoted to a better understanding of the scalar forcing influence on these spectrum behaviors. We also plan to couple the remeshed particle method with other solvers (as finite-volume solver) to be able to use this approach in more general flow configurations.

Acknowledgments

This work is supported by the Agence Nationale pour la Recherche (ANR) under Contracts No. ANR-2010-JCJC-091601 and ANR-2010-COSI-0009. G.-H. C. is also grateful for the support from Institut Universitaire de France. The authors have benefited from fruitful discussions with CTR summer program participants. In particular, they are grateful to J. A. Domaradzki for discussion about the scalar spectrum forms.

REFERENCES

Alvelius, K. 1999 Random forcing of three-dimensional homogeneous turbulence. *Phys. Fluids* **11**, 1880–1889.

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** (01), 113–133.

Bergdorf, M. & Koumoutsakos, P. K. 2006 A Lagrangian particle-wavelet method. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal* **5**, 980–995.

Bogucki, D., Domaradzki, A. J. & Yeung, P. K. 1997 Direct numerical simulations of passive scalars with Pr¿1 advected by turbulent flow. *J. Fluid Mech.* **343**, 111–130.

Bogucki, D., Luo, H. & Domaradzki, A. J. 2012 Experimental evidence of the Kraichnan scalar spectrum at high reynolds numbers. *J. Phys. Ocean* In press.

Brun, C., Balarac, G., da Silva, C. B. & Métais, O. 2008 Effects of molecular diffusion on the subgrid-scale modelling of passive scalars. *Phys. Fluids* **20** (2), 025102.

Corrsin, S. 1951 On the spectrum of isotropic temperature fluctuations in an isotropic turbulence. *J. Appl. Phys.* **22**, 469–473.

Cottet, G.-H., Balarac, G. & Coquerelle, M. 2009 Subgrid particle resolution for the turbulent transport of a passive scalar. In *Advances in Turbulence XII, Proceedings of the 12th EUROMECH European Turbulence Conference, September, 2009* (ed. B. Eckhardt), *Springer Proceedings in Physics*, vol. 132, pp. 779–782. Marburg, Allemagne: Springer.

Cottet, G.-H., Michaux, B., Ossia, S. & Vanderlinden, G. 2002 A comparison of spectral and vortex methods in three-dimensional incompressible flows. *J. Comput. Phys.* **175** (2), 702–712.

Donzis, D. A., Sreenivasan, K. R. & Yeung, P. K. 2010 The batchelor spectrum for mixing of passive scalars in isotropic turbulence. *Flow, Turbulence and Combustion* **85**, 549–566.

Eswaran, V. & Pope, S. 1988 Direct numerical simulations of the turbulent mixing of a passive scalar. *Phys. Fluids* **31**, 506–520.

Kraichnan, R. 1968 Small-scale structure of a scalar field convected by turbulence. *Phys. Fluids* **11**, 945–953.

Lesieur, M. 2008 *Turbulence in fluids*. Springer, Dordrecht.

Magni, A. & Cottet, G. 2012 Accurate, non-oscillatory, remeshing schemes for particle methods. *Journal of Computational Physics* **231** (1), 152 – 172.

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

van Rees, W. M., Leonard, A., Pullin, D. I. & Koumoutsakos, P. K. 2011 A comparison of vortex and pseudo-spectral methods for the simulation of periodic vortical flows at high reynolds numbers. *J. of Computational Physics* **230**, 2794–2805.

You, D. & Moin, P. 2009 A dynamic global-coefcient subgrid-scale model for large-eddy simulation of turbulent scalar transport in complex geometries. *Phys. Fluids* **21**, 045109.