ISSN No. 0976-5697

Volume 3, No. 3, May-June 2012

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Benchmarking of High Performance Cluster Reynolds

Sandip Sarkar*
Research and Development
Tata SteelJamshedpur, India
sandip.sarkar@tatasteel.com, thesandipsarkar3@gmail.com

Raghavendra Krishnamurthy
Research and Development
Tata SteelJamshedpur, India
r.krishnamurthy@tatasteel.com

Satish Kumar Ajmani
Research and Development
Tata SteelJamshedpur, India
skajmani@tatasteel.com

Abstract:This paper demonstrates a complete benchmarking exercise of a high performance cluster Reynolds at Tata Steel R&D division. A complete bench script has been reported for running parallel programs in queue and that can be directly used with all parallel computations purposes. Randomness in node distribution is found in batch processes. The scalability of the computations on a 80-processor SUSE Linux cluster is evaluated for smaller to large size problems. It is found that for a case with 0.417 million cells, after 32 processors performance curve flattens and then reduces marginally. This is due of partitioning the case into too many cores, which increases the communication between the cores and the scalability does not remain linear. For all other cases a super-liner speed up in computational efficiency and system functional rating is observed. System scalability study is performed for a live industrial case with OpenFOAM. For such case after 40 numbers of processors, reduction in speedup is observed, whereas there is a sharp reduction in clock time till 80 processors. Reynolds performance over the available supercomputing benchmarks is compared. A new parameter *Reynolds performance index* is defined to compare cluster's performance over other high performance machines. Depending upon the problem complexities and mesh sizes, recommendations have been given to limit the optimum node usage. This is for the first time such exhaustive study in benchmarking exercise being reported

Keywords:Reynolds, HPC, benchmarking, rating, efficiency

I. INTRODUCTION

Global competition in the steel industry has precipitated the need of increasing production by improving quality, safety and reducing production cost. This can be successfully accomplished by effective use of advanced computer simulation techniques. An integrated steel plant like Tata Steel is rapidly moving towards a leading-edge parallel computing techniques to support extensive use of visualization for the steel making processes. [1] To meet the demand of high speed computational facilities, at the Research and Development Division at Tata Steel Jamshedpur a new high performance computing server *Reynolds* has been commissioned recently. The main aim of the system is to meet day to day modeling and other computer simulation exercises pertaining to the plant problems [2]. One of the important aspects is aimed to give a readymade industrial solution. Depending upon the types of job and its computational and execution time *Reynolds* behaves differently [3].

This situation corresponds to different types of plant related problems [4]. In such cases, it is impossible to estimate the performance of a computational server by seeing the specifications provided by the vendor [5]. Therefore a test is necessary to evaluate the performance of a system and its inter node communications [6]. The process of doing such an activity is called benchmarking of a computational cluster [7]. By definition, benchmarking is the act of running a computer program, a set of programs, or other operations, in order to assess the relative performance of an object, normally by running a number of standard tests and trials by it [8]. Benchmarking is usually associated with assessing performance characteristics of computer hardware, for example, the floating point operation performance of a CPU. Process involved in benchmarking of a server includes submitting a job to the machine for different loading schedules [9]. For different schedules several benchmarking parameters are evaluated and performance characteristics are analyzed [10]. Those jobs are named as benchmarks and they are specifically designed to mimic a particular type of workload on a component or system [11]. Such benchmarking types are assembled in several categories, starting from system user based modules to industry standard computer benchmarks [12]. For most of the system user interfaces parallel benchmarks with multiple processors are of typical standard [13]. For the case with *Reynolds*, only parallel benchmarks methodology has been adopted [14].

In this paper, we have documented nuances of high performance cluster (HPC) benchmarking processes from the scratch. Staring from the system specifications, batch run in parallel processes are described exhaustively. Parallel Batch Scheduler is explained by showing sample batch scripts. This can be directly adopted for all high performance cluster batch processes. Several benchmarking issues are addressed specifically. We calculated system performance parameters for different standard benchmarking test problems reflecting the attributes of real industrial applications. System performance evaluations in batch processes are examined with increasing numbers of governing equations and associated numbers of finite volume cells. Results are discussed for different situations and corresponding critical analysis have been demonstrated. Systems superiority is compared with the available supercomputing benchmarks. In addition to the above, system performance is computed with real industrial 4-

strand tundish case. Few results of fluid flow simulations are presented and analyzed accordingly. Recommendations have been given at the end to limit the node usage depending upon the problem complexities and mesh sizes. To the best of author's knowledge this is for the first time such exhaustive study on server benchmarking being reported.

II. REYNOLDS ARCHITECTURE

High Performance Computing Cluster *Reynolds* has been built by SGI and having a series number of Altix 1300. The *Reynolds* cluster has a distributed memory system as opposed to a shared memory system like that used in the high-performance computer servers. Instead of passing pointers into a shared virtual address space, parallel processes in an application pass messages and each process has its own dedicated processor and address space.

There are three primary hardware component types in the *Reynolds* cluster:

i. Head node
ii. Compute nodes
iii. Network interconnect components (Gigabit Ethernet switches, InfiniBand switches, PCI cards, and cables)

The head node is connected to the interconnect network and also to the "outside world", typically via the local area network (LAN). The head node is the point of submittal for all *MPI* (Message Passing Interface) application runs in the cluster. An *MPI* job is started from the head node and the sub-processes are distributed to the cluster compute nodes from the head node. The main process on the head node will wait for the sub-processes to finish. For large clusters or clusters that run many *MPI* jobs, multiple head nodes may be used to distribute the load. *Reynolds* have 2x Dual core processors (Intel-Wolfdale Processor, 3.4 GHz, FSB 1600MHz, L2 Cache 6 MB per dual core processor), 32GB DDR RAM, 1 TB SAS hard disc, DVD/CD read/write drive, CPU Architecture 64 bit and is compatible with 32 bit, base cluster management console for system administrative purposes.

The compute nodes are identical computing systems that run the primary processes of *MPI* applications. These compute nodes are connected to each other through the interconnect network. *Reynolds* is associated with 20 computing nodes having factory integrated and tested, each node with 2x dual core processors (Intel Wolfdale Processor, 3.4 GHz, FSB 1600MHz, L2 Cache 6 MB per dual core processor), 16 GB DDR RAM, expandable to 32 GB, 146 GB SAS hard disc, CPU Architecture 64 bit and is compatible with 32 bit.

The network interconnect components are typically Gigabit Ethernet or InfiniBand. InfiniBand is an open standard for inter-processor and storage communication. It builds on the lessons learned from Ethernet, while incorporating features useful to enterprise-class computing and storage networks. It provides a flexible, manageable, high-speed communication infrastructure. The *MPI* messages are passed across this network between the processes. This compute node network does not connect directly to the "outside world" because mixing external and internal cluster network traffic could impact application performance. Intra node communications during parallel run use InfiniBand network. For communications from remote use Tata Steel network.

In addition to the above, *Reynolds* has got a usable data storage capacity of 1.5 TB with SAS Hard Disc Drives, Raid Level 0, 1, 5, Cache size 1 GB, controller supports 4 Gbps FC ports or 3 Gbps SAS ports. This bank can be accessed by the cluster.

Fig. 1 eludes a basic Gigabit Ethernet configuration using a single Ethernet switch for node-to-node communication.

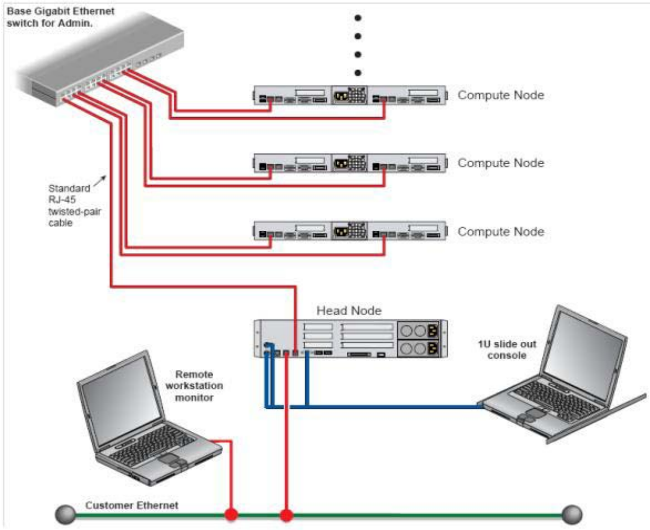


Figure 1. Gigabit Ethernet configuration using a single Ethernet switch for node-to-node communication (taken after permission from SGI altix, USA, SGI altix 1300 manual)

III. BENCHMARKING JOBS

Benchmarking jobs are essentially several workload codes designed for specific purposes only. Several benchmarking suits are available for characterizing system performance and evaluation. Among them the most suitable and reliable benchmarking codes are ANSYS Fluent benchmarking cases. They are well tested and have been used successfully in several high performance computational servers e.g. IBM, SUN X6250 etc.

Stating from ANSYS Fluent benchmarking cases [15], Reynolds performance was evaluated with a real industrial case running at Tata Steel plant. Depending upon the numbers of finite volume cells and fluid flow/heat transfer model used for computations, case types are categorized. Following's were the case types and its categorizations:

A. *ANSYS Benchmarking Cases:*

a. ***Small case:*** This case consists of two sub cases,
 a) # EDDY_417K
 b) # TURBO_500K

b. ***Medium case:*** This case consists of two sub cases,
 a) # AIRCRAFT_2M
 b) # SEDAN_4M

c. ***Large case:*** This case consists of two sub cases,
 a) # TRUCK_14M
 b) # TRUCK_POLY_14M

B. *A real industrial case running at Tata Steel plant:*

a. # Four_Strand_Tundish

IV. BENCHMARKING SUBMITTAL PROCESS SETUP WITH MPI

According to the different association degree, high performance calculation cluster can be divided into two kinds: First, ***Task Piece way:*** Separate calculation task into the task piece then distribute the task piece to each node, finally produce an eventual output by gathering all results which each node respectively calculate. Second, ***Parallel Calculation way***: Exchange large data between node in the course of calculation and calculate some strong couple relation data. Relying on super calculation cluster software, achieve the calculated task which large-scale servers usually do by fitting together several personal computers.

Benchmarking processes are set based on the Message Passing Interface (*MPI*). As per *MPI*, which is a kind of programming model database news can transmit, and becomes the representative of programming model. This is a kind of standard representative but doesn't specify certain realization. *MPI* that has not only various advantages such as good transplantation ability, powerful function, high efficiency and so on but also different free, efficient, practical realization edition, applies to the news-transmitting model based on the parallel computer system which has distributed memory, moreover, nearly all the manufacturers of parallel computer offer the support of technology, which other parallel environment does not compare.

MPI which was developed in 1994 became *MPI*-2 today defines the following standards：

a. *MPI* database can be used by Fortran and C, processor grammars can compile all application rules of function or process database. It has no difference with general function or process and ensures transplant ability that *MPI* program complied special standard scan run on any platform.
b. *MPI* database realization is offered by hardware manufacture, producing optimum edition suiting each hardware.
c. Support expands I/O, dynamic process, unilateral communication, unblocked crowd communication pattern.

Reynolds has got three *MPI* systems, *OPEN MPI*, *SGI MPI* and *HP MPI*. Depending upon the compiler compatibility one of these *MPI*'s are selected automatically.

For setting up benchmark programs with *MPI*, a system batch user module script has to be written to distribute the functionalities of *MPI* run. Such type of script is known as ***qsub***. So, by definition, ***qsub*** is a shell script that controls node distribution during batch processes.

After setting up *MPI* batch scheduler we made special ***qsub*** script file for running the benchmark. This ***qsub*** file is a bench script file which couples with batch scheduler *MPI* for allocation of the processors. The allocation is fixed as per *MPI* host schedule.

The structure of ***qsub*** file is as follows:

```
#!/bin/sh
#! This is an example of PBS script file for ANSYS FLUENT.
#PBS -S /bin/sh
#! specifies number of CPUs (ncpus) used in PBS
#PBS -l nodes=18:ppn=4
#PBS -e PBS_BATCH.err
#! specifies number of CPUs for parallel computing/number of open MPI threads
nCPU=72
#! specifies the version of ANSYS FLUENT (2d: 2-Dimension; 3d: 3-Dimension)
#! version=3ddp
#! specifies working directory
work_dir=$PBS_O_WORKDIR
#! starts simulation
cd $work_dir
#! nl=$work_dir/node.list
#! specifies journal file (simulation input)
#!journal=batch_tr.jou
#!fluent $version -t$nCPU -g -ssh -cnf=$PBS_NODEFILE -i $journal > log
/usr/app/ansys_inc/v120/fluent/bin/fluentbench.pl
turbo_500k -t$nCPU -pinfiniband -ssh -cnf=$PBS_NODEFILE
```

For example the above script is for ***"turbo_500k"*** case with **72** processors.

The same script was modified for running the jobs for multiple cores. It should be mentioned here that the above ***qsub*** script can directly be used for all other *MPI* runs.

There were several issues with system performances. The whole benchmarking exercise with *Reynolds* has taken several days. As per mandatory requirements, the system was kept free from any user interfaces during the calculation. Therefore all the users' login was blocked except root before running the benchmarking simulation.

V. NODE ALLOCATION BY QSUB DURING BATCH PROCESSING

It has been found that; ***qsub*** is following the same schedules for all cases. Typically it is following the given schedule for allocation of nodes/CPU in our benchmarking, which is as shown in Table 1.

Table 1: Distribution of nodes in ***qsub*** batch processes

CPU	Host Node
1	18
2	18
3	18
4	18
8	18, 19
16	18, 19, 8, 9
24	18, 19, 8, 9, 3, 2
32	18, 19, 8, 9, 3, 2, 1, 13
40	18, 19, 8, 9, 3, 2, 1, 13, 7, 6
48	18, 19, 8, 9, 3, 2, 1, 13, 7, 6, 5,
56	18, 19, 8, 9, 3, 2, 1, 13, 7, 6, 5, 4, 10, 15
64	18, 19, 8, 9, 3, 2, 1, 13, 7, 6, 5, 4, 10, 15, 14, 16
72	18, 19, 8, 9, 3, 2, 1, 13, 7, 6, 5, 4, 10, 15, 14, 16, 11, 17
80	18, 19, 8, 9, 3, 2, 1, 13, 7, 6, 5, 4, 10, 15, 14, 16, 11, 17, 12, 20

It should be noted that the above schedule is not fixed for all ***qsub*** modules. It depends upon several parameters such as system architecture and *MPI* set up. Typically this random structure of node distributions is same for all high performance servers.

VI. SYSTEM PERFROMANCE RESULTS

To evaluate *Reynolds* performance for different computational loadings, simulations have been carried out for all benchmark cases starting from single core to eighty cores. Results have been plotted in terms of benchmarking parameters. To evaluate the *Reynolds* performance over other commercially available High Performance Super Computers operating in several organizations, rating results are compared with these machines *viz*. **A**: BULL NOVASCALE_R422, **B**: HP BL460G6 (INTEL_X5570_NHM4, 2930, LINUX, IB), **C:** HP BL460G6 (INTEL_X5570_NHM4, 2930, WIN64, IB), **D**: MELLANOX_COLFAX CX1254, **E**: QLOG IC CLUSTER, **F**: SGI ALTIX_ICE (INTEL64_CLOVERTOWN_4CORE, 2600, LINUX, IB), **G**: SUN X2250

It must be noted that the architectures of those machines are having similar platforms with different numbers of nodes like *Reynolds*.

A multiple comparison has been made by defining a new parameter, *Reynolds* performance index as:

$$Reynolds\ performance\ index = \frac{Reynolds\ peak\ rating}{Other\ HPC\ machine's\ peak\ rating} \quad (3)$$

Typically assuming *Reynolds* peak rating to be 1, it can be said that;

Reynolds performance is superior if:

$Reynolds\ performance\ index > 1$

and, *Reynolds* performance is inferior if:

$Reynolds\ performance\ index < 1$

VII.SYSTEM PERFORMANCE RESULTS FOR ANSYS BENCHMARKING

A. Small Cases:

a. Case type: Reacting Flow with Eddy Dissipation Model:

Problem pertains to the situation of a reacting flow case with the eddy dissipation model. $k-\varepsilon$ turbulence and segregated implicit solver are used. The case has around 0.417 million hexahedral cells. The detail of the case is mentioned below:

Case Name	# EDDY_417K
Number of cells	0.417 million
Cell type	hexahedral
Models	$k-\varepsilon$ turbulence
Solver	segregated implicit

Simulations have been carried out for all the processors in batch mode. Performance plots have been generated with the actual run data of the *Reynolds*. Fig. 2, 3 and 4 show the variation of Rating, Speedup and Efficiency with Number of Processors. All plots are generated considering batch modules and batch processes. Data points are curve fitted using third degree polynomial. Maximum residual error of those fitting is 0.1%.

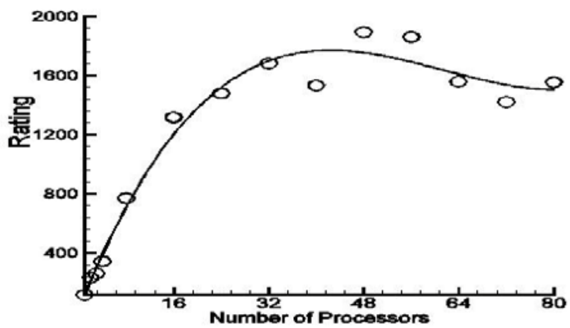


Figure 2.Variation of Rating with Number of Processors for EDDY_417K

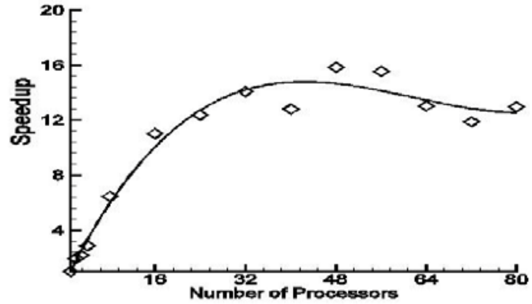


Figure 3. Variation of Speedup with Number of Processors for EDDY_417K

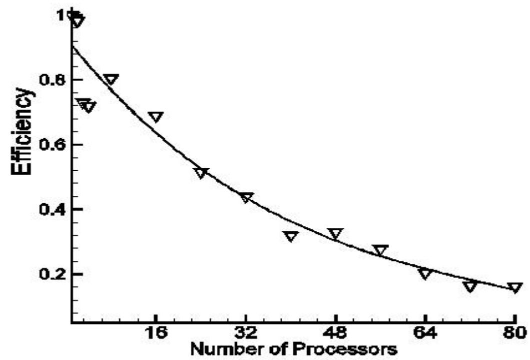


Figure 4Variation of efficiency with Number of Processors for EDDY_417K

For this case, to evaluate *Reynolds* performance over various HPC machines, *Reynolds* rating has been compared. Fig. 5 shows Comparison of *Reynolds* rating with Number of Processors for various HPC machines.

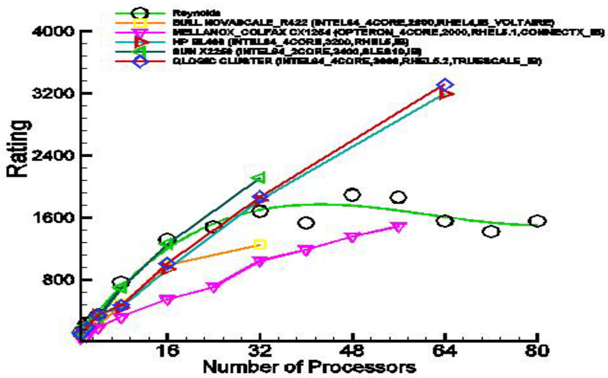


Figure 5. Comparison of Reynolds rating with Number of Processors for various HPC machines for EDDY_417K

It should be noted that the system scalability is not liner for this case and decreases within a narrow band having a saturation limit after certain numbers of processors. One may think that *Reynolds* has got unsatisfactory performance for the case concerned. Now let us clear this misunderstanding by citing expert comments on this typical characteristic. Benchmarking cases have been designed specifically for system performance test based on certain mandatory requirements. It is highly depending upon the physics of the problem concerned.

More specifically, there could be two reasons for parallel scalability for this case:

The number of cells in this case is ~414K and as we keep partitioning this case into too many cores, the communication between the cores also increases and the scalability does not remain linear. It can be understood that it is more beneficial to use large number of cores when the numbers of cells are higher.

It can be seen from the standard performance curve reported by ANSYS in the website that after 32 cores and 64 cores, it starts showing dip in the rating curve. For some of the machine configuration the dip starts little early and some shows dip little late. And same type of dip is also seen in the performance curve in *Reynolds* server.

This typical characteristic is identical for all the SGI servers, as it can be seen from performance plot, supplied by ANSYS data base [Fig. 6] benchmarks.

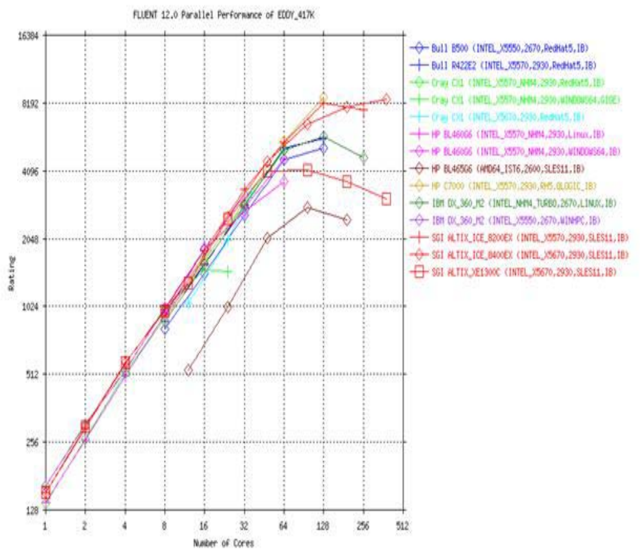


Figure 6. Rating Vs Number of cores (taken after permission from ANSYS Limited, Pune Office, India) for EDDY_417K

b. Case Type: Single-Stage Turbo Machinery Flow:

A single stage turbo machinery flow case using the Spallart-Allmaras turbulence model and the coupled implicit solver has been made for simulation set-up. The case has around 500,000 cells of mixed type. The detail of the case is mentioned below:

Case Name	# TURBO_500K
Number of cells	500,000
Cell type	mixed
Models	Spallart-Allmaras turbulence
Solver	coupled implicit

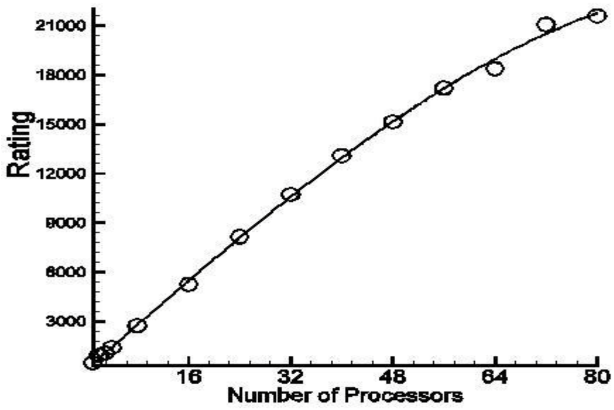


Figure 7. Variation of Rating with Number of Processors for TURBO_500K

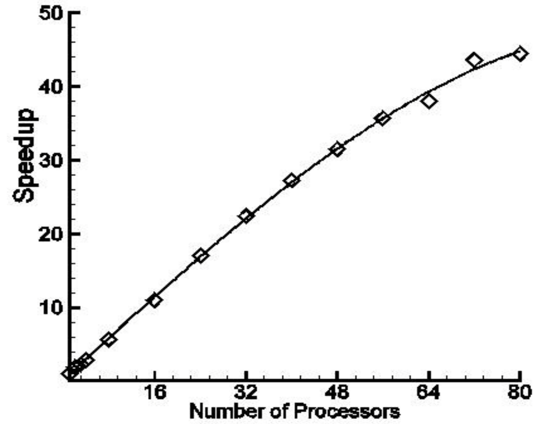


Figure 8. Variation of Speedup with Number of Processors for TURBO_500K

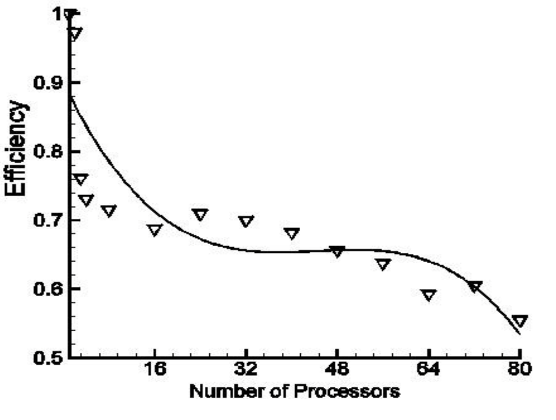


Figure 9. Variation of efficiency with Number of Processors for TURBO_500K

Parallel computations have been carried out for all the processors. Performance plots have been obtained with the actual run data of the *Reynolds*. Fig. 7, 8 and 9 show the variation of Rating, Speedup and Efficiency with Number of Processors. Data points are curve fitted using third degree polynomial. Maximum residual error of those fitting is 0.03%.

Similarly, to evaluate *Reynolds* performance over various HPC machines for this case, servers rating have been compared. Fig. 10 shows Comparison of *Reynolds* rating with Number of Processors for various HPC machines.

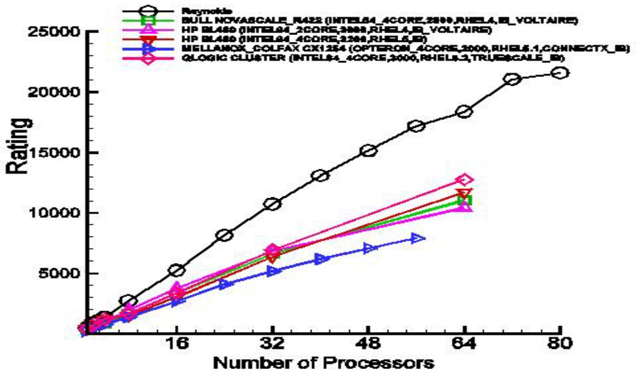


Figure 10. Comparison of Reynolds rating with Number of Processors for various HPC machines for TURBO_500K

The above figures [Fig. (2) – (10)] explains clearly the performance of *Reynolds* for different performance parameters. *Reynolds* shows steady speed up with increase in numbers of cores. This situation is quite different for the first case (Case –I) where after 32 processors its speed flattens and then reduces marginally. It is due to the fact that for such case numbers of cells are low compared to the Case II (Only 417K). Read case/data time and nodal distribution are the main limitation and thus reduce performance. But, for other case the performance of *Reynolds* is tremendous and has shown excellent speed up. Efficiency wise it shows exponential decrease that means low occupancy and increased availability of memories for doing other calculations. Comparison with other famous HPC machines eludes that *Reynolds* is a better machine for small cases.

Fig. 11 and 12 show the fact that *Reynolds* is how much time faster than other HPC machines with respect to benchmarking problems Eddy_417K and Turbo_500K. Y axis denotes *Reynolds performance index* and X axis denotes different HPC machines architectures.

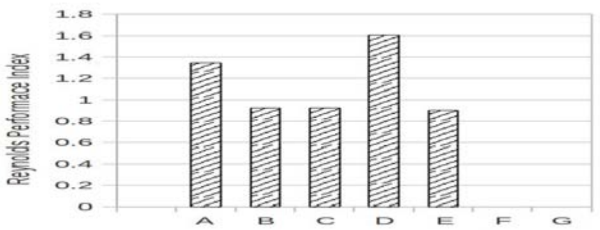


Figure 11: Performance multiple of Reynolds for different HPC Machines for EDDY_417K

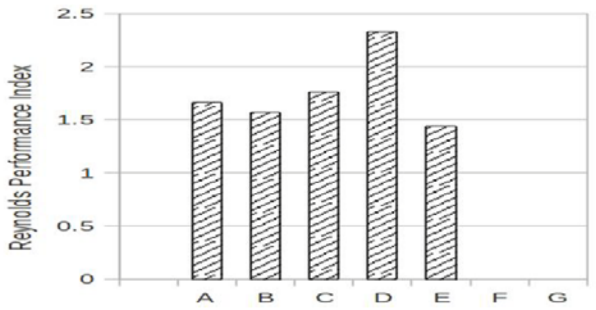


Figure 12. Performance multiple of Reynolds for different HPC Machines for TURBO_500K

It is evident from the Fig. 11 and Fig. 12 that for this small benchmarking cases *Reynolds* is performing much better in comparison to the other HPC machines. Only for Eddy_417K it is little bit slower than machines B, E and G (explanations have already been given).

B. Medium Cases:

a. Case Type: External Flow Over An Aircraft Wing:

Case setup corresponds to the situation of external flow over an aircraft wing. The case has around 1.8 million hexahedral cells and uses the realizable $k-\varepsilon$ model and the coupled implicit solver.

Case Name	# AIRCRAFT_2M
Number of cells	1.8 million
Cell type	hexahedral
Models	realizable $k-\varepsilon$ turbulence
Solver	coupled implicit

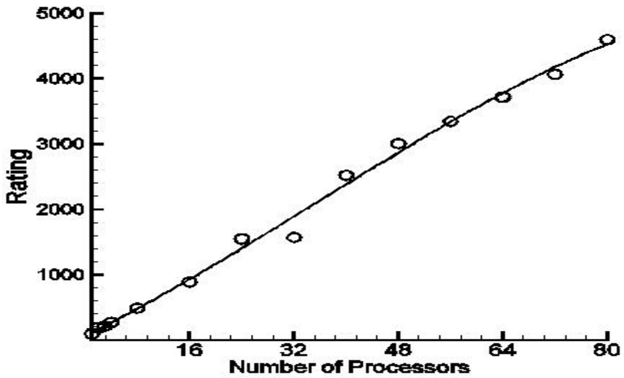


Figure 13. Variation of Rating with Number of Processors for AIRCRAFT_2M

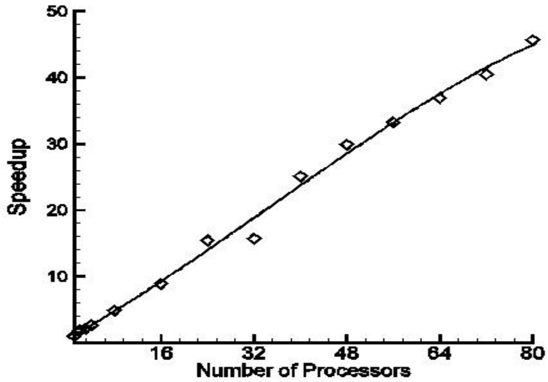


Figure 14. Variation of Speedup with Number of Processors for AIRCRAFT_2M

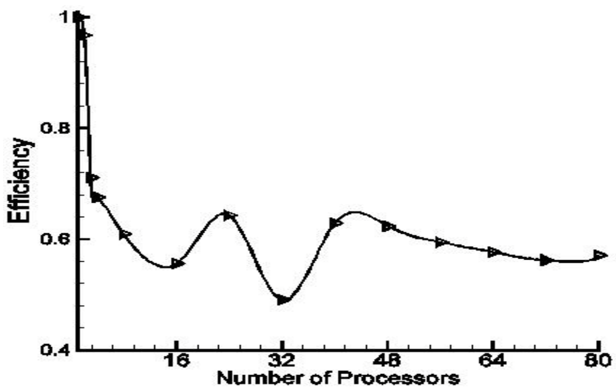


Figure 15. Variation of efficiency with Number of Processors for AIRCRAFT_2M

Batch simulations have been carried out for all the processors. System performance plots have been obtained with the actual batch run data of the *Reynolds*. Fig. 13, 14 and 15 show the variation of Rating, Speedup and Efficiency with Number of Processors. Data points are curve fitted using third degree polynomial. Maximum residual error of those fitting is 0.02%.

Here also, to evaluate *Reynolds* performance over various HPC machines for this medium case, machine rating has been compared. Fig. 16 shows Comparison of *Reynolds* rating with Number of Processors for various HPC machines.

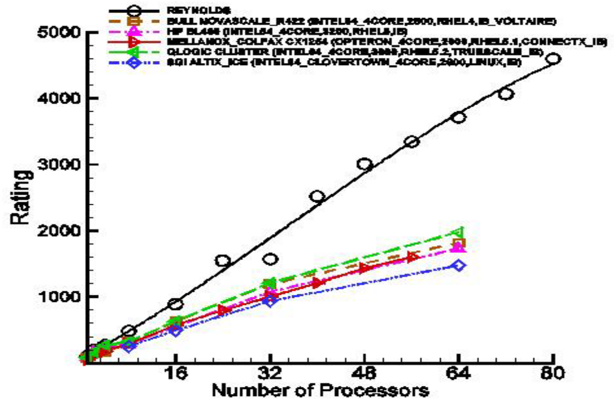


Figure 16. Comparison of Reynolds rating with Number of Processors for various HPC machines for AIRCRAFT_2M

b. Case Type: External Flow Over A Passenger Sedan:

Problem under consideration is external flow over a passenger sedan. The case has around 3.6 million cells of mixed type and uses a $k-\varepsilon$ model with the pressure-based coupled solver.

Case Name	# SEDAN_4M
Number of cells	3.6 million
Cell type	mixed
Models	$k-\varepsilon$ turbulence
Solver	pressure based coupled implicit

After completion of batch simulations, system performance plots have been generated with the actual run data of the *Reynolds*. Fig. 17, 18 and 19 show the variation of Rating, Speedup and Efficiency with Number of Processors. A third degree polynomial has been used for curve fitting of data points. Maximum residual error of those fitting is 0.1%.

A comparison has been made in terms of machine rating, to evaluate *Reynolds* performance over various HPC machines. Fig. 20 shows

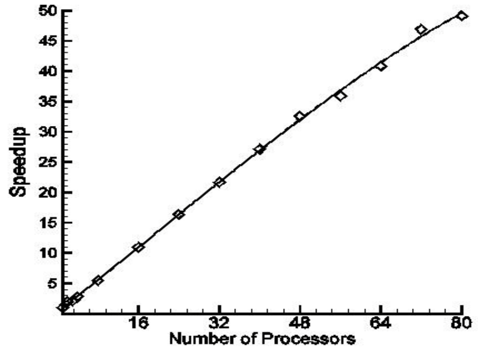


Figure 17. Variation of Rating with Number of Processors for SEDAN_4M

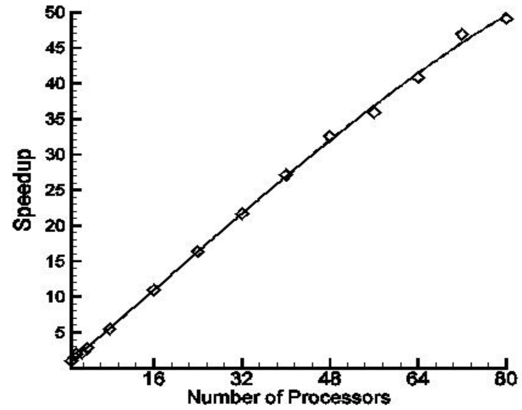


Figure 18. Variation of Speedup with Number of Processors for SEDAN_4M

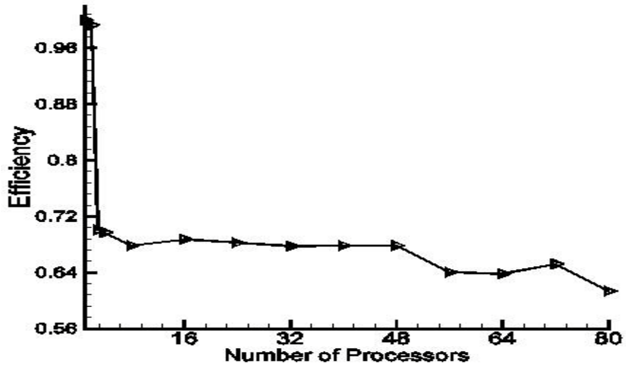


Figure 19. Variation of efficiency with Number of Processors for SEDAN_4M

Comparison of *Reynolds* rating with Number of Processors for various HPC machines.If we have a glance over the figures [Fig. (13) – (20)] *Reynolds* performance parameters show a liner scale-up in Rating and speed up, whereas there is an exponential decay of efficiency. This clearly establishes the splendid performance of *Reynolds* in medium cases. It can also be noticed that there is a small fluctuations in the decaying efficiency curve. This phenomenon may be due to the minute perturbations in intra node communications. Comparison of figures with different HPC machines shows superlative performance of *Reynolds*.

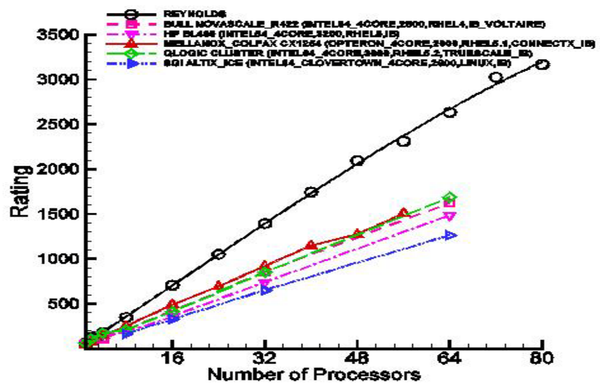


Figure 20. Comparison of Reynolds rating with Number of Processors for various HPC machines for SEDAN_4M

Like small cases, Fig. 21 and Fig. 22 show the fact that *Reynolds* is how much time faster than other HPC machines for medium cases. Y axis denotes *Reynolds performance index* and X axis denotes different HPC machines architectures.

Here also it is evident from the Fig. 21 and Fig. 22 that for this medium benchmarking cases *Reynolds* is performing stupendously in comparison to the other HPC machines.

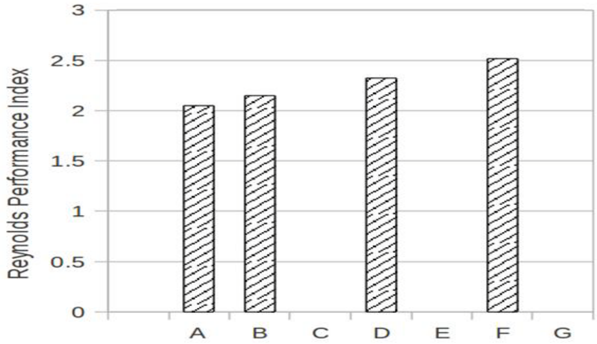


Figure 21. Performance multiple of Reynolds for different HPC Machines for AIRCRAFT_2M

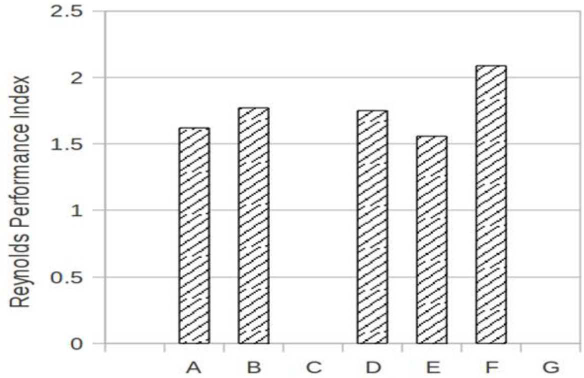


Figure 22. Performance multiple of Reynolds for different HPC Machines for SEDAN_4M

C. Large Cases:

For large case the speed up and efficiency parameters have not been calculated as those requires wall clock time for single core. Now with the case of having 14 million cells, single core simulation is not at all acceptable as it reduces system memory and sometimes system crashes during simulation. Another problem is that SUSE shells do not support large memories with single processors. One may refer to SUSE Enterprise manual [16] for detail discussions. Considering above facts we are limited with only rating calculations. It is known that Rating is proportional to Speedup, whereas it is inversely proportional to the efficiency.

a. Case Type: External Flow Over A Truck Boday:

Here the situation corresponds to the case of external flow case over a truck body. The case has around 14 million cells of mixed type and uses the DES model with the segregated implicit solver.

Case Name	# TRUCK_14M
Number of cells	14 million
Cell type	mixed
Models	DES turbulence
Solver	segregated implicit

After finishing of batch processes *Reynolds* rating has been calculated for multiple parallel processors. Fig. 23 show the variation of Rating with Number of Processors. A cubic polynomial has been used for curve fitting of data points. Maximum residual error of those fitting is 0.04%.

Fig. 24 shows comparison of *Reynolds* rating with Number of Processors for various HPC machines. It clearly indicates *Reynolds* superiority for the same platforms.

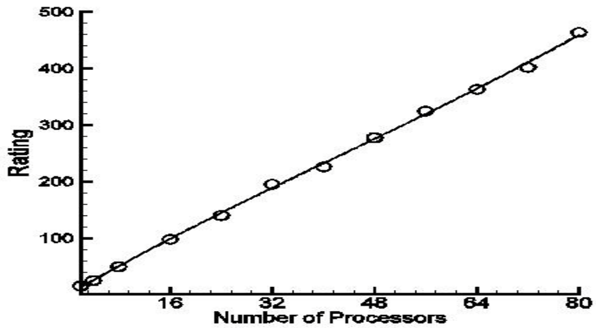


Figure 23. Variation of Rating with Number of Processors for TRUCK_14M

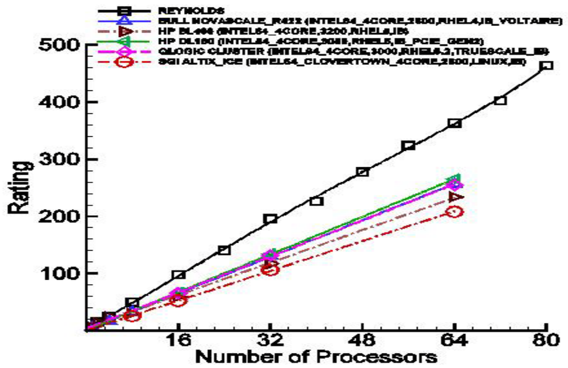


Figure 24. Comparison of Reynolds rating with Number of Processors for various HPC machines for TRUCK_14M

b. Case Type: External Flow Over A Truck Body with A Polyhedral Mesh:

Problem under consideration is the situation of external flow over a truck body using a polyhedral mesh. The case has around 14 million polyhedral cells and uses the DES model with the segregated implicit solver.

Case Name	# TRUCK_POLY_14M
Number of cells	14 million
Cell type	polyhedral
Models	DES turbulence
Solver	segregated implicit

Large scale batch processes involves increase in computation time. Simulations have been carried out for this case with all the parallel processors. Fig. 25 shows the variation of Rating with Number of Processors. A third order polynomial has been used for curve fitting of data points. Maximum residual error of those fitting is 0.1%.

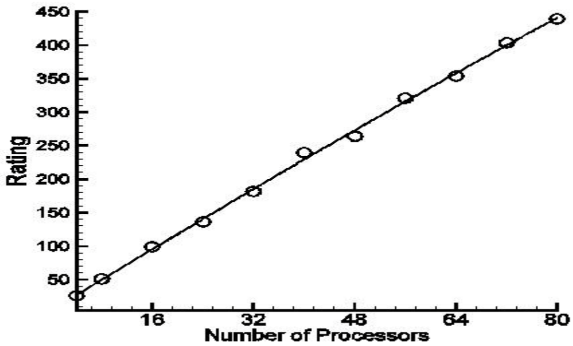


Figure 25. Variation of Rating with Number of Processors for TRUCK_POLY_14M

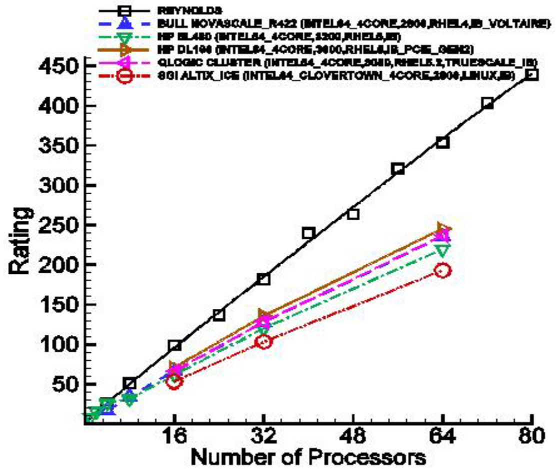


Figure 26. Comparison of Reynolds rating with Number of Processors for various HPC machines for TRUCK_POLY_14M

Fig. 26 shows comparison of *Reynolds* rating with Number of Processors for various HPC machines. It clearly indicates *Reynolds* superiority for the same platforms.

Fundamentally for system performance evaluation it has been proved and tested that speedup is directly proportional to the rating and inversely proportional efficiency. *Reynolds* performance curves for the large cases show a super-linear speed up in computational efficiency and system functional rating, which is increasing with increase in number of processors. The super-linear speedup is attributed to cache related effects. Characteristics scale up is linear with about 45^0slope. In this context it can be concluded that speedup must be having same scale up as it is directly proportional to the rating and exponential decay in efficiency for its inversely proportional characteristics. Fig. 23 to Fig. 26 convinces us to conclude that *Reynolds* is performing excellent in large case computations.

Fig. 27 and Fig. 28 have been plotted to show the fact that *Reynolds* is how much time faster than other HPC machines for large cases. Y axis denotes *Reynolds performance index* and X axis denotes different HPC machines architectures.

It is obvious from the Fig. 27 and Fig. 28 that for this large benchmarking cases *Reynolds* is performance is superior in comparison to the other HPC machines.

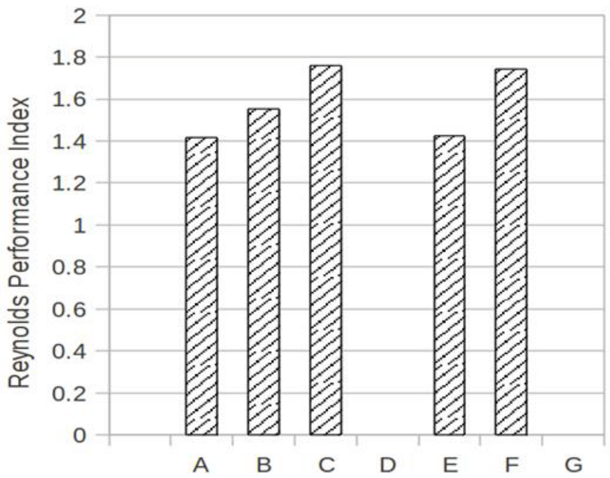


Figure 27. Performance multiple of Reynolds for different HPC Machines for TRUCK_14M

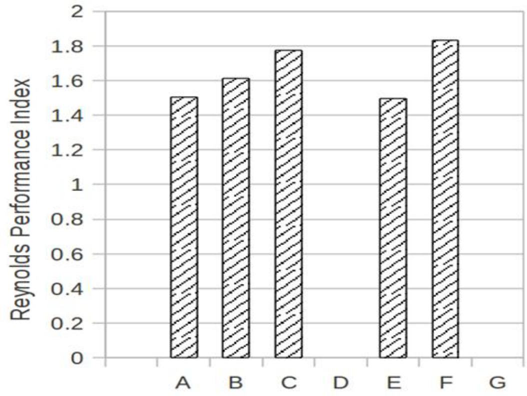


Figure 28. Performance multiple of Reynolds for different HPC Machines for TRUCK_POLY_14M

VIII. REYNOLDS PERFORMANCE FOR A REAL INDUSTRIAL CASE RUNNING AT TATA STEEL PLANT

To evaluate the performance of Reynolds for a real industrial case running at Tata Steel plant, we choose tundish with four strands [17]. In any integrated steel plant, the tundish in a continuous casting operation is an important link between the ladle, a batch vessel and the casting mould with a continuous operation. One of the important aspects of tundish vessel is to optimize the liquid metal flow rate and increasing the productivity. Such cases demand tundish having variety of geometrical configurations.

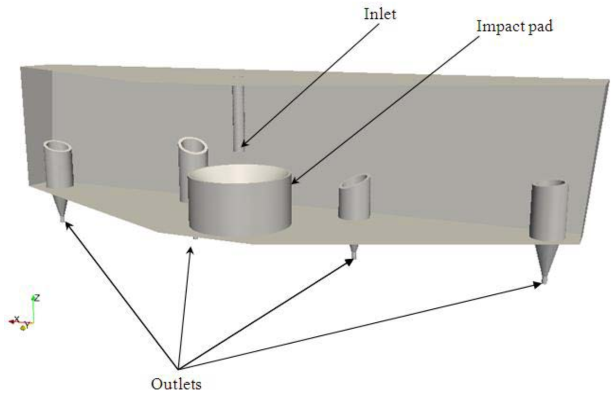


Figure 29. Typical geometry of four strand tundish

Fig. 29 show the typical tundish geometry operating at plant. As per as the physical dimensions are concerned, the longitudinal and transverse length were 3.5m and 1m respectively. The liquid steel level of the tundish was 0.85m. Actual plant conditions were incorporated in simulating the basic flow and turbulence equations. For doing such simulations we used OpenFOAM CFD package.

Preamble information's on OpenFOAM is as follows. OpenFOAM is a free, open source CFD software package produced by a commercial company, OpenCFD Ltd. It has a large user base across most areas of engineering and science, from both commercial and academic organizations. OpenFOAM has an extensive range of features to solve anything from complex fluid flows involving chemical reactions, turbulence and heat transfer, to solid dynamics and electromagnetics.

We carried out a sample performance test with OpenFOAM. For doing such calculations we generated the grids with Gambit. Mesh was incorporated in

OpenFOAMcase set. Solver used for doing such calculation was simpleFOAM, which is specially developed solver with OpenFOAM. Steady state computations were carried out with $k-\varepsilon$ **t**urbulence modeling. The basic fluid flow equations we solved were mass, momentum, and turbulence.

IX. MATHEMATICAL MODELLING

The mathematical model is based on the assumptions of continuum hypotheses that demand the mean free path within the permissible limits. The turbulence kinetic energy and intensity are assumed to be equilibrium with the fluid flow and liquid state enthalpy. Fluid is assumed to be incompressible and follows Boussinesq's approximation in density variation. Considering those approximations, the governing equation consists in the simultaneous solution of the continuity, momentum transfer and energy transfer equations under turbulent-unsteady conditions together with the equations of turbulent kinetic energy *k* and its dissipation rate ε :

Mass

$$\frac{\partial\rho}{\partial t}+\nabla\bullet\rho u=0 \quad (4)$$

Momentum

$$\frac{\partial(\rho u)}{\partial t}+u\bullet\nabla(\rho u)=-\nabla P+\nabla\bullet\left[(\mu_l+\mu_t)\nabla u\right]+F_B \quad (5)$$

In these equations F_B is the buoyancy force term, u is the time-averaged velocity vector and T is the steel temperature in the three-dimensional (3D) domain. Additionally, μ_l and μ_t are molecular and turbulent viscosity of steel, respectively and κ is the steel conductivity. The turbulent viscosity is calculated by knowing the turbulent kinetic energy and its dissipation rate, which are given by their conservation equations:

Kinetic energy

$$\frac{\partial(\rho k)}{\partial t}+\nabla\bullet(\rho uk)=\nabla\bullet\left[\Gamma_k\nabla k\right]+G-\rho\varepsilon \quad (6)$$

Dissipation rate

$$\frac{\partial(\rho\varepsilon)}{\partial t}+\nabla\bullet(\rho u\varepsilon)=\nabla\bullet\left[\Gamma_\varepsilon\nabla\varepsilon\right]+\frac{\varepsilon}{k}(C_1G-C_2\rho\varepsilon) \quad (7)$$

and

$$\mu_t=C_\mu\rho\frac{k^2}{\varepsilon} \quad (8)$$

Where Γ_k and Γ_ε are the diffusion coefficients for the turbulent kinetic energy and its dissipation rate, respectively, and are given by

$$\Gamma_k=\frac{\mu_{eff}}{\sigma_k}\quad \Gamma_\varepsilon=\frac{\mu_{eff}}{\sigma_\varepsilon} \quad (9)$$

Where μ_{eff} is the effective viscosity and is given by $\mu_{eff}=\mu_l+\mu_t$

The tensor expression for the generation term *G* is given as

$$G=\mu_t\frac{\partial u_j}{\partial x_i}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right) \quad (10)$$

Values for $C_\mu, C_1, C_2, \sigma_k, \sigma_h$, *and* σ_ε are 0.09, 1.44, 1.92, 1.0, 0.9, and 1.3, respectively.

X. INITIAL AND BOUNDARY CONDITIONS

At inlet the mean velocity is assumed to be uniform though its cross section and the other two perpendicular velocities are assumed to be zero. Corresponding to the plant condition, an uniform liquid steel velocity of 1.5m/sec was taken. The turbulent kinetic energy and its dissipation rate are assumed to be uniform and also calculated in terms of turbulence intensity by fixing as 6%. Boundary conditions for momentum transfer at all solid surfaces including walls and bottom of the tundish, surfaces of impact pad are those of non-slipping, zero normal gradients at symmetry planes and frictionless conditions at the free surface of liquid steel. Similar boundary conditions are established for turbulent kinetic energy and its dissipation rate. Near any solid surface, including walls and bottom of the tundish, impact pad and side walls of shroud and tundish outlets, a standard wall function for velocity distribution was applied [18]. At outlets pressure outlet boundary condition was adopted.

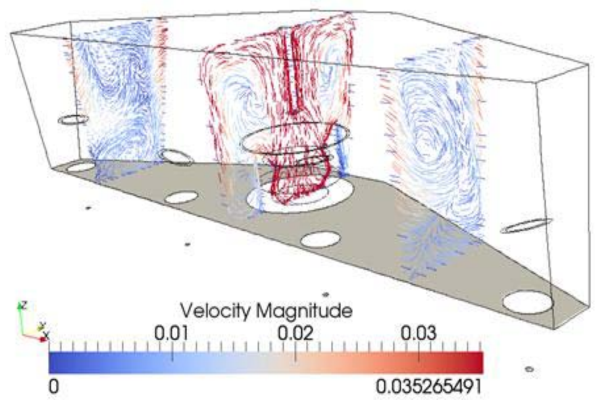
The case detail is as follows:

Case Name	# Four_Strand_Tundish
Number of cells	0.8 million
Cell type	unstructured
Models	$k-\varepsilon$ turbulence
Solver	simpleFoam
Iterations	2000
Write frequency	100 iterations

Parallelization was set with open *MPI* schedule. Processors selections were done with ***qsub***modules only.

XI. RESULTS AND DISCUSSIONS

The displays of velocity field presented in the following discussion correspond to different cross-sectional planes under steady state flow simulation. Velocity field under the steady state condition with an input of steel at a constant temperature of 1823K is shown in Fig. 30a – 30b, at different cross-sectional planes. As expected, it can be seen the formation of vortex structures at those planes. To capture the exact flow field away from the ladle stream, few mass less particles were injected from the inlets and their trajectories were tracked with the time instants. Fig. 31 shows the trajectory of mass less particles in terms of the time taken by particles to reach at the outlets.



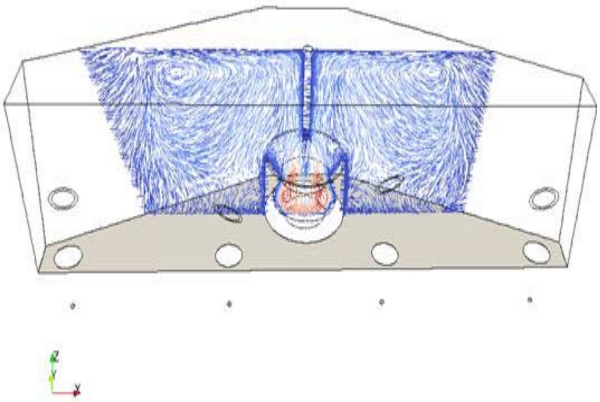


Figure 30 (a).Velocity vectors at transverse planes and Figure 30 b. velocity vectors at longitudinal plane

XII.PERFORMANCE PLOT

Fig. 32 and 33 show the variation of Speedup and Clock Time with Number of Processors. It was observed from the Fig. 32 that there is a reduction of speedup after 40 numbers of processors where asFig. 33 indicates a sharp reduction in clock time in computations with increase in nodes. This reduction is quite normal as one can understand that this selection of case is arbitrary and not at all comparable with benchmarking cases set up. Also the selection of mesh refining and mesh optimization has not been done. This increases intra node communication time, thus reduction in speedup. But, it has also been noted that *Reynolds* show a very good performance irrespective of such unfavorable situations in benchmarking with OpenFOAM. It can be said that the favorable region (or the Red zone in terms of Benchmarking) where the parallel processors can be used in this case is up to 32 CPUs.

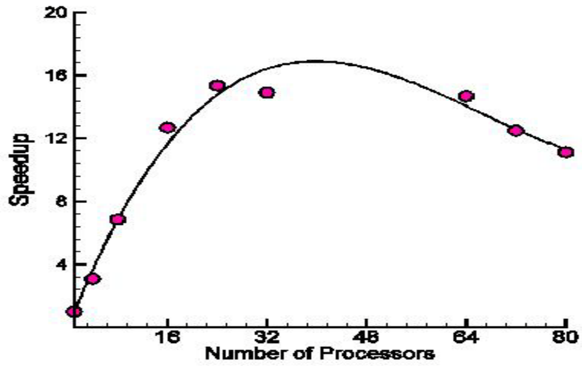


Figure 32. Variation of Speedup with Number of Processors for Four_Strand_Tundish

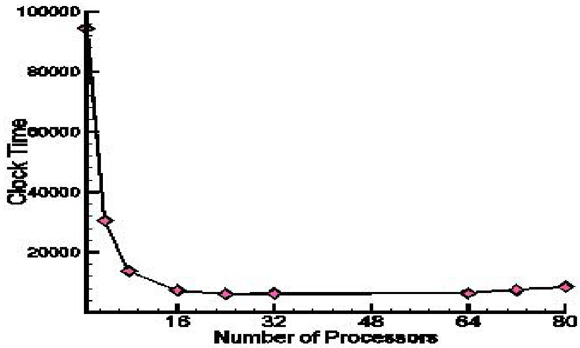


Figure 33. Variation of Clock Time with Number of Processors for Four_Strand_Tundish

XIII. RECOMMENDATIONS

Table 2 shows the recommendations for using number of processors for doing computations in *Reynolds.* This describes the best possible performance for fluent cases rather cases with different number of cells. This data are obtained after analyzing the performance plots.

Table 2: Final recommendations for number of processors in computations

Sl.	Case Type	Cell Type	Cells (millions)	Processors Min	Processors Max
1	Small – 1	hex	up to 0.417	1	32
2	Small – 2	mixed	0.417 – 0.5	1	70
3	Medium	hex/mixed	0.5 – 3.6	4	72
4	Large	mixed/poly	3.6 – 14	8	80

XIV. CONCLUSION

Utilization of advanced benchmarking exercise showed a surmountable help to improve the utilization of advanced computer technologies. Sequence of doing such activity provides a readymade tool to objectively assess performance for industrial applications. This is very important for the users to select the best computational platforms for the industrial needs. The results of such benchmarking exercises play a state of art role in guiding vendors as what is needed for improving performance industrial high performance computers on different platforms and thereby reducing the time required for the industrial customers to perform benchmarking exercises.

Reynolds benchmarking exercises have been performed exhaustively and nuances of whole process setup have been demonstrated. It is found that a random structure of node distributions while doing computations in parallel mode. System performance results have been analyzed in details. Computation shows that *Reynolds* performance rating exhibits a stupendous performance for all ANSYS Fluent benchmarking cases. This simply tells us the cluster system has a good extensibility and along with the increase in the number of processors, the float point calculation increases to maximum times in a second. But it has been found that for a very small case with having 0.417 million cells, its scale up is only up to certain number of nodes. It has been clarified that this reduction of scale up begins to decline mainly due to the need of exchange data between swap disk partition and node allocation time. Performance curves show liner scale-up in almost all the cases. Due to the minute perturbations in intra node communications, it has been found that there is a fluctuation in decaying efficiency curve for medium cases. An overall super-linear speedup is realized for large-scale problems. Benchmarking results with OpenFOAM shows that with increase in nodes a reduction of speedup is observed after 40 numbers of processors where as there is a sharp reduction in clock time till 80 processors. But overall performance result of *Reynolds* machine is found to excel in high speed computations. Recommendations for node usage have been given in terms of computational complexities and mesh sizes.

XV. ACKNOWLEDGEMENT

The authors would like to thank Tata Steel management for giving permission to publish this study. All supports from High Performance Lab colleagues are highly solicited.

XVI. REFERENCES

[1] R.Eigenmann, S. Hassanzadeh, “Benchmarking with real industrial applications: the SPEC high performance group”, IEEE Comput. Sci. Engrg.,vol. 3, pp. 18 – 22, March 1996.

[2] Julian Borrill, Leonid Oliker, John Shalf, Hongzhang Shan, Andrew Uselton, “HPC global file system performance analysis using a scientific-application derived benchmark”, Parallel Computing, vol 35, pp. 358–373, Sep 2009 .

[3] Myron Ginsberg, “High-speed applications in the automotive industry Influences, challenges, and strategies for automotive HPC benchmarking and performance improvement”, Parallel Computing, vol 25, pp. 1459 – 1476, Sep 2009.

[4] Jack J. Dongarra, “Performance of Various Computers Using Standard Linear Equations Software”, Dec 2003, doi: <http://www.netlib.org/benchmark/performance.ps>

[5] Oliver Schreiber, Scott Shaw, Brian Thatch, Bill Tang, “LS-DYNA on Advanced SGI Architectures, 11th LS-DYNA Users International Conference”, USA, March 2010.

[6] SGI Altix XE1300 ClusterQuick Reference Guide, version 002, April 2007.

[7] MPI-Tile-I/O. http://www-unix.mcs.anl.gov/pio-benchmark/

[8] SGI Home page, http://www.sgi.com/

[9] Matthias Brehm, “Technical Report on Overview of Research Projects on HLRB I”, December 2006, Germany.

[10] M. Ginsberg, “Challenges to the use of supercomputers and scientific visualization for automotive applications”, Computers and Graphics, vol 17, pp. 507 – 515, Jan 1993.

[11] M. Eldredge, T.J.R. Hughes, R.M. Ferencz, S.M. Rifai, “High-performance parallel computing in industry”, Parallel Computing,vol 23, pp. 1217 – 1233, July 1997 .

[12] Suresh Behara, Sanjay Mittal, “Parallel finite element computation of incompressible flows”, vol 35, pp. 195-212, Dec 2990.

[13] J. Dongarra, W. Gentzsch , “Benchmarking of high performance computers”, Parallel Computing, vol 17, pp. 1067-1069, March 1991.

[14] R.S. MonteroE,. Huedo and I.M. Llorente, “Benchmarking of high throughput computing applications on Grids”, vol 32, pp. 267-279, Dec 2006.

[15] ANSYS FLUENT Benchmark Suite, <http://www.fluent.com/software/fluent/fl6bench/index.htm>.

[16] SUSE Linux Enterprise, <http://www.novell.com/linux/>

[17] OpenFOAM, <http://www.openfoam.com/>

[18] Sandip Sarkar, R. Sambasivam, S.K.Ajmani, and M. B. Denys, “Numerical analysis of unsteady hydrodynamics and thermal transport in a five strand asymmetric tundish, Ironmaking and Steelmaking”, doi: http://dx.doi.org/10.1179/1743281212Y.0000000001, Jan 2012.

