Large Eddy Simulation's Parallel Processing Performance for Urban Air Pollution Studies Using FLUENT

Salim Mohamed Salim

***Abstract*—The paper shares the experience of executing enormous computations by Large Eddy Simulation (LES) for environmental studies, using the commercial Computational Fluid Dynamics (CFD) code FLUENT 6.3® on an Intel Xeon® quad-core workstation and Intel Core® dual-core PC. The computational performancein terms of iteration resource and time expenditure for both single and parallel processingare assessed, with the later proving more efficient for large computational domainswhich arenecessary for simulations where both external and internal fluctuations dominate such in the case of flow and pollutant dispersion within urban street canyons.**

***Index Terms*—CFD, LES, Parallel Processing.**

I. Introduction

Air quality in urban and industrial complexes has garnered great interest because of numerous implications on human health, pedestrian comfort and environmental concerns. There is need to understand the mechanism of pollutant dispersion in urban street canyon and its ramifications on environmental and structural engineering practices. This has resulted in continuous development of new simulation methods, improvement of existing modelling techniques and best practices in order to assist regulators, policy makers, architects and urban planners to mitigate air pollution issues in their cities.

This has motivated a number of field, experimental and numerical investigations to access the interaction of buildings, trees, moving vehicles and other large obstacles with the atmospheric boundary layer in order comprehend the resulting pollutant accumulation and/or dissipation patterns within urban and industrial complexes. Computational Fluid Dynamics (CFD) has become the preferred method of investigation at the micro scale level[1] due to the ever increasing availability of computer resources to researchers.

However, majority of previous CFD studies have employed the conventional Reynolds-averaged Navier-Stokes (RANS) approach, which although performed qualitatively well, poorly predicted the pollutant concentration levels and distribution when validated against wind tunnel (WT) experiments [2, 3]. RANS equations are widely used and favoured in industry because they can be approximated to determine averaged solutions to the Navier-Stokes equations, which are sufficient for most engineering problems without the need to resolve for each and every detail contained within the entire range of eddy length scales[4]. This makes them very attractive as a result of their relatively low computational demand, allowing mean solutions to be obtained rapidly and provides flexibility in performing numerous simulations with varying parameters allowing for design optimization.

Unfortunately, it has been deduced that in order to obtain reliable and accurate predictions of the developing flow and pollutant concentration fields around individual buildings and in urban street canyons, it is necessary to resolve both the internally and externally induced fluctuations so as to account for the turbulent mixing process by which the dispersion of airborne pollutants depend on. This implies that the RANS techniques are unsuitable for environmental air pollution applications and Large Eddy Simulations (LES) need to be used instead [5,6]. The same can be said for any flow phenomena where intermittent unsteadiness is a prominent feature of the resulting velocity and pressure field.

The drawback and general lack of appeal of LESthough, particularly in the industry, is that they require much larger computational resources in terms of RAM and processing time (ranging between ten to even a hundred times that of RANS for the same computational grid and boundary conditions) and there is a broad belief amongresearcher and practitioners that such simulations require High Performance Computer (HPC) clusters, which many research labs budget might not permit.

This has discouraged many commercial users such as consultants in implementing LES for the prediction of airflow and pollutant dispersion for environmental impact studies, for example, with many citing the high computational time expense as the major setback. Therefore,the present study aims to demonstrate the speed up in physical simulation time obtainable using parallel processing on conventional PCs for large computational domains, in order to motivate the employment of high fidelity LES instead of relying on the steady-state RANS models which do not provide accurate results.

II. Computational Set-up

A. Domain and Boundary Conditions

LES simulations are performed with the aim of reproducing the wind tunnel (WT) experiment by Gromke and Ruck [7] available on the online data base www.codasc.de. A summary of the computational domain and implemented boundary conditions are illustrated in Fig. 1.

The dynamic Smagorinsky-Lily Sub-grid Scale model is selected. Bounded central differencing scheme for momentum, 2^{nd}order time-advancement and 2^{nd} order upwind for energy and species transport equations are chosen. *PRESTO* and *SIMPLEC* are employed for pressure and

Manuscript received May 10, 2012; revised September 7, 2012.

Salim Mohamed Salim is with the School of Engineering, Taylor's University, Subang Jaya, Selangor, Malaysia. (e-mail: salim@alumni.nottingham.ac.uk).

pressure-velocity coupling, respectively. Convergences at 1 x 10^{-3} for the scaled residual are set with a dimensionless time-step of 2.5 x 10^{-3}. The advection-diffusion method is employed for modelling the dispersion of pollutants species.

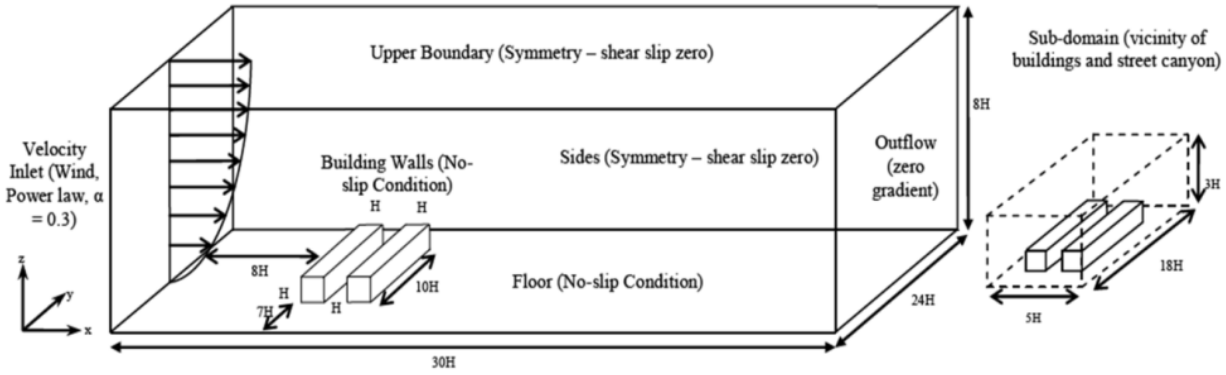

Fig. 1. Computational domain and boundary conditions for the CFD simulation setup

B. Spatial and Temporal Resolution

Three meshes are generated to assess the grid dependence and are summarized in Table I. The domain is discretized using hexahedral elements incorporating recommendations based on the wall y^+ approach as described by Salim et al. [8].Approximately half of the total cells are placed in the sub-domain defining the vicinity of the buildings and street canyon (see Fig. 1 – with a volume of 250 H^3), where majority of the flow separation, recirculation and reattachment occurs with steep gradients in the flow variables.

The Reynolds number of the main flow is Re_H= 50,000 therefore a y^+> 30 is selected for the meshes in order to resolve the log-law region because wall functions work best for the employed turbulence models. Cell stretching (i.e. use of successive ratio) particularly within the sub-domain is avoided since no homogeneous direction exists in the flow, and therefore, equal spatial resolution is imposed in all directions.

In addition, a time resolution, Δt/Tb(with Δtthe time-step size and Tb = H/Ub) sensitivity is performed with different time-step sizes, Δt, maintaining the same number of flow-through time, Tf and turnover time, tc. This is summarized in Table II. Following the work of Cai et al. [9], the turnover time of the primary circulation, tc, in the canyon is of the order of

$$t_c = \frac{W + H}{U_c}$$

where $U_c$is the velocity scale of the mean wind in the canyon which is found to be ~ 0.14 ms^{-1} resulting in a value of t_c = 2 s. Initially, the simulation is performed for 33 flow-through time (T_f = L/U_b with L being the streamwise length of the domain and $U_b$the bulk velocity) corresponding to 10 t_c. Statistically steady-state is achieved at this point. The flow statistics are then reset, and the simulation is performed for a further 33 flow-through time (= 10 t_c) to ensure that the final time-averaged results are independent of the initial conditions.

TABLE I: SPATIAL RESOLUTION

Mesh identity	Minimum grid spacing	Cell count in canyon	Total cell count
Mesh A (Coarse)	$\Delta x = \Delta y = \Delta z = 0.1\ H$	250,000	538,000
Mesh B (Fine)	$\Delta x = \Delta y = \Delta z = 0.077\ H$	598,246	1,111,246
Mesh C (Finest)	$\Delta x = \Delta y = \Delta z = 0.067\ H$	728,568	1,569,242

TABLE II: TEMPORAL RESOLUTION

Temporal resolution $\Delta t/T_b$	Study performed on corresponding mesh	Total number of dimensionless time-steps
1/4	Mesh A	8,000
1/8	Mesh A & Mesh B	16,000
1/16	Mesh B & Mesh C	32,000

III. OBSERVATIONS

Initially, an Intel Core® dual-corePC was used for the investigation, and when the larger computational grid domains were used (i.e. Mesh C with over one and a half million cells) the PC reported a memory error. In order to circumvent the restriction, two nodes parallel processing was activated, effectively dividing the task overtwo CPU processors of the dual-core PC thus allowing the simulation to proceed.

The efficiency in this context is quantified as the physical duration required for the computer to compute each iteration.

This motivated a further study to determine the consequence of activating parallel processing over additional nodes on a higher end Intel Xeon ® quad-core workstation (which has 4 CPUunlike the Intel Core® dual-core whichcomes with 2 CPU), and it was observed that the total duration required with settings and conditions remaining unchanged was reduced by almost half when four nodes were used compared to just two nodes, in other words improving efficiency twofold. However, the efficiency was reduced by half when the simulation task was executed over eight nodes.

The duration required for each iteration was higher for Mesh C compared to Mesh B due to the larger number of computational cells, with the task divided over equal number of nodes and boundary condition and simulation settings remaining similar.

An examination was then performed for Mesh B on the lower PC configuration (i.e. the Intel Core® dual-core PC), and it was deduced that two nodes parallel processing improved the performance in comparison to single processing, but four nodes parallel processing reduced the efficiency. However, when comparing the overall performance of the lower PC configuration, it was observed to be slower than the Intel Xeon® quad-core workstation, indicating the dependency on CPU processor speed as an important factor too.

Finally, a separate study was carried out for a smaller computational grid (Mesh A with approximately half a million cells) and interestingly, the computation efficiency was reduced when parallel processing was employed on both the PC and workstation, instead of running the simulation in single processing.

IV. Conclusion

The findings suggest that the efficiency of parallel processing in the case of FLUENT® was a combined function of the defined flow problem's computational cell count, and the employed computer's microprocessor, number of nodes utilized and CPU configuration. For example, it might be more efficient to use a workstation in parallel processing over four CPU nodes but with a better microprocessor speed compared to running the simulation in eight nodes on a HPC cluster with a lower microprocessor speed for a computational grid made up of about a million cells.

However, this might not be the case if the cells were increased to over two million. There is a need to further investigate this in order to quantify the dependence of each factor, in order to provide guidelines to researchers and CFD practitioners on the best practices on optimizing computation performance when considering parallel processing to execute a given simulation task.

References

[1] R. E. Britter and S. R. Hanna, "Flow and dispersion in urban areas," *Annual Review of Fluid Mechanics.* vol. 35, pp. 1817 – 1831, 2003.

[2] R. Buccolieri, S. M. Salim, L. S. Leo, S. D. Sabatino, A. Chan, P. Ielpo, G. D. Gennaro, and C. Gromke, "Analysis of local scale tree-atmosphere interaction on pollutant concentration in idealized street canyons and application to a real urban junction," *Atmospheric Environment*. vol. 45, pp.1702 – 1713, 2011.

[3] K.-J. Hsieh, F.-S. Lien, and E. Yee, "Numerical modeling of passive scalar dispersion in an urban canopy layer," *Journal of Wind Engineering and Industrial Aerodynamics*. vol. 95, pp. 1611-36, 2007.

[4] H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, 2nd ed. Pearson Education Limited, Harlow.2007.

[5] S. M. Salim, R. Buccolieri, A. Chan, and S. D. Sabatino, "Numerical simulation of atmospheric pollutant dispersion in an urban street canyon: comparison between RANS and LES," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 99, pp. 103 – 113, 2011.

[6] S. M. Salim, A. Chan, and S. C. Cheah, "Numerical simulation of atmospheric pollutant dispersion in tree-lined street canyons: comparison between RANS and LES," *Journal of Building and Environment*, vol. 46, pp. 1735-1746, 2011.

[7] C. Gromke, R. Buccolieri, S. D. Sabatino, and B. Ruck, "Dispersion study in a street canyon with tree planting by means of wind tunnel and numerical investigation – evaluation of CFD Data with experimental data," *Atmospheric Environment*, vol. 42, pp. 8640-8650, 2008.

[8] S. M. Salim, M. Ariff, and S. C. Cheah, "Wall y^+ approach for dealing with turbulent flows over a wall mounted cube," *Progress in Computational Fluid Dynamics*, vol. 10, pp. 1206 – 1211, 2010.

[9] X. M. Cai, J. F. Barlow, and S. E. Belcher, "Dispersion and transfer of passive scalars in and above street canyons--Large-eddy simulations," *Atmospheric Environment*, vol. 42, pp. 5885 – 5895, 2008.