Francis 99 CFD through RapidCFD accelerated GPU code

D Molinero, S Galván, F. Domínguez, L Ibarra and G Solorio.

Faculty of Mechanical Engineering, Faculty of Physics and Mathematics, Universidad Michoacana de San Nicolás de Hidalgo, Morelia, México.

Abstract. Francis turbines research and development (R&D) requires performance assessment through hydraulic laboratory model testing which can be assisted by auxiliary tools like computational fluid dynamics (CFD), widely used recently. CFD has a history of seeking and requiring ever higher computational performance (HPC) because of the parallelism where Graphics Processor Units (GPUs) have emerged as a major paradigm for solving complex computational problems. However, their implementation to CFD solvers is still a challenge and the tremendous computational power of the GPUs has been wasted. This work presents how the open source RapidCFD code, based on OpenFOAM and ported to Nvidia CUDA, enabled GPUs to be able of running almost entire simulations in thousands of parallel stream cores packed in small form factor hardware in order to solve the incompressible Reynolds-Average-Navier-Stokes (RANS) equations. The simulations were based on a full 3D Francis turbine case which consisted of a grid domain of 23 million cells, including spiral case with stay vanes, distributor, runner and draft tube. CFD results of shaft torque, static pressure and velocity components in steady state deploying a multiple reference frame (MRF) motion approach were compared with available experimental data for main operation conditions at different distributor opening angles: best efficiency operation point (BEP), part load operation point (PL) and full load operation point (HL). The obtained data showed that by transferring directly all the computations to the GPUs, it is possible to make CFD simulations faster compared with central processing units (CPUs). Thus, it is expected to obtain an affordable low computational cost in optimization processes or full range performance evaluations.

1. Introduction

Over the last decades the increasing computing power of new hardware has allowed a wide implementation of CFD in industrial and non-industrial applications from aerodynamics to biomedical engineering [1]. In this scenario some groups of turbomachinery researchers have encouraged and enhance spaces to explore more in deep the use of CFD in related phenomena as well as validate and evaluate results against experimental data [2-4]. More recently the Francis 99 test case [5] has been opened in order to improve CFD capabilities and apply modern tools and techniques to turbomachinery applications.

Nowadays CFD is a reliable tool that can be used in performance prediction of turbomachinery as proved by [6], where a series of tests were conducted for a Francis turbine under different specific speeds. CFD calculations were compared with experimental data, in general it was observed a good agreement for the quantities analyzed. Deployment of CFD to turbomachinery applications is in fact quite important in designing and optimizing new equipment for retrofit power plant projects. In their
work [7] inverse design and CFD simulations combination gives as a result a new design which surpassed its predecessor by 9.93% in peak efficiency along with better cavitation characteristics.

As the compute power has raised the complexity and accuracy of phenomena solved and analyzed in CFD has followed the same path, beginning with potential flow to follow with the solution of Reynolds Average Navier-Stokes (RANS) equations in steady and transient state in order to include viscous effects [8]. Thus, we have come up to analyze more complex geometries within an even finer threshold of accuracy, here is were GPUs have emerged as a new paradigm for solving complex computational problems because of their design features result in computational power and memory bandwidth that exceed the features of the fastest multi-core CPUs by almost an order of magnitude [9].

Nevertheless GPUs are suitable to perform large computing tasks and have huge performance advantages over CPUs for CFD applications, there are yet some drawbacks to their full implementation in general CFD codes and their extended use is still a challenge [10-12].

In our opinion, the Francis 99 test case is an excellent opportunity to test GPUs implementation to turbomachinery since the numerical modeling of hydraulic turbines is quite a challenge. First because the specific modeling of a problem to investigate an operational condition does not always work for others; second, the simulation of a complete turbine requires substantial computational resources.

Thus, in order to take advantage of the tremendous computing power of GPUs in CFD, the RapidCFD library based on OpenFOAM and coded in CUDA was used in the numerical solution of the Francis 99 benchmark. A series of tests were carried out to determine the fixed size grid problem speedup according to Amdahl's law with the available computational resources. Finally, the validation of the numerical results demonstrates the accuracy of the solution that can be achieved through CFD and GPUs coupling.

2. Francis 99 benchmark
Verification and validation of CFD simulations is almost a titanic and impossible mission due to lack of public available information regarding well defined geometries and reliable data from experiments. In order to overcome this breach in the development of CFD methods and technics, the Francis 99 workshops series were made open to the research community.

2.1. Francis 99
The workshops were held at Norwegian University of Science and Technology (NTNU) which included valuable works for the state of the art in the CFD and turbomachinery field. Different software packages including commercial and open source, and numerical approaches were used by the participants in the workshops.

Since the main development to accelerate CFD computations by means of GPUs usage has been made in the open source software field, we would like to remark those works using OpenFOAM and/or its forks [13-16], whose insights have guided to set up the CFD process in this work.

The Francis 99 test case consists of a Francis type turbine model (1:5.1 scale) of the turbines operating at Tokke power plant in Norway. Figure 1 shows the complete model of the Francis turbine and the runner blades.

![Figure 1. Francis 99 turbine’s isometric (left), close up (center) and runner blades (right) views.](image-url)
2.2. Experimental data

For all CFD analysis we used the test-case provided by NTNU – Norwegian University of Science and Technology under the Francis-99 workshop series [5]. The operating conditions analyzed were part load (PL), best efficiency point (BEP) and high load (HL). Table 1 shows the available data from 1st and 2nd workshops that were used as boundary conditions and to validation in this work.

| Parameter                                | PL1      | PL2 | BEP | BEP | HL     | HL |
|------------------------------------------|----------|-----|-----|-----|--------|-----|
| Turbine inlet pressure absolute (kPa)    | 219.93   | 218.08 | 216.54 | 215.57 | 210.01 | 212.38 |
| Differential pressure across the turbine (kPa) | 120.39 | 104.91 | 114.98 | 104.44 | 114.03 | 102.79 |
| Water density (kg/m$^3$)                 | 999.23   | 999.80 | 999.19 | 999.80 | 999.20 | 999.80 |
| Kinematic Viscosity (m$^2$/s)            | 9.57E-7  | 9.57E-7 | 9.57E-7 | 9.57E-7 | 9.57E-7 | 9.57E-7 |
| Gravity (m/s$^2$)                        | 9.82     | 9.82  | 9.82  | 9.82  | 9.82  | 9.82  |
| Net head (m)                             | 12.29    | 11.87 | 11.91 | 11.94 | 11.24 | 11.88 |
| Discharge (m$^3$/s)                      | 0.07100  | 0.13962 | 0.20300 | 0.19959 | 0.22100 | 0.24246 |
| Runner torque (N-m)                      | 144.06   | 420.79 | 628.41 | 620.65 | 605.62 | 744.39 |
| Runner speed (rpm)                       | 406.20   | 332.84 | 335.40 | 332.59 | 369.60 | 332.59 |
| Hydraulic efficiency (%)                 | 71.69    | 90.13  | 92.61  | 92.39  | 90.66  | 91.71  |
| Guide vane angle (degree)                | 3.91     | 6.72   | 9.84   | 9.84   | 12.44  | 12.44  |

*a Second workshop

3. GPU implementation

3.1. Software

In this work is intended to show how the astonishing computing power of GPUs and their thousands of parallel stream cores are able to accelerate the solution of turbomachinery CFD simulations through the implementation of the open source OpenFOAM in the CUDA API (Compute Unified Device Architecture Application Programming Interface). Beyond only to code the linear solvers, as several libraries have done before (e.g. Cufflink, ofgpu, speed IT, and parallution) without modifying drastically the original code and applied as a simple plug-in that allows running the linear solvers on the GPUs [17, 18], the entire solution algorithms have been implemented to accelerate OpenFOAM through GPUs. The tool available to such task has been given the name of RapidCFD [19, 20] and it is based on OpenFOAM 2.3.1, limited only by the lack of preprocessing and post processing utilities which can be handled by standard versions of OpenFOAM.

Deployment of RapidCFD in turbomachinery CFD applications has been previously reported by [21] showing the software’s capabilities and giving hints about superior performance characteristics of GAMG (Geometric Algebraic Multi Grid) over PCG/PBiCG (Pre Conjugate Gradient/Pre Bi Conjugate Gradient) linear solvers regarding solution time for a centrifugal pump impeller. Speed up evaluation through Amdahl’s law was carried out by [22] getting an acceleration of solution around 9x when comparing a serial CPU with GPU parallel execution in the solution of T99 draft tube case.

3.2. Hardware

The Supermicro server equipped with the appropriate GPU cards that were used to accelerate the CFD solution through massive parallelization are detailed in Table 2.

| Parameter                        | CPU                   | GPU                   |
|----------------------------------|-----------------------|-----------------------|
| 2 x Intel Xeon E5-2640 v2, 2.0 GHz | 4 x Nvidia Tesla K40 |
| 8 cores per processor / 16 threads | 2880 CUDA cores, 745 MHz |
| 64 GB Memory DDR3, 1600 MHz     | 12 GB Memory GDDR5, 3GHz |
4. CFD parameters
For the sake of the comparison analysis of speed up and accuracy of the simulations the same set up and boundary conditions were used in OpenFOAM and RapidCFD, or at least equivalent options.

4.1. Numerical set up
The Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) algorithm was used to solve the continuity equation (1) and the momentum equation (2) in the computation process.

\[
\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{1}
\]

\[
\frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \mathbf{u} \phi) - \nabla \cdot (\Gamma \nabla \phi) = -\nabla p + S_u \tag{2}
\]

In order to reach a target residual of 10E-03, all cases were solved with double floating precision using simpleFoam application for incompressible and steady state flow, i.e., density is constant \(\frac{\partial \rho}{\partial t} = 0\) and the temporal term \(\frac{\partial \phi}{\partial t} = 0\). Since the general transport equation used in the Finite Volume Method (FVM) is second order, it is recommended to use at least second order discretization schemes. However, in order to assure stability and convergence for the test cases, a first order interpolation scheme (upwind differencing) was used for the convective (divergence) terms \(\nabla \cdot \phi\), what is equivalent to assuming that the cell values are isotropic with a value that represents the average value. Even when is well known that upwind discretization scheme tends to be diffusive it has been demonstrated that it can be used as a good initial approximation in turbomachinery applications [23]. The remaining gradient \(\nabla \phi\) and diffusive (laplacian) \(\nabla^2 \phi\) terms used a second order linear interpolation scheme (central differencing).

The linear solver of the pressure discretized equation was GAMG since it can solve symmetric or asymmetric matrices and presents an efficient transport of information across the solution domain. The velocity equations were solved through the GAMG linear solver. Using a Gauss-Seidel smother the performance of solver is improved.

In some cases, the goal of fluid flow simulations on supercomputers that use GPUs is typically to study turbulence, not complex geometries [24]. However this study involves both of them, a complex geometry and the viscous effect of the fluid flow. Thus, the \(k – \omega\) SST turbulence model was used in all the simulations since it is a common used model in turbomachinery applications. The GAMG linear solver was used to solve the turbulent scalar quantities \(k\) and \(\omega\).

The above described set up was used to solve a domain of 22,984,188 cells for each operating conditions. The mesh was provided by NTNU [5] in separate domains for each component which were merged in one single domain with four regions and two cell zones using OpenFOAM preprocessing tools like mergeMesh, stitchMesh, and topoSet among others. In order to transfer information between regions in the merged domain interfaces the Arbitrary Mesh Interface (AMI) boundary condition was used. Boundary conditions used for all analysis were fixed value for velocity at inlet based on the flow rate and fixed value for pressure at outlet (zero gauge pressure). All faces in the domain were set as walls with no slip condition, except those belonging to runner which were set as moving walls in order to use the MRF approach.

5. Results of the CFD simulations
5.1. Best Efficient Point
The acceleration quantification for a fixed size problem \(n\) was calculated through the speed up, equation (3). This parameter is one of the most important actions in parallel computing and it actually measures how much faster a parallel algorithm runs with respect to the best sequential one.

\[
S_p = \frac{T_s(n, 1)}{T_p(n, N)} \tag{3}
\]

Where \(T_s\) represents the time solution in serial by one processor, in this case by one thread, and \(T_p\) the time required to solve the same problem using \(N\) threads.
BEP condition from 1\textsuperscript{st} workshop was selected as comparison benchmark for the acceleration. The computational domain was solved using 1, 4 and 32 threads in OpenFOAM v1912 while in RapidCFD was solved using 4 GPUs equivalent to 11,520 stream cores. In Figure 2 the speed up calculated from equation (3) is shown. Figure 3 shows the wall clock time measured during simulations which is the time elapsed between the start of the computation process till it ends.

As can be observed, the maximum speed up using OpenFOAM with full machine CPU capacity was 9.78, while using RapidCFD with 4 CPU threads plus 4 GPUs was 20.53. This value surpasses the previously reported ones in research works related to GPU applications in CFD [22, 25, 26]. This could be a consequence of the test case size, since CPUs and GPUs work in a better way in parallel as the size problem increases.

![Figure 2. Speed up for BEP condition using multiple number of threads.](image1.png)  
![Figure 3. Wall clock time for BEP condition using multiple number of threads.](image2.png)

Validation of CFD results has huge relevance if desired to have a reliable model in future optimization processes. Therefore, validation with experimental data were done for pressure difference across the entire turbine, torque generated by runner, net head and hydraulic efficiency.

The values of differential pressure and torque were directly extracted from CFD simulations at inlet, outlet and runner patches respectively. Regarding net head $h_{net}$ and hydraulic efficiency $\eta_H$ these were calculated using additional data as follows:

$$h_{net} = \frac{p_{in} - p_{out}}{\rho g} + \frac{v_{in}^2 - v_{out}^2}{2g} + (z_1 - z_2)$$ \hspace{1cm} (4)

$$\eta_H = \frac{T \omega}{\rho g h_{net} Q}$$ \hspace{1cm} (5)

Where $p_{in} - p_{out}$ represents the differential pressure across the turbine, $v_{in}^2 - v_{out}^2$ represents difference in velocity magnitude at inlet and outlet, $z_1 - z_2$ is the height difference between inlet and outlet, $T \omega$ is the power produced by the runner and $\rho g h_{net} Q$ is the available power in the system.

As can be observed in Table 3 differential pressure, runner torque and net head are over estimated by OpenFOAM and RapidCFD; however, the hydraulic efficiency is under estimated. This behavior could be consequence of the diffusive nature of the first order discretization scheme used in the convective terms. However, the CFD data can be considered as a good initial approximation. Also it can be observed that there is not big difference between the results from RapidCFD and OpenFOAM despite the number of threads used for the solution in both cases.

| Differential pressure (kPa) | Experiment | RapidCFD | Diff. (%) | OpenFOAM | Diff. (%) |
|----------------------------|------------|----------|-----------|-----------|-----------|
| 114.9800                   | 122.4897   | 6.13%    | 123.4399  | 6.85%     |
| Runner torque (N-m)        | 628.4100   | 7.60%    | 679.3510  | 7.50%     |
| Net head (m)               | 11.9100    | 13.76%   | 13.9072   | 14.36%    |
| Hydraulic efficiency (%)   | 92.61      | -6.40%   | 86.34     | -7.26%    |

| Wall clock time [s]        | Number of threads |
|----------------------------|-------------------|
| 0                            | 1                 |
| 50,000                      | 4                 |
| 100,000                     | 32                |
| 150,000                     | 11520             |
| 200,000                     | 1                 |
| 250,000                     | 4                 |
| 300,000                     | 32                |
| 350,000                     | 11520             |
| 400,000                     | 1                 |
| 500,000                     | 4                 |
| 1,000,000                   | 32                |
| 1,500,000                   | 11520             |

Table 3. Experimental vs. CFD data comparison for BEP.
5.2. Whole range of operating conditions

CFD simulations for the rest of operation conditions and their variations for 1st and 2nd workshops were carried out only in RapidCFD. The required time and number of time steps (iterations) to reach convergence were close to each other and a similar behaviour in relation to the speed up is expected. In Figure 4 it is shown the simulation time required for each operation condition. Even when the HL operating condition took more time to be solved, it is not the maximum time by time step, but the PL1 operating condition (Figure 5), the fastest solved case. This shows how CFD computations are affected by the geometry, mesh, boundary conditions and discretization schemes when small changes occur; consequently, a set up for one condition cannot always work for others, which will be shown further.

![Figure 4](image-url) Wall clock time for different operation conditions using RapidCFD.

![Figure 5](image-url) Time per time step for different operation conditions using RapidCFD.

Similar as previously done, experimental data were compared to CFD results for all operation conditions. Differential pressure, torque, net head and hydraulic efficiency were evaluated. Figure 6 shows how the differential pressure is in most of the operating conditions over predicted by the CFD at exception of PL1 and HL.

It could be assumed that the runner torque values will follow the same tendency as the differential pressure in CFD data, i.e., it would be over predicted for all operation conditions, it is not the case though. As shown in Figure 7, torque for PL1 condition is dramatically under predicted, the experimental value is 144.06 N·m, while the CFD value is 4.59 N·m. This could be attributed to multiple causes, one of them is the discretization scheme used for the convective terms, and other could be the turbulence model or even the boundary conditions.

![Figure 6](image-url) Differential pressure values for different operation conditions using RapidCFD vs. experimental data.

![Figure 7](image-url) Runner torque values for different operation conditions using RapidCFD vs. experimental data.

Due to net head values are related to the differential pressure among other variables, values from the CFD are close to the experimental ones, however they are over predicted for all operating conditions as can be observed in Figure 8. The combined influence of the runner torque and net head
calculations by the CFD produces that hydraulic efficiency values fall below the experimental ones in all cases. This trend is more visible for the PL1 condition in Figure 9, where the experimental value is 71.69% and 2.20% for the CFD, the main reason is the poor approximation of runner torque.

Figure 8. Net head values for different operation conditions using RapidCFD vs. experimental data.

Figure 9. Hydraulic efficiency values for different operation conditions using RapidCFD vs. experimental data.

Related research papers [27-30] reported significant discrepancies respect experimental data for hydraulic efficiency values obtained from CFD at PL1 condition (guide vane angle=3.91°), while the other operation conditions values of hydraulic efficiency present better agreement with experimental values.

In this work the mayor discrepancies for the hydraulic efficiency are present precisely for the PL1 condition. We should mention that the referred research works over predict the values of hydraulic efficiency for all conditions while our results under predict them. It its worth to mention too, that in other works the discretization schemes used for convective terms are second order, which improves the accuracy of the CFD but tend to be oscillatory and unbounded if the mesh quality is not good enough or relaxation factor are aggressive.

6. Conclusions
The relevance of CFD in turbomachinery applications has grown up very fast over the last decades and its use is wide approved in design and optimization processes. However accurate results in short time require the coupling of high performance hardware and software, which in many cases is still a challenge.

A speed up analysis was carried out in order to demonstrate the huge potential of GPUs in the solution of CFD turbomachinery problems. It was shown that a speed up of 20x can be achieved when the RapidCFD code is used in combination with Nvidia Tesla K40 GPUs.

Validation of CFD results with experimental data has shown that differential pressure, runner torque and net head are over predicted, nevertheless hydraulic efficiency is under predicted. It has been inferred that this discrepancy is mainly caused by the discretization scheme deployed for solving the convective terms. Special attention was paid to the PL1 condition CFD results, which were the exception to the global tendency in this work, showing very bad approximations to the experimental data.

It is planned for future work to improve the accuracy of CFD by means of higher order discretization schemes for convective terms and different turbulence models. It could be assumed that these solutions have a bigger computational cost, however it is expected that solution times be shorter using GPUs than using CPUs, which has been demonstrated.

References
[1] Versteeg H K and Malalasekera W 2007 An Introduction to Computational Fluid Dynamics: The Finite Volume Method (Harlow, Essex: Pearson Education Limited)
[2] Sottas G and Ryhming I L 1993 3D-Computation of Incompressible Internal Flows. Proceedings of the GAMM Workshop held at EPFL, 13–15 September 1989, Lausanne, Switzerland ed G Sottas and I L Ryhming (Braunschweig/Wiesbaden: Vieweg)

[3] Engström T, Gustavsson L and Karlsson R, 2001 Proceedings of Turbine-99 – Workshop 2 on draft tube flow in Älvkarleby, Sweden, 18–20 June, 2001 ed T Engström, L Gustavsson and R Karlsson (Älvkarleby: Lulea University of Technology)

[4] M. Cervantes, T. Engstöm and L. Gustavsson 2005 Proceedings of the third IAHR/ERCOFTAC Workshop on draft tube flows ed M Cervantes, T Engstöm and L Gustavsson (Lulea University of Technology)

[5] Francis 99 workshop of the Norwegian Hydropower Centre https://www.ntnu.edu/nvks/ francis-99 accessed: October 15th, 2020

[6] Kurosawa S, Lim S M and Enomoto Y 2010 Virtual model test for a Francis turbine 25th IAHR Symposium on Hydraulic Machinery and Systems IOP Conf. Ser.: Earth Environ. Sci. 12 012063

[7] Choi H J, Zullah M A, Roh H W, Ha P S, Oh S Y and Lee Y H 2013 CFD validation of performance improvement of a 500 kW Francis turbine Renewable Energy 54 111-23

[8] Keck H and Sick M 2008 Thirty years of numerical flow simulation in hydraulic turbomachines, Acta Mechanica 201 211–29

[9] Niemeyer K E and Sung C J 2014 Recent progress and challenges in exploiting graphics processors in computational fluid dynamics The Journal of Supercomputing 67 528-64

[10] Navarro C, Hitschfeld-Kahler N and Mateu L 2014 A survey on parallel computing and its applications in data-parallel problems using GPU architectures Communications in Computational Physics 15 285-329

[11] Posey S, See S and Wang M 2015 GPU Progress and Directions in Applied CFD Eleventh International Conference on CFD in the Minerals and Process Industries ed Editors : C B Solnordal, P Liovic, G W Delaney, S J Cummins, M P Schwarz and P J Witt (Melbourne: CSIRO)

[12] Afzal A, Ansari Z, Faizabadi A R and Ramis M K 2017 Parallelization Strategies for Computational Fluid Dynamics Software: State of the Art Review Archives of Computational Methods in Engineering 24 337–63

[13] Stefan D and Rudolf P 2015 Proper Orthogonal Decomposition of Pressure Fields in a Draft Tube Cone of the Francis (Tokke) Turbine Model Journal of Physics: Conference Series 579 012002

[14] Jošt D, Škerlavaj A, Morgut M, Mežna P and Nobile E 2015 Numerical simulation of flow in a high head Francis turbine with prediction of efficiency, rotor stator interaction and vortex structures in the draft tube Journal of Physics: Conference Series 579 012006

[15] Lenarcic M, Eichhorn M, Schoder S J and Bauer C 2015 Numerical investigation of a high head Francis turbine under steady operating conditions using foam-extend Journal of Physics: Conference Series. 579 012008

[16] Stoessel L and Nilsson H 2015 Steady and unsteady numerical simulations of the flow in the Tokke Francis turbine model, at three operating conditions Journal of Physics: Conference Series 579 012011

[17] Rathnayake T, Jayasena S and Narayana M 2017 Openfoam on GPUs using AMGX Proceedings of the 25th High Performance Computing Symposium (San Diego, CA: Society for Computer Simulation International)

[18] AlOnazi A 2014 Design and Optimization of OpenFOAM-based CFD Applications for Modern Hybrid and Heterogeneous HPC Platforms Master thesis (Thuwal, Kingdom of Saudi Arabia: King Abdullah University of Science and Technology)

[19] Jasiński D 2015 Adapting OpenFOAM for massively parallel GPU architecture The 3rd OpenFOAM User Conference (Stuttgart, Germany: ESI Group)

[20] simFlow CFD Software Atizar/RapidCFD-dev https://github.com/Atizar/RapidCFD-dev.
[21] Nocente A, Arslan T, Jasinski A and Nielsen T K 2016 A Study of Flow inside a Centrifugal Pump: High Performance Numerical Simulations Using GPU cards 16th International Symposium on Transport Phenomena and Dynamics of Rotating Machinery (Hawaii, Honolulu)

[22] Molinero D, Galván S, Pacheco J and Herrera N 2019 Multi GPU Implementation to Accelerate the CFD Simulation of a 3D Turbo-Machinery Benchmark Using the RapidCFD Library Supercomputing. ISUM 2019. Communications in Computer and Information Science vol 1151 ed M Torres and J Klapp (Springer)

[23] Galván S, Reggio M and Guibault F 2011 Assessment Study of k-e Turbulence Models and Near-Wall Modeling for Steady State Swirling Flow Analysis in Draft Tube Using Fluent Engineering Applications of Computational Fluid Mechanics 5 459-78

[24] Khajeh-Saeed A and Perot J B 2012 Computational Fluid Dynamics Simulations Using Many Graphics Processors Computing in Science & Engineering 10-19

[25] He Q, Hongli C and Jingchao F 2015 Acceleration of the OpenFOAM-based MHD solver using graphics processing units Fusion Engineering and Design 101 88-93

[26] Malecha Z, Mirosław Ł, Tomczak T, Koza Z, Matyka M, Tarnawski W and Szczerba D 2011 GPU-based simulation of 3D blood flow in abdominal aorta using OpenFOAM Archives of Mechanics 63 137-61

[27] Trivedi C, Cervantes M J, Gandhi B K and Dahlhaug O G 2013 Experimental and Numerical Studies for a High Head Francis Turbine at Several Operating Points Journal of Fluids Engineering 135

[28] Trivedi C, Cervantes M J and Dahlhaug O G 2016 Numerical Techniques Applied to Hydraulic Turbines: A Perspective Review Applied Mechanics Reviews 68

[29] Trivedi C, Cervantes M J and Dahlhaug O G 2016 Experimental and Numerical Studies of a High-Head Francis Turbine: A Review of the Francis-99 Test Case Energies 9 1-24

[30] Trivedi C and Cervantes M J 2016 State of the art in numerical simulation of high head Francis turbines Renew. Energy Environ. Sustain 1 1-5