

## Speed Comparison of Single Equation Method based on Saturated Steam Water Spaces Models

Jacob Philip  
*Simulator Group*  
Nuclear Power Corporation of India Limited  
Mumbai, India  
jkphilip@gmail.com

Faruk Kazi, Harivittal Mangalvedekar  
*Electrical Engineering Department*  
VJTI, Mumbai University  
Mumbai, India

**Abstract** - Simulators are extensively used as a tool for industrial training, design and development. The overall improvement in computational speed of simulator is very important and can be improved by redesigning the mathematical simulation models or by using faster computational platforms. Modern day simulators are using cloud computing platforms and other high performance computing systems. The complex algorithms employed in simulation of steam-water spaces and steam tables resulted in longer computational time. The use of cloud computing for simulation makes essential the optimization of cost of computing and cost of transmission. The quasi-numerical, quasi-steady state, Single Equation Method based saturated steam water space modeling can be employed to improve the speed of computation and to suit the new computing environments. This makes the simulator faster and cost effective. Experiments using different computing platforms show that the proposed single equation based quasi-numerical, quasi-steady state is around eight times faster than the steam-table iteration based modeling approach.

**Keywords** - simulation; saturated steam water space; quasi-steady state modeling; steam tables; performance optimization; cloud computing, performance comparison

### I. INTRODUCTION

Most of the heavy industries use steam for heating purposes and as a power fluid. Fossil and Nuclear Power Plant (NPP) cycles use water & steam as their primary process fluid for generation of electricity. There are many steam and water based equipments operating at saturated steam pressure/temperature used in fossil fired power plant cycles like feed water heaters, steam condensers, steam flash tanks and de-aerators. NPPs, additionally need pressurizers for maintaining high pressure in Heat Transport System (HTS) [1] and bleed condensers (BCD) for bringing down the temperature of pressurized HTS water from 300°C to 200°C [1],[2]. These equipments contain steam & water at saturation temperatures and may be collectively termed as Saturated Steam Water Space (SSWS) systems. SSWS models heavily depend on the saturated steam-tables (SST). The SSTs relate the properties of specific volume, density, specific enthalpy of steam & water at different temperatures [3]. The SST is a large table which needs to be accessed continuously to match the properties of steam and water so as to get the best fit for the operating temperature and pressure. Iteration and interpolation is generally carried out to obtain the best fit of operating temperature and pressure. To reduce the amount of iteration & interpolation, large steam-table needs to be attached to the computer systems which involve consumption of computation time and memory [4]. Hence it is prudent to avoid or at least reduce the use of steam-tables to reduce the computation time [11].

Towards this one effective method is to convert the steam-table into a set of equations. Least squares approximation based formulas for steam density, specific liquid enthalpy properties are used in [4]. Simple non-linear functions for superheated and sub-cooled water properties in the range of 0.085 to 21.3MPa are used in [5]

International Atomic Energy Agency, IAEA, recommends the use of mathematical equations for estimating the thermodynamic properties of light water & heavy water including their vapor states [6],[7]. To increase the computational speed for industrial applications, International Association of Properties of steam (IAPS) formulations developed an industrial formulation IAPS-IF97 based on the computationally intensive IAPS-95 formulation [6].

In a control room operator training simulator, all the calculations pertaining to all the models of all systems needs to be computed within the cycle time of computation, say 100ms or 200ms. In a power plant simulator, especially NPPs, many SSWS based models are computed every cycle. Some critical equipment models are computed many times within one cycle to improve its accuracy. Therefore, in FPPs as well as NPPs, the SSWS equipment simulation forms a considerable simulation load.

Cloud computing (cloud) is a computing environment offered as a service at a cost to the user. The costs are based on the consumption of resources like processor time, network bandwidth/data, storage and memory. Cloud can be considered as a large set of virtual computing machines (VM) working together across a network. To save cost, cloud computing environment needs optimized models to take advantage of the cloud benefits like scalability & multiprocessing capability and to overcome its limitations in cost of overheads in computing, network traffic, computation time, and processor time [8],[9],[10].

A fossil powered boiler system, including its sub-models, was simulated with MATLAB as the back-end & LABVIEW based client on a cloud computing platform. Differential Algebraic Equations were used for the same [11]. Conventional domain specific models need suitable model adaptation when these simulators are ported on a cloud

platform. A very good example for domain specific model adaptation is the development of polynomial multiplication of the order of millions using FFT algorithms for cloud and multiprocessor architectures. This was developed especially for higher order polynomials used in simulation of aero-space, signal processing and image processing [12]

SSWS modeling using combined properties of steam & water as employed in [13] greatly simplifies the SSWS modeling and reduces the overall turn around time for SSWS modelers. Faster model computations, to a great extent, can reduce the overall computational cycle-time of simulators. Reduced cycle time may be used in increasing the accuracy, depth or width of simulation [13][14][15].

This paper proposes a numerical approach with polynomial approximation functions to replace each steam & water properties of SSWS. These polynomial approximation functions can be directly used in modeling of SSWS in quasi-steady state. This makes the model suitable for simulations in normal as well as cloud computing environments. We had conducted extensive simulation experiments for comparing the performance of steam table based iterations and approximation methods. The methodology adopted in avoiding errors in comparison is also explained. Results of the experiments for time performance comparison of steam table iteration based approach and proposed quasi-steady state approach is also discussed.

The paper refreshes the basic equation used in SSWS from [15]. Final enthalpy equation of SSWS along with the approach to derive the approximation equations are reproduced from [15] in Section-II. Performance comparison in virtual and physical machines is given in Section-IV along with details of simulation experiment methodology adopted.

II. SINGLE EQUATION BASED SSWS MODEL

Consider a tank of volume  $v$  operating at saturation temperature  $T$  and pressure  $P$ . Let mass in Kg of steam & water be denoted by  $m_s$  &  $m_w$ , specific enthalpy by  $h_s$  &  $h_w$ , density by  $d_s$  &  $d_w$ , and volume of steam and water by  $v_s$  &  $v_w$ . Let the total mass  $m$  in kg is  $m=m_s + m_w$  and total SSWS volume is  $v=v_s + v_w$ . Then total enthalpy (heat)  $H$  in the tank/SSWS shall be [13],[14],[15]

$$H = m_s h_s + m_w h_w \tag{1}$$

Further  $v_s = m_s/d_s$  and  $v_w = m_w/d_w$ . Substituting and rearranging, we get after eliminating  $m_s, m_w, v_w,$  &  $v_s$

$$H(d_s - d_w) = md_s h_s - vd_s d_w h_s + vd_s d_w h_w - md_w h_w \tag{2}$$

$$md_s h_s - vd_s d_w h_s + vd_s d_w h_w - md_w h_w - H(d_s - d_w) = 0 \tag{3}$$

Equation (3) is termed as the ‘Single Equation’ as this describes the whole SSWS in one equation. The properties  $d_s, d_w, h_w,$  &  $h_s$  are some functions of the saturation temperature  $T$ . The temperature at which (3) is satisfied shall give the saturation temperature. Here  $H, m, v$  are known

quantities,  $d_s, d_w, h_w,$  &  $h_s$  are drawn from SST as per the resultant saturation temperature of the SSWS [3][14][15].

A. Relation between properties and temperature

Steam table provides the steam/water properties at different temperatures. Each property has a different range. The relationship of  $d_s, d_w, h_w,$  &  $h_s$  as a variation of temperature in degree centigrade are shown in Fig-1. Properties  $d_s, d_w, h_w,$  &  $h_s$  can be considered as functions of SSWS temperature though their relationships are complex and different[4],[3],[13],[14],[15].

Rewriting (3),  $d_s, d_w, h_w,$  &  $h_s$  with as functions of temperature  $T$  (Kelvin or Degree Centigrade) we get

$$H = [md_s(T)h_s(T) - vd_s(T)d_w(T)h_s(T) + vd_s(T)d_w(T)h_w(T) - md_w(T)h_w(T)] \div [d_s(T) - d_w(T)] \tag{4}$$

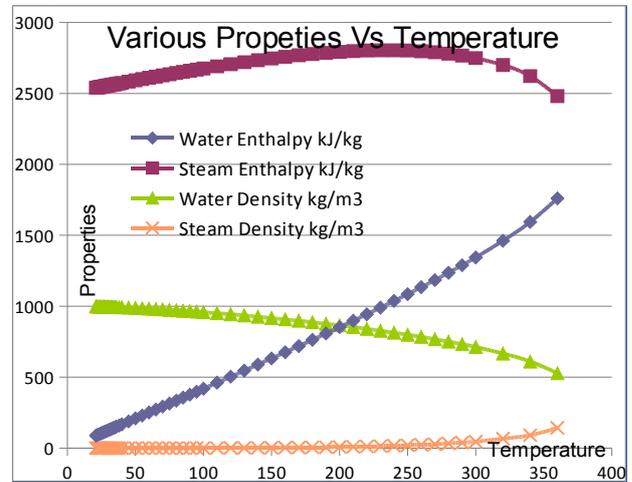


FIGURE 1: ACTUAL STEAM/WATER PROPERTIES VS TEMPERATURE[15]

B. Deriving Approximation functions

Consider the property, water density  $d_w$  and its behavior respect to  $T$ . Now by normalizing temperature  $T$  in the range of 20 to 360°C range to normalized temperature  $T_N$  (an  $N$  in the suffix denotes the variable is normalized).

$T_N = (t-20)/(360-20)$ , where  $T_N$  is normalised in Per Unit (PU) range.

Similarly the saturated liquid density  $d_w$  decreases from 998 to 528.6 kg/m3 as temperature varies from 20 to 360°C. Normalizing  $d_w = (d_{Nw} - 998)/(528.6 - 998)$

The a relationship between  $T_N$  and  $d_{Nw}$  can be described using a fourth order polynomial of  $T_N$  (derived using Least Square Method) as

$$d_{Nw}(T_N) = d_{Nw}(T_N, 4) = 1.39612(T_N)^4 - 1.95123(T_N)^3 + 1.41663(T_N)^2 + 0.121467(T_N)^1$$

Here  $(T_N, m)$  represents an  $m^{th}$  order polynomial  $\sum_{k=1}^{k=m} A_k T^k$  where  $A_k$  are constants.

This approximation polynomial introduces a maximum error -1.51% at 360°C. Fig-2 shows the comparison of fourth order polynomial & simple  $(T_N)^2$  approximation of normalized water density with the SST based normalized water density for a normalized temperature of 20 to 360°C. Here fourth order polynomial approximation provides a very accurate representation of actual water density[14],[15].

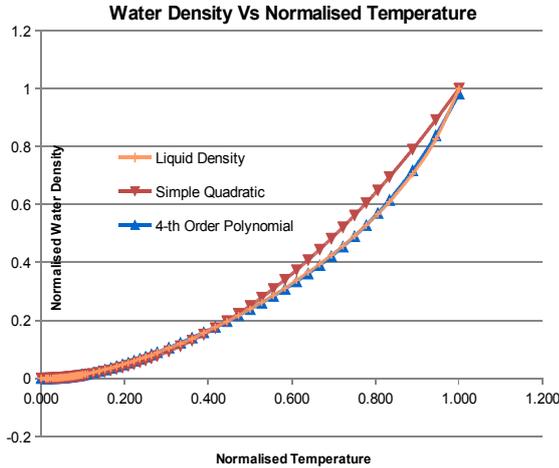


FIGURE 2: WATER DENSITY, POLYNOMIAL AND QUADRATIC FIT[15]

C. Approximation functions for Simulation of Steam/Water Properties

Specific enthalpy of water can be expressed as  $h_w=A \cdot h_{Nw}(T_N,4)+B$  with maximum error of 0.75% at 360°C where

$$h_{Nw}(T_N,4) = 0.366(T_N)^4 - 0.332(T_N)^3 + 0.1197(T_N)^2 + 0.839(T_N)^1 \tag{5}$$

Similarly  $h_{Ns}$  can be expressed as  $h_{Ns}(T_N,6)$  and that of  $d_{Ns}$  as  $d_{Ns}(T_N,4)$ .

III. SIMULATION OF SSWS ENTHALPY USING APPROXIMATED EQUATION

Each property term  $d_s, d_w, h_w, & h_s$  in SSWS equation as described in (3) is a function of SSWS temperature T. Therefore as explained in Section-2.2,  $d_{Ns}, d_{Nw}, h_{Nw}, & h_{Ns}$  are functions of  $T_N$ . Now expanding each term  $d_s, d_w, h_w, & h_s$  as a function of SSWS normalised temperature  $T_N$  [15]

$$[m \cdot d_s(T_N)h_s(T_N) - v \cdot d_s(T_N)d_w(T_N)h_s(T_N) + v \cdot d_s(T_N)d_w(T_N)h_w(T_N) - m \cdot d_w(T_N)h_w(T_N)] - [H \cdot (d_s(T_N) - d_w(T_N))] = 0 \tag{6}$$

Consider an SSWS with volume  $v$  in =12m<sup>3</sup>and mass of light water  $m=1200$  kg. The Enthalpy H in kJ of the SSWS for the temperature from 100 - 360°C is calculated using(4). This considers that no mass is added to or removed from the

system during the time of computation and the system has attained saturation conditions.

Fig-3 shows the SSWS total enthalpy values obtained using approximation functions and enthalpy values obtained from the steam table (SST) for temperatures in the range of 100 to 360°C. It is clear that the trend of calculated enthalpy using approximation functions and actual enthalpy computed with steam-Table iteration-interpolation are similar with limited error [13],[14],[15].

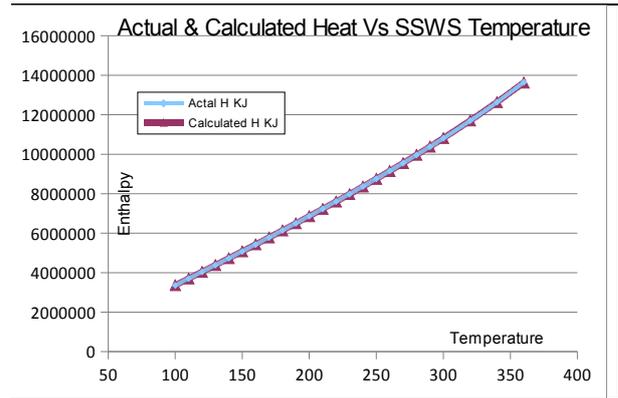


FIGURE 3: APPROXIMATE &ACTUAL HEAT VS TEMPERATURE[15]

In a real life SSWS modeling scenario, it is important to compute the saturation temperature of an SSWS, when a disturbance in mass, volume or enthalpy is applied to the system[15]. These disturbances are introduced by:

- Disturbance in mass m: steam or water is added to or removed from the SSWS
- Disturbance in volume v: change in shape/size of the SSWS volume or due to expansion in pipes
- Disturbance in enthalpy H: heating or cooling by external heat sources. This disturbance can also occur when mass or steam is added/removed from the system.

The disturbances in mass m, volume v or enthalpy H can be computed and appropriately applied to the SSWS equation(6). After computing and accommodating the disturbances, (6) can be solved for new normalized temperature  $T_N$ . SSWS temperature T can be computed from  $T_N$  [15].

IV. SPEED COMPARISON, RESULTS AND ANALYSIS

A. Comparison of Computing Speeds in Virtual Machines

Simulation experiments were conducted to compare the speed of computation of SSWS equations using single equation based, quasi-normalised quasi-numerical approach and the solution obtained using Saturated Steam Tables iteration and interpolation.

Simulations were carried out using various personal computers (PC) as well as virtual machines (VM). The VMs were created using VMWare Player (VMW) & Windows Virtual PC (WVPC). In general GNU Compiler Collection (gcc) compiler was used for compiling C programs. In Microsoft Windows, two environments were used, first using minGW (Minimalist GNU for Windows) and second with

CYGWin (a Unix-like environment & command-line interface for Microsoft Windows) platforms.

```

Define max_repeat = 5
Define max_iter = 900 Million
START
For num_repeat 1 to max_repeat
{
  Print cur_time
  Call Function polynomial approx()
  Print cur_time
  Call Function steam_table_iteration()
  Print cur_time
} End num_repeat FOR
END
Function polynomial approx()
{
  For num_iter 1 to max_iter
  {
    Compute SSWS T by Polynomial_approximation
  }
}
Function Steam_Table_iteration()
{
  For num_iter 1 to max_iter
  {
    Compute SSWS T by Steam Table iteration
  }
}
    
```

FIGURE-4 PSUEDO-CODE FOR SPEED COMPARISON ROUTINE

**B. Methodology Adopted for Simulation Experiments**

For experiments instead of using a full fledged steam table, a steam-table of just 14 rows (100-360°C, at intervals of 20°C) in an array form was used. The fit of temperature was derived using continuous iteration & interpolation within the 14 row SST. We employed Newton-Raphson based solution with four iterations for quasi-numerical approach. Four iterations provided accurate solutions up to four decimals for quasi-numerical approach[15]. The structure/pseudo-code for the program is given in Fig-4.

As in Fig-4, two functions were created, one for finding out the results in array format with steam table iteration & interpolation method and the other for finding out the resultant temperature using Polynomial approximation-Newton Raphson method. Before the actual speed comparison experiments were conducted, the two functions were tested for accuracy of results with single iteration. This was obtained by using print functions (printf function) in C program. This program was enhanced to call the two functions, steam-table-iteration & polynomial-approximation function, to be executed in two loops to compare their speeds. The number of times each loop executed was equal and the two functions were executed one after the other sequentially. Any variation in computing speed affected the two functions together, steam table iteration and polynomial approximation in a similar manner. This helped in reducing the error arising from variation in speed of computation of a machine.

We used simple time function (ctime) available in C library to measure the time. For each loop, time was printed before and after the loop. The ctime function was used up to the level of seconds and this necessitated to overcome the

inaccuracies of time bunching, the number of iterations needed to be increased, especially for the polynomial-approximation method function in fast processors. Therefore to increase the resolution of computation time, both modules were executed for 900 million times in a single program sequentially. Here also, a two pronged approach was used; in first one, within the same program executing the two loops multiple times as in Fig-4 and in the second, the two functions executing only once.

```

start_time, STEAM_TABLE, Thu Jan 29 23:28:42 2015
, End_time, STEAM_TABLE, Thu Jan 29 23:31:49 2015

start_time, SSWS_APPRX, Thu Jan 29 23:31:49 2015
, End_time, SSWS_APPRX, Thu Jan 29 23:32:09 2015

start_time, STEAM_TABLE, Thu Jan 29 23:32:09 2015
, End_time, STEAM_TABLE, Thu Jan 29 23:35:12 2015

start_time, SSWS_APPRX, Thu Jan 29 23:35:12 2015
, End_time, SSWS_APPRX, Thu Jan 29 23:35:33 2015

start_time, STEAM_TABLE, Thu Jan 29 23:35:33 2015
, End_time, STEAM_TABLE, Thu Jan 29 23:38:47 2015

start_time, SSWS_APPRX, Thu Jan 29 23:38:47 2015
, End_time, SSWS_APPRX, Thu Jan 29 23:39:08 2015

start_time, STEAM_TABLE, Thu Jan 29 23:39:08 2015
, End_time, STEAM_TABLE, Thu Jan 29 23:42:20 2015

start_time, SSWS_APPRX, Thu Jan 29 23:42:20 2015
, End_time, SSWS_APPRX, Thu Jan 29 23:42:41 2015

start_time, STEAM_TABLE, Thu Jan 29 23:42:41 2015
, End_time, STEAM_TABLE, Thu Jan 29 23:45:54 2015
    
```

FIGURE-5 SAMPLE EXPERIMENT RESULT FROM MINT LINUX VM

TABLE-I PROCESSED SAMPLE RESULT FROM MINT UNIX VM

PC/VM	Enviro- nment/ Compi- ler	Experi- ment sequence number	Steam- Table Iter time (S)	Polyno- mial apprx Time (S)	Speed Up (%)
VM Mint AMD	gcc	1	201	30	670.00
VM Mint AMD	gcc	2	192	24	800.00
VM Mint AMD	gcc	3	173	24	720.83
VM Mint AMD	gcc	4	187	20	935.00
VM Mint AMD	gcc	5	183	21	871.43
VM Mint AMD	gcc	6	194	21	923.81
VM Mint AMD	gcc	7	192	21	914.29
VM Mint AMD	gcc	8	193	20	965.00

From the final program, all the print functions within the loops for printing the result were removed to avoid the load on the Input/output system. Since the programs were already

validated and only comparison of computational speed of the two functions was of prime concern, the print statements for results were avoided.

Fig-5 shows the sample result obtained from a machine by continuously running the two functions sequentially for five times. Table-I shows the details of results obtained from a sample experiments conducted on one machine.

Simulation experiments were conducted randomly in different machines, either in batches of five or as isolated single experiments. The results were later processed to find the SpeedUp similar to Table-I and average SpeedUp and standard deviation as in Table-II. The SpeedUp was calculated as

$$\text{SpeedUp (\%)} = (\text{SteamTable\_time}/\text{PolynomialTime}) * 100$$

Table-2 shows the comparison of average computational speed-up in % obtained using Polynomial approximation method against steam-table iteration method. The details of the various virtual machines & PCs where the experiments were carried out are also given in the Table-II.

The minimum average SpeedUp observed was 733.85 and the maximum average SpeedUp obtained was 1028.54 as seen in Table-II. Thus approximation functions based models for SSWS are around 7 to 10 times faster than steam-table iteration- interpolation models as evident from Table-2. Standard deviation of SpeedUp is also shown in Table-2. This variation in SpeedUp in a particular machine may be attributed to the then state of the machine with respect to computational and other Input-Output operational load on the processors & machine at that point of time.

TABLE II: COMPARISON OF SPEED OF COMPUTATION

Machine Details Machine Type, Operating System, Virtualization Platform	Number of Simulation Experiments	Standard Deviation	Average SpeedUp (%)
PC, AMD FX-6100 6core, Windows-7, 4GB RAM, minGW	11	24.80	899.59
PC, AMD FX-6100 6core, Windows-7, 4GB RAM, CYGWin	11	36.25	858.50
VM, Live Linux Mint-14, 1GB RAM, VMW	8	108.8	850.04
VM, Kubuntu, 1GB RAM, VMW	5	36.15	792.09
VM, SUSE, 1GB RAM, VMW	10	21.82	871.59
VM WinXP Mode, 512MB RAM, WVPC, CYGWin	5	39.33	741.55
PC Intel Celeron Windows XP, 256MB RAM, CYGWin	10	77.54	1028.54
PC Intel I-5, 4 Core Windows-7, 3GB RAM, CYGWin	5	35.5	733.85
PC Intel Pentium Dual Core, Windows Vista, 2GB RAM, minGW	5	16.51	781.44

## V. CONCLUSION

The simulators of industries using steam as power fluid use many Saturated-Steam-Water-Space equipments and their simulation models. This paper presents the results of the experiments conducted to compare the computational speeds using the quasi-steady state, quasi-numerical single equation based numerical solutions method and Steam table based iteration & interpolation methods. From the results of the tests on different virtual machines & personal computers, it is evident that the proposed quasi-numerical approximation method is around eight times faster compared to steam-table iteration based approach as seen from our experiments. Thus the proposed approach may reduce the overall cycle time for simulation, thereby providing increased depth & width in the process simulation. The contribution of this paper is in opening a new path in simulation of SSWS to increase the speed of computation of SSWS models using an efficient as well computational friendly modeling approach. This efficient & computational friendly approach may reduce the computational overheads, penalty of overheads and improving the performance of simulators using cloud computing.

## ACKNOWLEDGMENT

This is a part of the project for “Adaptation of Parallel Processing in Large Physical Process Simulation” funded by Dept of Atomic Energy of India (DAE) through Board of Research in Nuclear Sciences (BRNS) wide sanction # 2012/36/52-BRNS and World Bank funding under TEQIP-II (Sub-component 1.2.1) for Center of Excellence (CoE) in Complex & Nonlinear Dynamical Systems (CNDS). We express our gratitude for kind funding.

- [1] “CANDU fundamentals training course”, Module-15, Heat Transport Auxiliary Systems, COG-CANTEACH Canadian Nuclear Safety Commission Technical Documents Library, 2003.
- [2] M MacBeth, “Digital instrumentation & control”, Work Book-6&7, CANDU Owners Group COG-CANTEACH, Chulalongkorn University Technical Documents, 2003
- [3] MD Koretsky, "Properties of steam (SI units), engineering and chemical thermodynamics", John Wiley & Sons, 2004
- [4] WJ Garland, JD Hoskins, “Approximate functions for the fast calculation of light water properties at saturation”, International Journal on Multiphase Flow, pp333-348, 1988, Elsevier.
- [5] WJ Garland and BJ Hand, “Simple functions for the fast approximation of light water thermodynamic properties”, Nuclear Engineering and Design, 1989, pp21-34, Elsevier Science Publishers.
- [6] “Thermophysical properties database of materials for light water reactors and heavy water reactors”, IAEA-TECDOC-1496, International atomic energy agency, 2006
- [7] “Thermophysical properties of materials for water cooled reactors”, IAEA-TECDOC-949, International atomic energy agency, 1997
- [8] “Cloud computing: a practical approach”, Tata McGraw Hill Education, ISBN-13-9780070683518, 2009.
- [9] Yashpalsinh Jadeja, Kirit Modi, “Cloud computing concepts, architecture and challenges”, 2012 International conference on Computing, Electronics and Electrical Engineering [ICCEET], pp.877-880, 2012
- [10] Ajith Ranabahu et al, “Application portability in cloud computing: an abstraction driven perspective”, IEEE transactions on Service Computing, Issue-99, In Press

- [11] PS Saikrishna, R. Pasumarthy, P.Raman, S, Chakrabarty, L Sivakumar, "Simulation of a boiler model in a cloud environment", ACODS, Feb 2012, Bangalore, India.
- [12] Ganesh Iyer, Bhardwaj Veeravalli, SG Krishnamoorthy, "On handling large-scale polynomial multiplications", IEEE Transactions on Aerospace and Electronic Systems, Vol-48, pp. 820-831, Jan-2012
- [13] Jacob Philip, Faruk Kazi, Harivittal Mangalvedekar, "Mathematical modeling of steam-water-vapor-space with mfact-vfact approach using combined steam properties", Asia modeling symposium (AMS-2013), pp 252-257, Hong Kong, 2013
- [14] Jacob Philip, Harivittal Mangalvedekar, Faruk Kazi, "Modeling of coupled saturated steam-water spaces using quasi-steady state, quasi-numerical approach", 2014 Fifth International Conference on Intelligent Systems, Modelling and Simulation(ISMS-2014), pp 474-478, Malasia, 2014
- [15] Jacob Philip, Faruk Kazi, Harivittal Mangalvedekar, "Quasi steady state, quasi numerical modeling of saturated steam-water spaces using normalized steam & water properties for efficient computing", 2014 Fifth International Conference on Intelligent Systems, Modelling and Simulation(ISMS-2014), pp 675-679, Malasia, 201