



॥ त्वं ज्ञानमयो विज्ञानमयोऽसि ॥

INDIAN INSTITUTE OF TECHNOLOGY, JODHPUR, RAJASTHAN

UNDERGRADUATE RESEARCH AND INNOVATION PROGRAM 2013

**VIABILITY OF USING SODIUM AND POTASSIUM BASED SALTS AS HEAT
STORAGE MEDIUM**

Investigators:

Yash Ganatra

VIT University – Chennai

Dr. Prodyut Chakraborty

Dr. Laltu Chandra

IIT, Jodhpur - Rajasthan

25 May – 18 July 2013



DECLARATION OF STUDENT:

I, Yash Ganatra hereby declare that the report is an original work. The contents of the work have not been published before and reflect the work done during my Summer Internship at INDIAN INSTITUTE OF TECHNOLOGY, JODHPUR, RAJASTHAN from 25 May 2013 to 18 July 2013.

ACKNOWLEDGEMENTS:

I would like to thank my guides – Dr. Prodyut Ranjan Chakraborty and Dr. Laltu Chandra for selecting me for the Undergraduate Research and Innovation program at IIT Rajasthan. Without their support and vision, the project would not have got completed. I would like to thank them for being patient at all times. I would like to take this opportunity to thank Dr. Barun Pratiher for his support during the project. I am grateful for all the facilities and resources offered to me. It was a wonderful learning experience working in Fluid Mechanics and Heat Transfer Laboratory and for that I would like to thank Mr. Anurag. Lastly, I would like to thank my parents for their support and belief in me.

Date:

Signature:

Dr. Prodyut Chakraborty

Dr. Barun Pratiher – UGRI Coordinator

ABSTRACT:

Diminishing fossil fuel reserves and increasing effects of climate change have led to an unprecedented global interest in renewable energy sources. A leading candidate among them is electricity generation via concentrated solar power plants (CSP). In CSP solar energy is concentrated via large array of mirrors that focus sunlight on to a Heat Transfer Fluid (HTF). In India, Rajasthan has excellent potential to harness this due to the presence of direct solar radiation.

As a consequence of nighttime conditions, CSP are forced to undergo repeated startup and shut-down operations. Thus, Fossil based or other backup generation is needed. An alternative is to generate excess power at peak insolation for temporary storage that is later dispatched during insufficient periods of insolation. Thermal Energy storage uses an excess volume of hot HTF to maintain a high temperature reservoir. For implementation, sensible heat storage is used. Traditionally a mixture of organic compounds, are used as HTF. The synthetic oil offers best combination of low freezing point (12°C) and upper temperature limit (393°C). This effectively limits steam temperature to 370°C , which limits overall efficiency of the Rankine Cycle [2].

This project aims at studying the feasibility of using molten salt as thermal energy storage medium. Molten salts have upper temperature limits of 500°C , lower unit cost and lower vapor pressure. The primary disadvantage of molten salt is their high freezing point ranging from 130°C to 230°C . Considerable care must be taken to prevent freezing.

LIST OF FIGURES:

Figure 1	Concentrating solar power with heliostats	1
Figure 2	Schematic of a photovoltaic system	1
Figure 3	Annual average global insolation map of India showing hotspots	2
Figure 4	Forms of energy storage	3
Figure 5	Different stages in design of Latent Heat Thermal Energy Storage Systems	6
Figure 6	Boundary Conditions	11
Figure 7	Entities in FLUENT	13
Figure 8	Solution Procedure for Pressure Based solver	14
Figure 9	Geometry made in ANSYS Design Modeller	15
Figure 10	Meshed Geometry	15
Figure 11	Imported mesh in FLUENT	16
Figure 12	User Defined Memory Allocations	16
Figure 13	User Defined Scalar	17
Figure 14	Compiled UDF window	17
Figure 15	Directory of stored UDF	17
Figure 16	Loading UDF	18
Figure 17	Hooking UDF	18
Figure 18	Defining Source Terms	19
Figure 19	Solution Methods	19
Figure 20	Solve Set Expert Command	20
Figure 21	Solution Initialization	20
Figure 22	Isometric view and section of storage system	21
Figure 23	Comparison of temperature contours from the code implemented (left) and the standard results (right)	
Figure 24	Comparison of stream function between code implemented (left) and reference results (right)	
Figure 25	Comparison of temperature contours from the UDS Transport equation (left) and the standard results (right)	22
Figure 26	Comparison of temperature contours from the code implemented (left) and the UDS Transport Equation (right)	22
Figure 27	Variation of Viscosity with temperature in Sodium Nitrate	23
Figure 28	Variation of Solid specific heat with temperature in Sodium Nitrate	24
Figure 29	Variation of Solid Thermal conductivity with temperature in Potassium Nitrate	24
Figure 30	Variation of Viscosity with temperature in Potassium Nitrate	25
Figure 31	Variation of Viscosity with temperature in Sodium and Potassium Nitrate Eutectic mixture	26

LIST OF EQUATIONS

Equation 1	Sensible Energy Storage	4
Equation 2	Latent heat storage integral form	4
Equation 3	Latent heat storage general form	4
Equation 4	Stefan Problem	8
Equation 5	General Form of Differential Equation of a conserved quantity	9
Equation 6	X-Momentum Equation	9
Equation 7	Y-Momentum Equation	9
Equation 8	Carman-Kozeny Equation	9
Equation 9	Energy equation in terms of enthalpy	10
Equation 10	Generalized energy equation in terms of temperature	10
Equation 11	Discretization equation	11
Equation 12	Enthalpy update scheme	11
Equation 13	Enthalpy update scheme general equation	12
Equation 14	User Defined Scalar Transport Equation solved by FLUENT solver	13

LIST OF TABLES

Table 1	Brief Classification of PCM	8
Table 2	Properties of Gallium	21
Table 3	Thermophysical Properties of Sodium Nitrate	23
Table 4	Thermophysical Properties of Potassium Nitrate	24
Table 5	Thermophysical Properties of Sodium and Potassium Nitrate (Eutectic)	24

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ACKNOWLEDGEMENTS	i
	ABSTRACT	a
	LIST OF FIGURES, EQUATIONS, TABLES	a
1.	1. INTRODUCTION	6
	1.1 Energy Scenario in India	6
	1.2 Scope of Solar Power in India	7
2.	2. THERMAL ENERGY STORAGE – THE NEED AND DIFFERENT FORMS	10
	2.1 Energy Storage Methods	10
	2.1.1 Mechanical Energy Storage	14
	2.1.2 Electrochemical Energy Storage	16
	2.1.3 Thermal Energy Storage	20
	2.1.3.1 Sensible Heat Storage	
	2.1.3.2 Latent Heat Storage	
	2.1.3.3 Thermochemical Heat Storage	
3.	LATENT HEAT STORAGE MATERIALS	21
	3.1 Properties of Phase Change Materials (PCM)	
	3.1.1 Thermal Properties	
	3.1.2 Physical Properties	
	3.1.3 Kinetic Properties	
	3.1.4 Chemical Properties	
	3.2 Brief Classification of Phase Change Materials	
4.	HEAT TRANSFER IN LATENT HEAT STORAGE MATERIALS	21
	4.1 The Stefan Problem	
	4.2 Governing Equations for Phase Change problems	
	4.3 Enthalpy-Porosity Formulation	
5.	INTRODUCTION TO FLUENT COMMERCIAL CODE	22
	5.1 User Defined Functions in FLUENT	
6.	SOLUTION PROCESS	22
7.	VALIDATION OF CODE WITH BENCHMARK SOLUTION	
8.	THERMOPHYSICAL PROPERTIES OF SELECT SALTS	
9.	CONCLUSION	
	REFERENCES	
	APPENDIX A - ABBREVIATIONS	I
	APPENDIX B – CODE	I
	APPENDIX C – LIST OF COMMON ERRORS WHILE COMPILING UDF	V

1. INTRODUCTION:

1.1 Energy Scenario in India

Solar Power encompasses all the techniques which are used to harness the energy of the Sun and convert to electricity. Solar Energy is a clean and green source of energy. It is one of the most promising sources of energy especially in tropical countries where the insolation is abundant. It can be harnessed by concentrating solar energy to a certain point with the help of parabolic troughs or mirrors known as heliostats. This in turn increases the temperature of the working fluid in the Rankine Cycle. In most cases, the working fluid is water which passes through a turbine which is coupled to a motor to generate electricity. The use of Photovoltaic cells is also common. It is based on the Photoelectric effect. Solar cells produce D.C power which alternates with lights intensity. Inverters are used and solar cells are coupled to form arrays to give stable A. C output.



Figure 1 : Concentrating solar power with heliostats

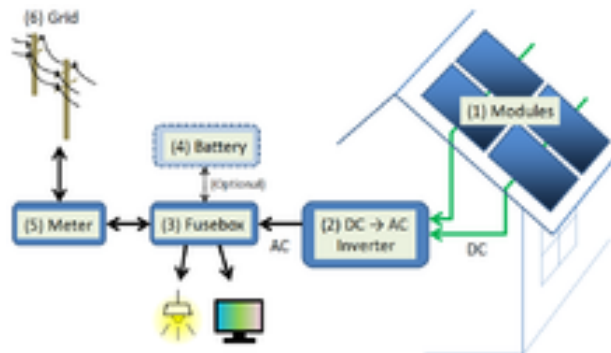


Figure 2 : Schematic of a photovoltaic system

India is a densely populated country and it is blessed with a high density of insolation. It has nearly 300 clear days of sunshine in a calendar year. Solar Energy is one of the most promising sources of renewable energy in India.

1.2 Scope of Solar Power in India

According to a new report by GTM Research and Bridge to India, the nation is facing a perfect storm of factors that will drive solar photovoltaic (PV) adoption at a furious pace over the next five years and beyond. The falling prices of PV panels, mostly from China but also from the U.S., have coincided with the growing cost of grid power. Government support and ample solar resources have also helped push the adoption along, but perhaps the biggest factor has been need. India, as a growing economy with a surging middle class, is now facing a severe electricity deficit that often runs between 10 and 13 percent of daily need. India is aiming to expand its power generation capacity by 44% however according to government data it fell short by 5.8% when faced with a peak load of 128GW. The suburbs of metropolitan cities experience power cut or load shedding of two hours daily. Indian faced a power deficit of 12MW according to Central Electricity Agency (CEA).

Keeping this in mind, various programs have been announced to boost development in the renewable energy sector. In 2009, Jawaharlal Nehru National Solar Mission was launched to create an environment for solar energy penetration.

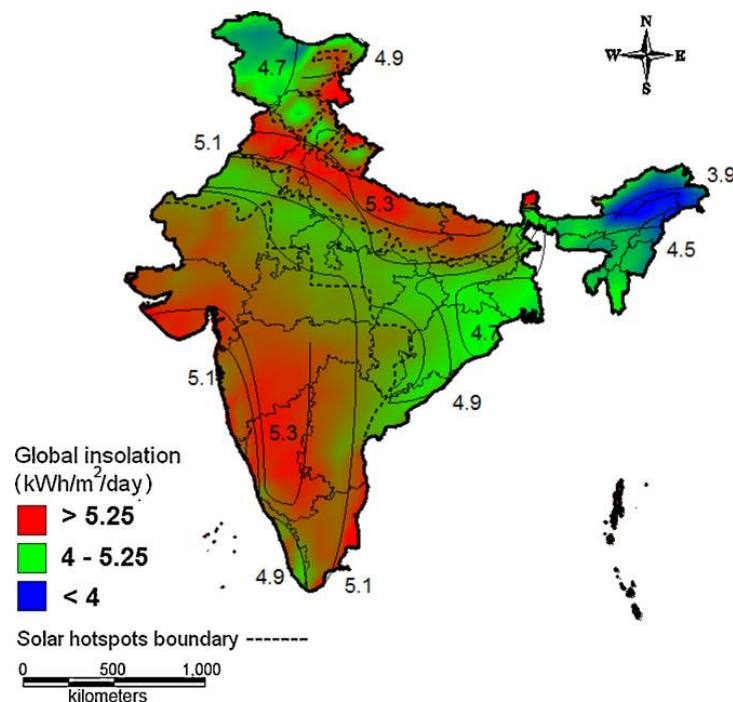


Figure 3: Annual average global insolation map of India showing hotspots [3]

The state of Rajasthan and Gujarat are among the pioneers of Solar Energy development in India. Rajasthan produces 510.25 MW power which accounts for 20% of solar energy production in India. Recent land use statistics reveal that nearly 15% of the land is barren or uncultivable in Rajasthan which can be used to harness solar power. [3]The city of Jodhpur which is the second largest city in Rajasthan is also known as the Sun City as it receives nearly 14 hours of daylight during the summer. The district of Jodhpur alone produces 210 MW of power. Mahindra and Mahindra and Reliance Power, have setup a

5MW and 40MW capacity plant. AREVA, the French based energy company announced 250 MW solar plant in Rajasthan which will save 557,000 tons of coal annually [4]. This will be Asia's largest concentrated solar power plant.

2. THERMAL ENERGY STORAGE – THE NEED AND DIFFERENT FORMS

In many parts of the world specially countries like India, Solar Energy is a promising aspect. Data analysis reveals that 58% of geographical area represents hotspots with more than $5\text{kWh/m}^2/\text{day}$ of annual average global insolation [3]. Although India is one of the best recipients of solar energy due to its favorable location in the solar belt (40°S to 40°N), a meager aggregate of 66MWp (Megawatt peak) solar applications (80% of which are solar lanterns, home/street lighting systems and solar water pumps) are installed in the country.

During monsoons or periods of low insolation, the energy supplied by the solar plant is unable to match the peak load and hence fossil fuel based power sources have to be used as alternatives. To avoid this, researchers are looking at different ways to store energy. Energy storage not only reduces the mismatch between demand and supply but also improves the performance of power generation plant and leads to less wastage of energy. Different forms of energy storage methods are shown in the form of a flow chart below.

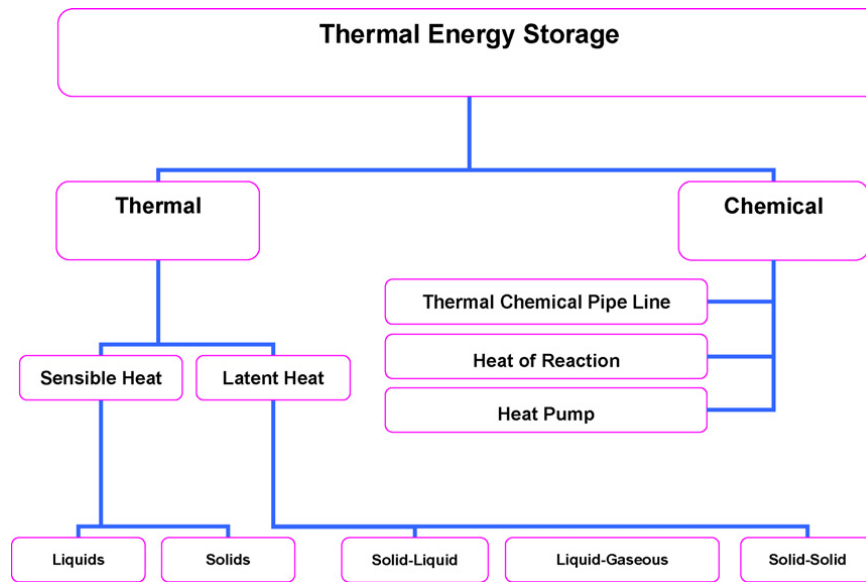


Figure 4: Forms of energy storage

2.1 Energy Storage Methods:

2.1.1 Mechanical energy storage:

Mechanical energy storage systems include gravitational energy storage or pumped hydropower storage (PHPS), compressed air energy storage (CAES) and flywheels. The PHPS and CAES technologies can be used for large-scale utility energy storage while flywheels are more suitable for intermediate storage. Storage is carried out when inexpensive off-peak power is available, e.g., at

night or weekends. The storage is discharged when power is needed because of insufficient supply from the base-load plant.

2.1.2 Electrical Storage:

Energy storage through batteries is an option for storing the electrical energy. A battery is charged, by connecting it to a source of direct electric current and when it is discharged, the stored chemical energy is converted into electrical energy. Potential applications of batteries are utilization of off-peak power, load leveling, and storage of electrical energy generated by wind turbine or photovoltaic plants. The most common type of storage batteries is the lead acid and Ni–Cd.

2.1.3 Thermal Energy Storage:

Thermal energy storage can be stored as a change in internal energy of a material as sensible heat, latent heat and thermochemical or combination of these. An overview of major technique of storage of solar thermal energy is shown in Fig. 4[5].

2.1.3.1 Sensible Energy Storage:

In sensible heat storage (SHS), thermal energy is stored by raising the temperature of a solid or liquid. SHS system utilizes the heat capacity and the change in temperature of the material during the process of charging and discharging. The amount of heat stored depends on the specific heat of the medium, the temperature change and the amount of storage material.

$$Q = \int_{T_i}^{T_f} m C_p dT \quad (1)$$

Water appears to be the best SHS liquid available because it is inexpensive and has a high specific heat. However above 100 °C, oils, molten salts and liquid metals, etc. are used. For air heating applications rock bed type storage materials are used.

2.1.3.2 Latent Heat Storage:

Latent heat storage (LHS) is based on the heat absorption or release when a storage material undergoes a phase change from solid to liquid or liquid to gas or vice versa. The storage capacity of the LHS system with a PCM medium is given by:

$$Q = \int_{T_i}^{T_m} m C_{psolid} dT + m f_l \Delta H + \int_{T_m}^{T_f} m C_{pliquid} dT \quad (2)$$

$$Q = m (C_{psolid} (T_m - T_i) + f_l \Delta H + C_{pliquid} (T_f - T_m)) \quad (3)$$

2.1.3.3 Thermochemical Energy Storage

Thermochemical systems rely on the energy absorbed and released in breaking and reforming molecular bonds in a completely reversible chemical reaction. In this case, the heat stored depends on the amount of storage material, the endothermic heat of reaction, and the extent of conversion.

3. LATENT HEAT STORAGE MATERIALS:

Amongst above thermal heat storage techniques, latent heat thermal energy storage is particularly attractive due to its ability to provide high-energy storage density and its characteristics to store heat at constant temperature corresponding to the phase transition temperature of phase change material (PCM). Phase change can be in the following form solid–solid, solid–liquid, solid–gas, liquid–gas and vice versa. In solid–solid transitions, heat is stored as the material is transformed from one crystalline to another. These transitions generally have small latent heat and small volume changes than solid–liquid transitions. Solid–solid PCMs offer the advantages of less stringent container requirements and greater design flexibility. Most promising materials are organic solid solution of pentaerythritol (m.p. 188 °C, latent heat of fusion 323 kJ/kg), pentaglycerine (m.p. 81 °C, latent heat of fusion 216 kJ/kg), Li₂SO₄ (m.p. 578, latent heat of fusion 214 kJ/kg) and KHF₂ (m.p. 196 °C, latent heat of fusion 135 kJ/kg). Trombe wall with these materials could provide better performance than a plain concrete Trombe wall.

Solid–gas and liquid–gas transition through have higher latent heat of phase transition but their large volume changes on phase transition are associated with the containment problems and rule out their potential utility in thermal-storage systems. Large changes in volume make the system complex and impractical [7]. Solid–liquid transformations have comparatively smaller latent heat than liquid–gas. However, these transformations involve only a small change (of order of 10% or less) in volume. Solid–liquid transitions have proved to be economically attractive for use in thermal energy storage systems. PCMs themselves cannot be used as heat transfer medium. A separate heat transfer medium must be employed with heat exchanger in between to transfer energy from the source to the PCM and from PCM to the load.

The heat exchanger to be used has to be designed specially, in view of the low thermal diffusivity of PCMs in general. The volume changes of the PCMs on melting would also necessitate special volume design of the containers to house PCM. It should be able to absorb these volume changes and should also be compatible with the PCM used. Any latent heat energy storage system therefore, possess at least following three components:

- (i) A suitable PCM with its melting point in the desired temperature range,
- (ii) A suitable heat exchange surface, and
- (iii) A suitable container compatible with the PCM.

The development of a latent heat thermal energy storage system hence, involves the understanding of three essential subjects: phase change materials, containers materials and heat exchangers. A wide range of technical options available for storing low temperature thermal energy is shown in Fig. 5.

Phase change materials (PCM) are “Latent” heat storage materials. The thermal energy transfer occurs when a material changes from solid to liquid, or liquid to solid. This is called

a change in state, or “Phase.” Initially, these solid–liquid PCMs perform like conventional storage materials; their temperature rises as they absorb heat. Unlike conventional (sensible storage materials, PCM absorbs and release heat at a nearly constant temperature. They store 5–14 times more heat per unit volume than sensible storage materials such as water, masonry, or rock. A large number of PCMs are known to melt with a heat of fusion in any required range. However, for their employment as latent heat storage materials these materials must exhibit certain desirable thermodynamic, kinetic and chemical properties. Moreover economic considerations and easy availability of these materials has to be kept in mind. The PCM to be used in the design of thermal-storage systems should possess desirable thermophysical, kinetics and chemical properties which are as follows [5, 6]:

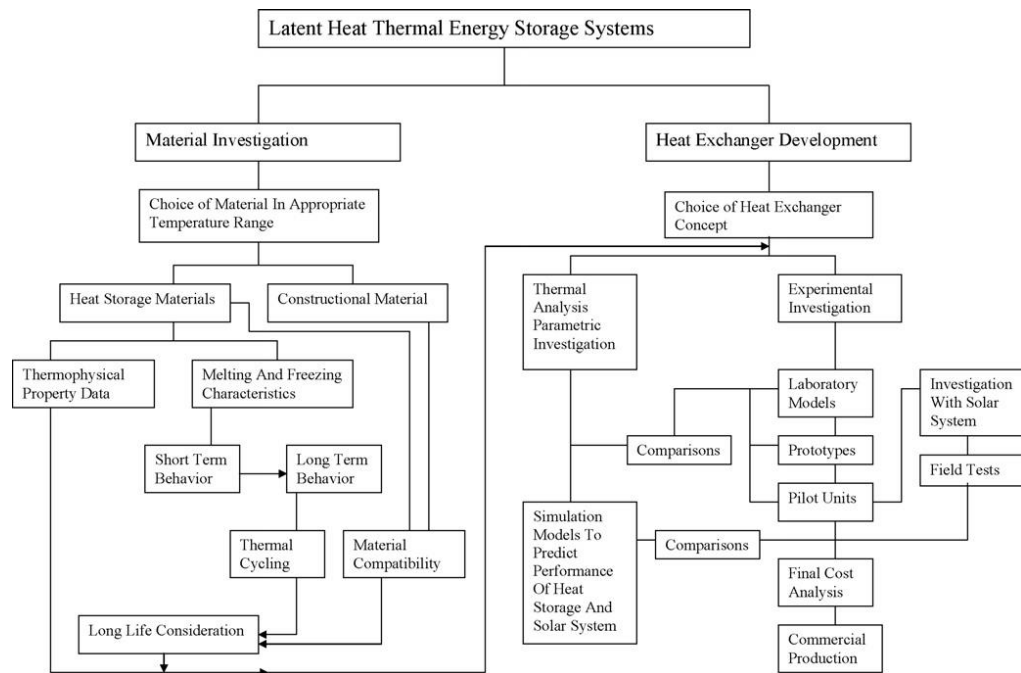


Figure 5 : Different stages in design of Latent Heat Thermal Energy Storage Systems

3.1 Properties of Phase Change Materials:

3.1.1 Thermal Properties:

- (i) Suitable phase-transition temperature.
- (ii) High latent heat of transition.
- (iii) Good heat transfer.

Selecting a PCM for a particular application, the operating temperature of the heating or cooling should be matched to the transition temperature of the PCM. The latent heat should be as high as possible, especially on a volumetric basis, to minimize the physical size of the heat store. High thermal conductivity would assist the charging and discharging of the energy storage.

3.1.2 Physical Properties

- (i) Favorable phase equilibrium.
- (ii) High density.
- (iii) Small volume change.
- (iv) Low vapor pressure.

Phase stability during freezing melting would help towards setting heat storage and high density is desirable to allow a smaller size of storage container. Small volume changes on phase transformation and small vapor pressure at operating temperatures to reduce the containment problem.

3.1.3 Kinetic Properties:

- (i) No supercooling
- (ii) Sufficient crystallization rate

Supercooling has been a troublesome aspect of PCM development, particularly for salt hydrates. Supercooling of more than a few degrees will interfere with proper heat extraction from the store and 5-10°C of supercooling can totally prevent it.

3.1.4 Chemical Properties:

- (i) Long-term chemical stability.
- (ii) Compatibility with materials of construction.
- (iii) No toxicity.
- (iv) No fire hazard.

PCM can suffer from degradation by loss of water of hydration, chemical decomposition or incompatibility with materials of construction. PCMs should be non-toxic, nonflammable and non-explosive for safety.

3.1.5 Economics

- (i) Cost effective
- (ii) Available

3.2 Classification of Phase Change Materials:

A large number of phase change materials (organic, inorganic and eutectic) are available in any required temperature range. A classification of PCMs is given in Table 1. There are a large number of organic and inorganic chemical materials, which can be identified as PCM from the point of view melting temperature and latent heat of fusion.

However, except for the melting point in the operating range, majority of phase change materials does not satisfy the criteria required for an adequate storage media as discussed earlier. As no single material can have all the required properties for an ideal thermal-storage media, one has to use the available materials and try to make up for the poor physical property by an adequate system design. For example metallic fins can be used to increase the thermal conductivity of PCMs. In general inorganic compounds have almost double volumetric latent heat storage capacity (250–400 kg/dm³) than the organic compounds (128–200 kg/dm³).

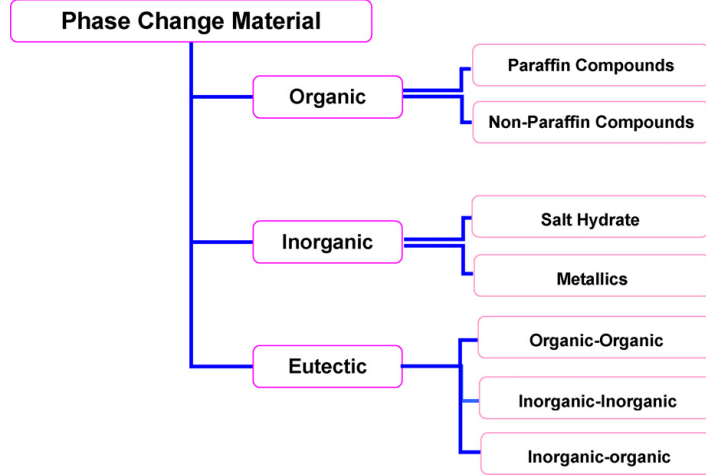


Table 1 : Brief Classification of PCM

4. HEAT TRANSFER IN LATENT HEAT THERMAL ENERGY STORAGE SYSTEMS:

The study of heat transfer in Latent Heat Storage systems involves the simulation of melting and solidification using computational techniques. This study can be used to determine the charging and discharging time and thus estimate the overall performance of the system. A brief review on the analytical and numerical schemes available is given in the following section.

4.1 The Stefan Problem:

In 1889 Stefan in his work on the freezing of the ground posed and solved the following problem: The heat conducting material occupies the space $-\infty < x < \infty$. At the initial time the liquid phase fills the domain $0 < x < \infty$ at temperature $T_l > 0^\circ\text{C}$ while the solid occupies the domain $-\infty < x < 0$ at temperature $T_s < 0^\circ\text{C}$. At $x = 0$, it is maintained at constant temperature $T < \infty^\circ\text{C}$, under the effect of which the solidification occurs isothermally at temperature $T_m = 0^\circ\text{C}$ without the solid-liquid interface given by Stefan is:

$$\lambda \rho \frac{ds}{dt} = k_s \frac{\delta T_s}{\delta t} - k_l \frac{\delta T_l}{\delta t} \quad (4)$$

This is known as the moving boundary problem and results from energy conservation across the interface.

The problem of predicting the behavior of phase change systems is difficult due to its non-linear nature at moving interface and in addition, the two phases have different thermophysical properties. Classically, the Stefan problem was first approached as a pure conduction in semi-infinite medium and later on natural convection has been considered during melting and solidification of phase change materials. The different classes of solution available for Stefan problem are analytical and numerical. Many approximate analytical techniques such as the heat

balance integral variation technique, isothermal migration, source and sink method and periodic solution .A common drawback of these approximate techniques is limited to one dimensional analysis and they become very complicated when applied to multidimensional problem. Numerical methods, both finite difference and finite element appear more powerful in solving the moving boundary problem. In general, a time variant mesh approach offers good accuracy but is limited to simple problems and geometry's. The fixed mesh approach, in which the latent heat of fusion is usually absorbed into the material's specific heat is much simpler. One of the most common methods is to solve by enthalpy formulation. The enthalpy method is used in a particular way so that the only unknown variable is the temperature of the phase change material and the solidification occurs at a uniform temperature. Enthalpy method treats the enthalpy as a temperature dependent variable and constructs the latent heat flow through the volume integration with the use of the enthalpy of the system. Heat transfer with moving boundary involving phase change is very important in latent heat storage application, i.e., ice formation, freezing of food, castings, metallurgy, crystal growth and various other solidification techniques. The predication of temperature distribution and rate of melting or solidification is very important in order to design such storage device.

4.2 Governing Equations for Phase Change Problems:

The differential equations governing the phase change are of the general form as mentioned in Patankar[7].

$$\frac{\partial \rho \phi}{\partial t} + \nabla(\rho \mathbf{u} \phi) = \nabla(\tau \nabla \phi) + S_{\phi} \quad (5)$$

Continuity:

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \mathbf{u}) = 0 \quad (6)$$

X-Momentum:

$$\frac{\partial \rho u}{\partial t} + \nabla(\rho \mathbf{u} u) = \nabla(\mu \nabla u) - \frac{\partial P}{\partial x} + A u \quad (7)$$

Y-Momentum:

$$\frac{\partial \rho v}{\partial t} + \nabla(\rho \mathbf{u} v) = \nabla(\mu \nabla v) - \frac{\partial P}{\partial y} + S_b$$

$$S_b = \frac{\rho_{ref} g \beta (h - h_{ref})}{c}$$

Laminar flow is assumed and Boussinesq treatment is assumed to be valid i.e. density is constant in all terms except a gravity source term.

The term 'A' represents the porosity term and is given according to the Karman-Kozeny Equation as:[8]

$$A = -C \frac{(1-\epsilon)^2}{\epsilon^3 + b} \quad (8)$$

Energy Equation:

$$\frac{\partial \rho h}{\partial t} + \nabla(\rho \mathbf{u} h) = \nabla(k \nabla h) - \frac{\partial(\rho \Delta H)}{\partial t} - \nabla(\rho \mathbf{u} \Delta H) \quad (9)$$

Thermal conductivity, density are treated as constants, so Equation (9) becomes

$$\frac{\rho \partial h}{\partial t} + \rho \nabla(\mathbf{u} h) = k \nabla(\nabla h) - \frac{\rho \partial(\Delta H)}{\partial t} - \rho \nabla(\mathbf{u} \Delta H)$$

$$\begin{aligned} \Delta H &= f(T) = L & T > T_m \\ &= 0 & T < T_m \end{aligned}$$

4.3 Enthalpy Porosity Formulation:

By introducing an enthalpy method, the phase change problem becomes much simpler and has the following advantages: (i) the governing equation is similar to the single phase equation; (ii) there is no condition to be satisfied at the solid–liquid interface as it automatically obeys the interface condition and (iii) the enthalpy formulation allows a mushy zone between the two phases. Phase change problems are usually solved with finite difference or finite element methods in accordance with the numerical approach. The enthalpy formulation is one of the most popular fixed-domain methods for solving the Stefan problem. The major advantage is that the method does not require explicit treatment of the moving boundary. To introduce the formulation, we define an enthalpy function h as a function of temperature and equations are applied over the whole fixed domain as given by Voller [8].

The enthalpy methods account for the Latent heat in energy equation by assigning a nodal latent heat value to each computational cell in the domain according to the temperature of the cell. Upon changing phase, the nodal latent heat value is adjusted to account for latent heat absorption or evolution, the adjustment being reflected in the energy equation as a source or sink. Computational cells that are undergoing phase change are modeled as pseudo-porous media.

Also

$$f_l + f_s = 1 \quad \mathbf{u} = u_l f_l + u_s f_s \quad c_p = c_{pl} f_l + c_{ps} f_s$$

$$k = k_l f_l + k_s f_s \quad \Delta H = f_l L$$

Formulating a generalized energy equation taking into account the variations of thermal conductivity and specific heat in the solid and liquid state, following the footsteps of Chakraborty and Dutta [9], we get

$$\frac{\partial \rho c_p T}{\partial t} + \nabla(\rho c_{pl} \mathbf{u} T) = \nabla(k \nabla T) - \frac{\partial(\rho \Delta H)}{\partial t} - \nabla(\rho u_l \Delta H) + \nabla(\rho f_s u_s (C_{pl} - C_{ps}) T) \quad (10)$$

This equation takes into account the motion of solid particles in the mushy zone. However, to simulate the melting of molten salts (which will be used as Heat storage materials), $u_s = 0$. Thus only first two terms are there on the right side of the energy equation. The energy equation is solved in this form to account for a more general case.

To obtain the numerical solution, the equations are discretized according to Patankar[7] in the form:

$$a_p T_p = \sum a_{nb} T_{nb} + a_p^0 T_p^0 + d \quad (11)$$

The discretized domain is shown in Figure 6.

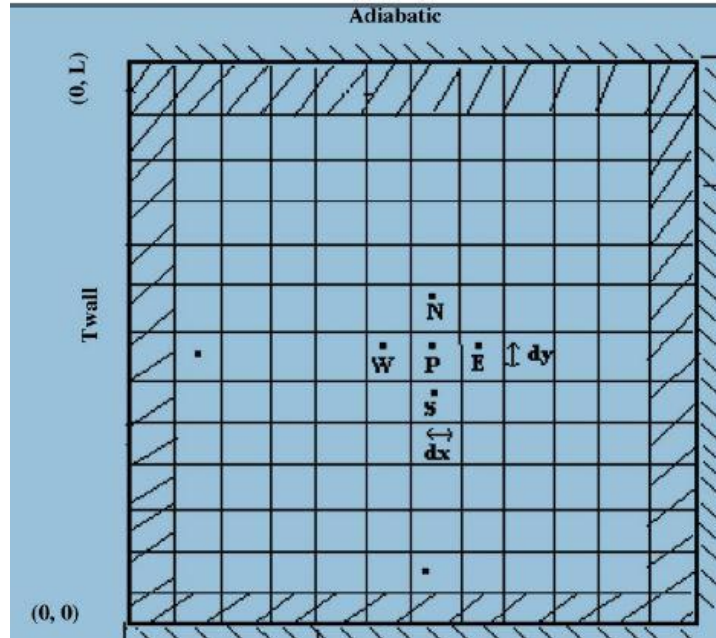


Figure 6 : Boundary Conditions

The nodal latent heat is updated according to Voller et al as:

$$[H_p]_{n+1} = [H_p]_n + \frac{a_p}{a_p^0} c \lambda ([T_p]_n - T_m) \quad (12)$$

The values of the discretization coefficients are computed during iterations. Tri Diagonal Matrix Algorithm (TDMA) or Line by Line TDMA is the preferred choice of most solvers. A detailed derivation of the generalized enthalpy update scheme is given below:

At n^{th} iteration step the discretization equation is written as:

$$a_p (T_p)_n = \sum a_{nb} (T_{nb})_n + a_p^0 T_p^0 - A_p (\Delta H_p)_n + A_p^0 \Delta H_p^0 + \sum A_{nb} (\Delta H_{nb})_n$$

At (n+1)th iteration this can be written as:

$$a_p (T_p)_{n+1} = \sum a_{nb} (T_{nb})_{n+1} + a_p^0 T_p^0 - A_p (\Delta H_p)_{n+1} + A_p^0 \Delta H_p^0 + \sum A_{nb} (\Delta H_{nb})_{n+1}$$

Subtracting two equations,

$$\begin{aligned} (\Delta H_p)_{n+1} &= (\Delta H_p)_n + \frac{a_p}{A_p} ((T_p)_n - (T_p)_{n+1}) \\ -\frac{1}{A_p} \sum (a_{nb} (T_{nb})_n - a_{nb} (T_{nb})_{n+1}) - \frac{1}{A_p} (\sum A_{nb} (\Delta H_{nb})_n - A_{nb} (\Delta H_{nb})_{n+1}) \end{aligned} \quad (13)$$

In the preceding iteration scheme attainment of convergence implies $(\Delta H_p)_{n+1} = (\Delta H_p)_n$. And all other terms on the right hand side tend to zero. For such an iterative scheme it is common practice to use nodal value with some relaxation factor. With this modification, the equation takes the following form:

$$(\Delta H_p)_{n+1} = (\Delta H_p)_n + \frac{a_p \lambda}{A_p} ((T_p)_n - (T_p)_{n+1})$$

5. INTRODUCTION TO FLUENT COMMERCIAL CODE:

ANSYS INC offers a wide range of inbuilt modules by the name of ANSYS to solve problems related structural, mechanical and electromagnetism. FLUENT INC. is now completely owned by ANSYS and it is a commercial solver written in C language and integrated into ANSYS. FLUENT is one of the most robust solvers available to solve flow and heat transfer problems with an ability to write user defined functions to suit problem conditions. Different solution schemes are available for solving flow and heat transfer problems.

5.1 User Defined Functions in FLUENT:

To simulate melting and solidification in terms of generalized energy equation (Equation 11), a scalar transport equation was written in FLUENT. A brief description about user defined functions is given.

The need of a User Defined Function (UDF) arises because :

1. The codes solved in FLUENT cannot anticipate all needs.
2. New physical models can be developed in a user friendly environment.
3. Large number of problems can be addressed with same implementation.
4. While solving equations FLUENT takes into account certain assumptions. To include source terms or generalize the equation or modify the equation as needed a UDF needs to be written.

Figure 6 shows the entities in FLUENT. A collection of entities is called a zone or thread. Domain comprises of all the threads or different zones. An advantage of writing a UDF is that their syntax is derived from C language. As an example, to give variable inlet velocity, the steps are as follows:

1. Identify the inlet faces.
2. Loop over them
3. Find the centroid of the face
4. Apply a velocity profile

The inlet faces are recognized from their ID which is displayed in Define → Boundary Conditions tab. The sample code is shown in Appendix A.

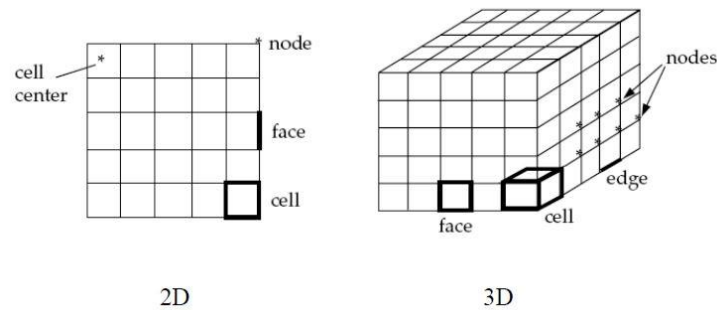


Figure 7 : Entities in FLUENT

A UDF must be linked to the main code. This can be done in two ways:

1. Interpreted UDF:
 - The code is executed on a line by line basis.
 - It does not need a separate compiler.
 - It slows down execution.
 - The scope is limited as Structure references cannot be implemented.
2. Compiled UDF:
 - A library of UDF is compiled and linked to the main code.
 - Overcomes all the shortcomings listed above.
 - For Windows Operating Systems (64 bit), it requires installation of Visual C++ 2008/2010 Express Edition along with Windows Software Development Kernel (SDK). A detailed procedure can be found in Appendix A.

The general solution procedure and the sequence of calling the UDF for Pressure based segregated solver is shown below.

The User defined Scalar Transport equation is solved in the general form as:

$$\frac{\partial \rho \phi}{\partial t} + \nabla(\rho \mathbf{u} \phi - (\tau \nabla \phi)) = S_\phi \quad (14)$$

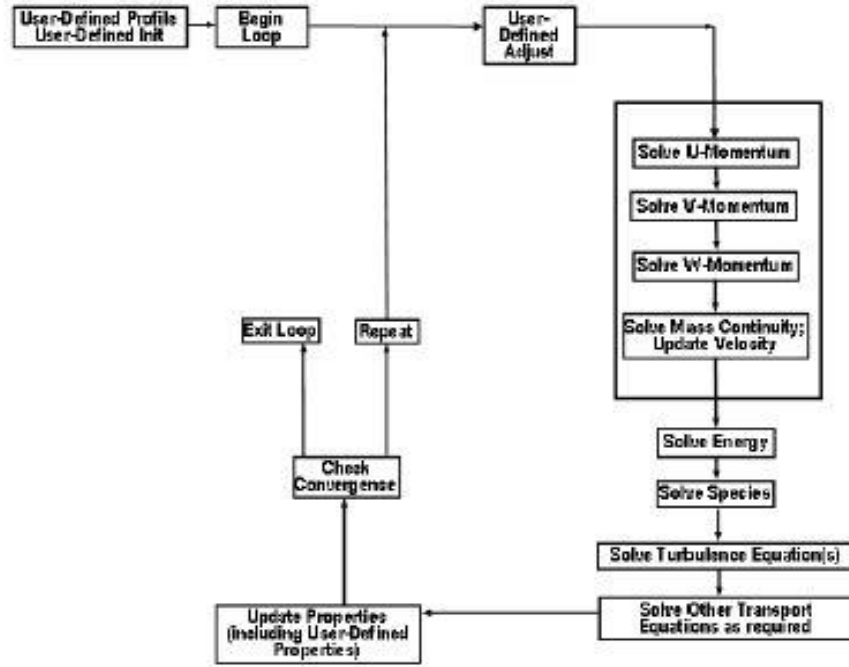


Figure 8: Solution Procedure for Pressure Based solver

6. SOLUTION PROCESS:

To generalize the energy equation to account for variation in specific heat and thermal conductivity, the energy equation is solved in terms of temperature. FLUENT solves energy equation in the form of enthalpy. Thus, a User Defined Scalar Transport (UDS) equation with temperature as the variable was written. The Energy Module in FLUENT was NOT switched on.

The benchmark paper [8] studied the melting of Gallium. The properties were taken as mentioned in the paper by Voller [8]. The geometry was a square enclosure. It was modeled in ANSYS Workbench. A grid independence test was carried out and a 42x32 grid was chosen. Same values of under-relaxation factors were used and convergence criteria were to 1e-5 for all residuals (continuity, momentum and energy equations). SIMPLE algorithm as suggested by Patankar[7] was used for pressure-velocity coupling. The detailed procedure of compiling the UDF and the solution setup is explained below.

6.1 Preprocessing – Meshing and Importing Geometry in FLUENT

Preprocessing involves:

- Modeling the geometry
- Meshing
- Setting up the solution

The geometry is shown in Figure 7. The named selections are used to denote the walls of the enclosure.

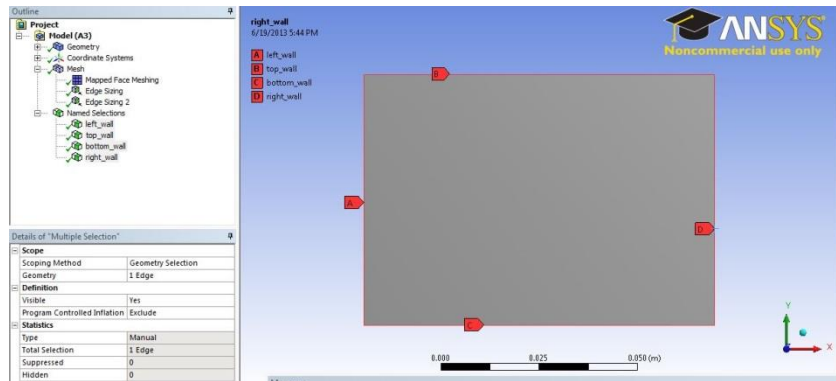


Figure 9 : Geometry

Mapped Face Meshing is applied to get a quadrilateral mesh shown in Figure 8. Mapped Face Meshing maps the face to a rectangular domain.

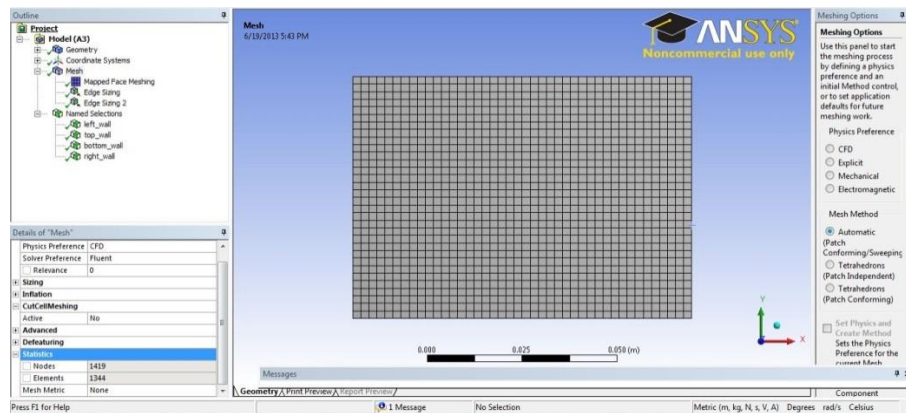


Figure 10 : Meshed Geometry

After meshing successfully, the project is saved and FLUENT is opened. The mesh will be automatically imported if FLUID FLOW was selected as the standalone project type from the left window in ANSYS workbench as shown in Figure 9.

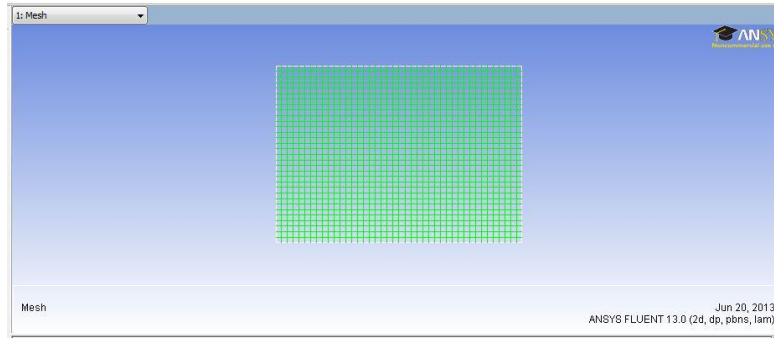


Figure 11: Imported mesh in FLUENT

In the Main window beside, click on **CHECK**. This displays the cell size and volume. Make sure that the cell volume is not negative. **REPORT QUALITY** reports the Minimum Orthogonal Quality of the mesh.

From Main window, set solution type to **Transient**.

6.2 Preprocessing –Solution Setup

This section deals with setting the physics of the problem and compiling and hooking user defined functions.

6.2.1 Compiling UDF

From the main window, go to **Define → User Defined → Memory**

Input the number of memory allocations needed. As a caution, do not input more than that needed as an error will be displayed in the later stages.

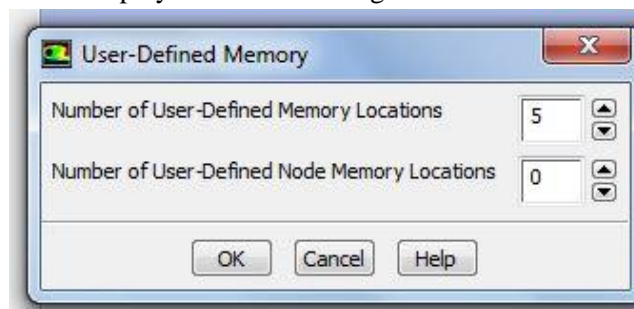


Figure 12: User Defined Memory Allocation

To incorporate a scalar transport equation, go to **Define → User Defined → Scalars**

From the dialog box that appears, select the number of scalars i.e. the transport equations that need to be solved. Also the unsteady term and flux can be given.

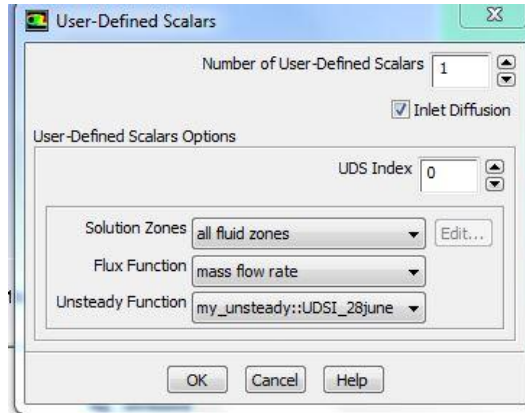


Figure 13 : User Defined Scalar

To add a scalar variable – temperature in this case go to **Define → User Defined → Scalars**. Only one scalar equation is solved.

To compile the UDF, go to **Define → User Defined → Functions → Compiled**. A dialog box opens as shown in Figure 11.

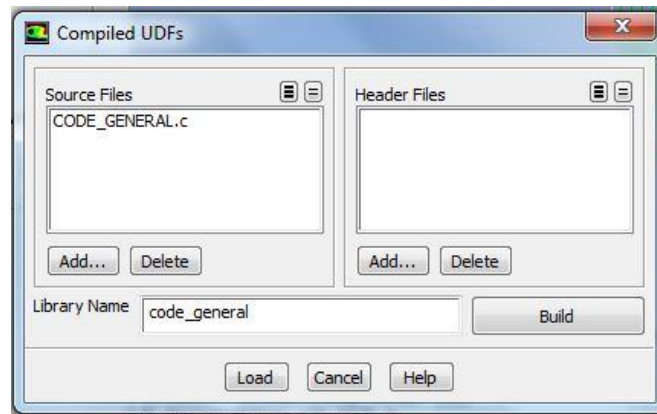


Figure 14: Compiled UDF window

From the **Add** button, add the source file. **Make sure that your source file is in the same directory as your case and data files**. By default the case and data files are saved in:

Place where you chose to save project → Name of project_files → dp0 → FFF-1 (if it's a standalone project or FLU-1 otherwise) as shown in Figure 12.

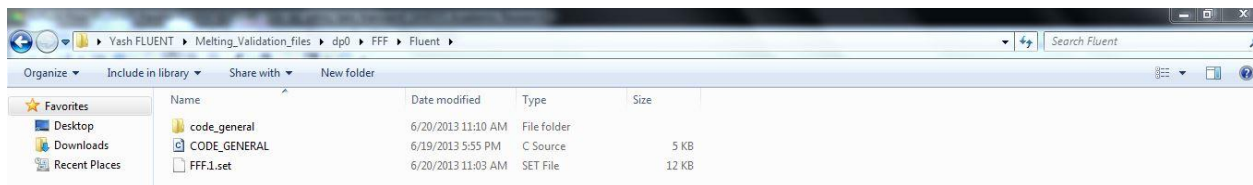


Figure 15 : Directory of Stored UDF

Click **Build**.

After successful compilation, click **Load** button. The text command interface of FLUENT will display all the User defined functions as shown in Figure 13.

```
Opening library "C:\Users\IITJ\Desktop\Yash FLUENT\Melting_Validation_files\dp0\FFF\Fluent\code_general"...
Library "C:\Users\IITJ\Desktop\Yash FLUENT\Melting_Validation_files\dp0\FFF\Fluent\code_general\ntx86\2ddp_host\...
Opening library "C:\Users\IITJ\Desktop\Yash FLUENT\Melting_Validation_files\dp0\FFF\Fluent\code_general"...
Library "C:\Users\IITJ\Desktop\Yash FLUENT\Melting_Validation_files\dp0\FFF\Fluent\code_general\ntx86\2ddp_node\...
Opening library "C:\Users\IITJ\Desktop\Yash FLUENT\Melting_Validation_files\dp0\FFF\Fluent\code_general"...
Library "C:\Users\IITJ\Desktop\Yash FLUENT\Melting_Validation_files\dp0\FFF\Fluent\code_general\ntx86\2ddp_node\...
Opening library "C:\Users\IITJ\Desktop\Yash FLUENT\Melting_Validation_files\dp0\FFF\Fluent\code_general"...
Library "C:\Users\IITJ\Desktop\Yash FLUENT\Melting_Validation_files\dp0\FFF\Fluent\code_general\ntx86\2ddp_node\...
initialize
x_source
y_source
eng_source
my_adjust
Done.
```

Figure 16: Loading UDF

From Figure 7, it can be seen that **DEFINE_INITIALIZE MACRO** is called during initialization and **DEFINE_ADJUST** macro is called each iteration. To Hook these macros, go to

Define → User Defined → Function Hooks.

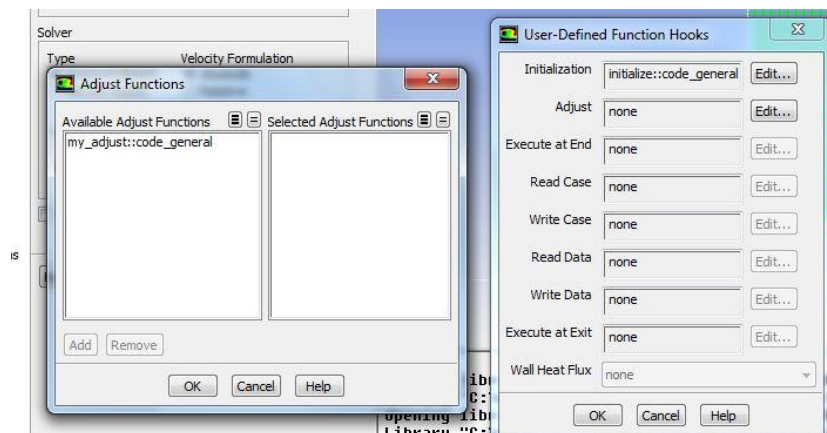


Figure 17 : Hooking UDF functions

From Figure 15, attach the Adjust macro by clicking on **Edit** from User defined Function Hooks window and from the subsequent **Adjust Functions** window, **add** the adjust macro.

To define source terms, go to **Define → Cell Zone Conditions**. Check the **Source Terms**. Click the **Source Terms Tab**.

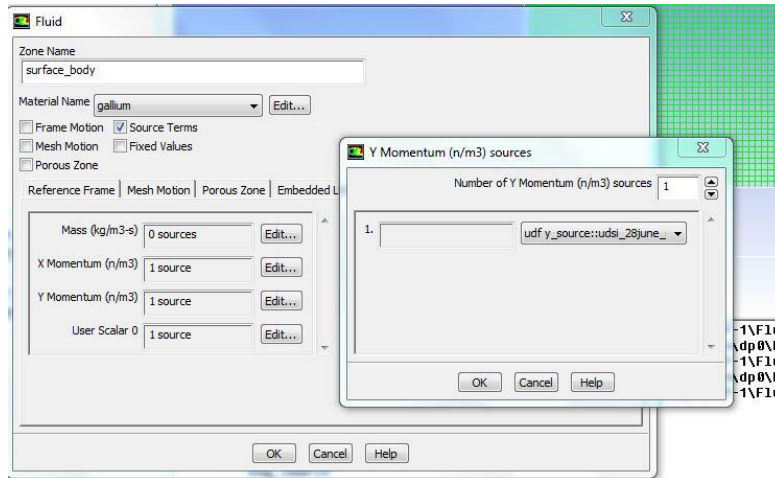


Figure 18: Define Source Terms

To add the X-Momentum source term, click on **Edit** and insert one source; from the dropdown list that appears appropriately select the corresponding DEFINE_SOURCE UDF. The same procedure holds good for all source terms.

For the transport equation, the diffusivity term is needed. This can be defined from **Define → Materials → Select appropriate material → Edit → Diffusivity → User Defined Coefficient**

The appropriate temperature boundary conditions are added from **Define → Boundary Conditions**. It is to be noted that since the Energy Equation is off, **Thermal Tab can't be selected. Temperature is a User defined Scalar in our case so the boundary condition is specified from the UDS Tab**. The side walls are differentially heated whereas the top and the bottom walls are adiabatic.

The **Reference Values** are set so that these property values will be taken as the reference.

The **Solution Methods** are set as: SIMPLE pressure velocity coupling, **Power Law** schemes for Momentum and UDS-0 (Temperature), **Green Gauss Cell Based** discretization.

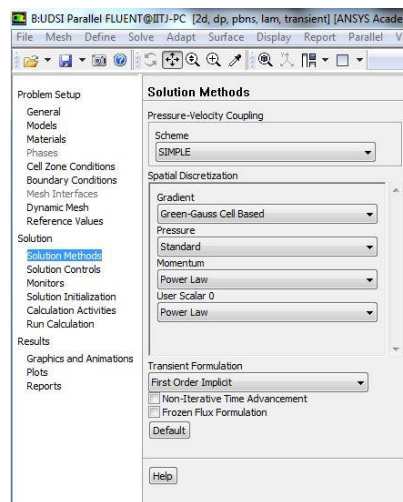


Figure 19 : Solution Methods

By default, FLUENT does not store the values of the user defined memories which will be required in different macros. To do so, **type in solve→set→expert** and **Keep temporary memory from being freed → YES** as shown in Figure 17.

```

writing left_wall (type wall) (mixture) ... Done.
writing top_wall (type wall) (mixture) ... Done.
writing bottom_wall (type wall) (mixture) ... Done.
writing right_wall (type wall) (mixture) ... Done.
writing zones_map name-id ... Done
solve
/solve> set
/solve/set> expert
use alternate formulation for wall temperatures? [no] no
Save cell residuals for post-processing? [no] no
Keep temporary solver memory from being freed? [no] yes
Allow selection of all applicable discretization schemes? [no] no

```

Figure 20 : SOLVE SET EXPERT COMMAND

Initialize solution by giving a velocity of 1e-3 m/s and define the temperature as 290 K in **UDS Scalar-0 field**. An initial velocity is given to create a disturbance which helps in initiating the solution.

Initially a time step size of 0.001s is used. After some stability in residuals, it is gradually increased. However, the time step was never increased beyond 1s.

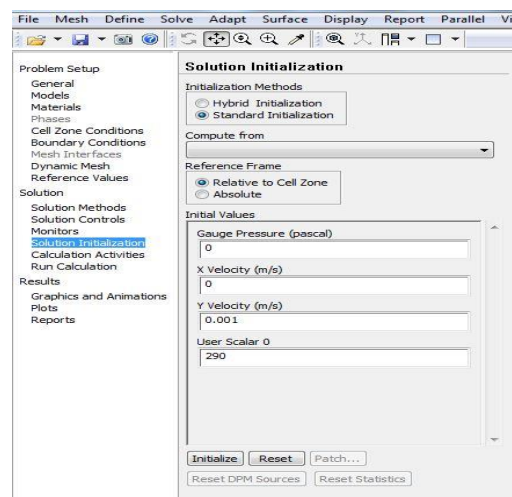


Figure 21 : Solution Initialization

7. VALIDATION OF CODE WITH BENCHMARK SOLUTION:

The problem geometry is as shown. The phase change material is in the circular storage tank through which a heat transfer fluid supplies or extracts heat leading to melting and solidification. Due to the symmetry of the problem, a cut section is used for subsequent analysis. A transient solution is done.

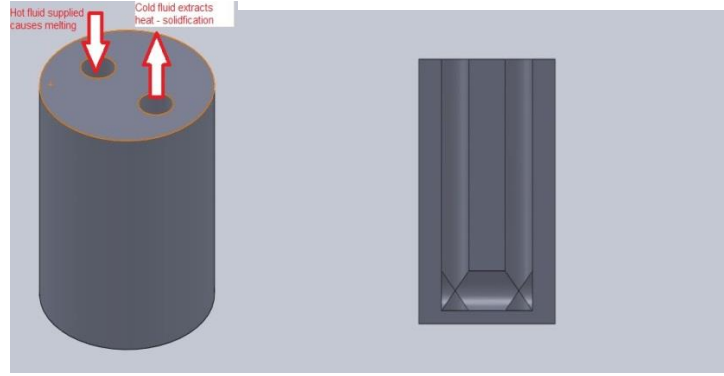


Figure 22 : Isometric and Section view of the Thermal Energy storage system

The User defined function implemented in FLUENT is checked using the results obtained by Voller et. al [8]. The reference material is Gallium. The standard properties of Gallium are shown in Table 2.

Property	Value	Units
Density	6093	kg/m ³
Melting Point	29.78	°C
Thermal Conductivity	32	W/m-K
Viscosity	0.00181	kg/m-s
Specific Heat	381.5	J/kg-K
Latent Heat of fusion	80160	J/kg

Table 2: Properties of Gallium

The size of the enclosure and the boundary conditions are taken as per Voller et. al[8]. A User defined function (a code in C language) is implemented which accounts for updating the enthalpy and liquid fraction. User Defined Macro's are implemented. [11]

The Energy equation is switched on and the results obtained are compared with the reference results.

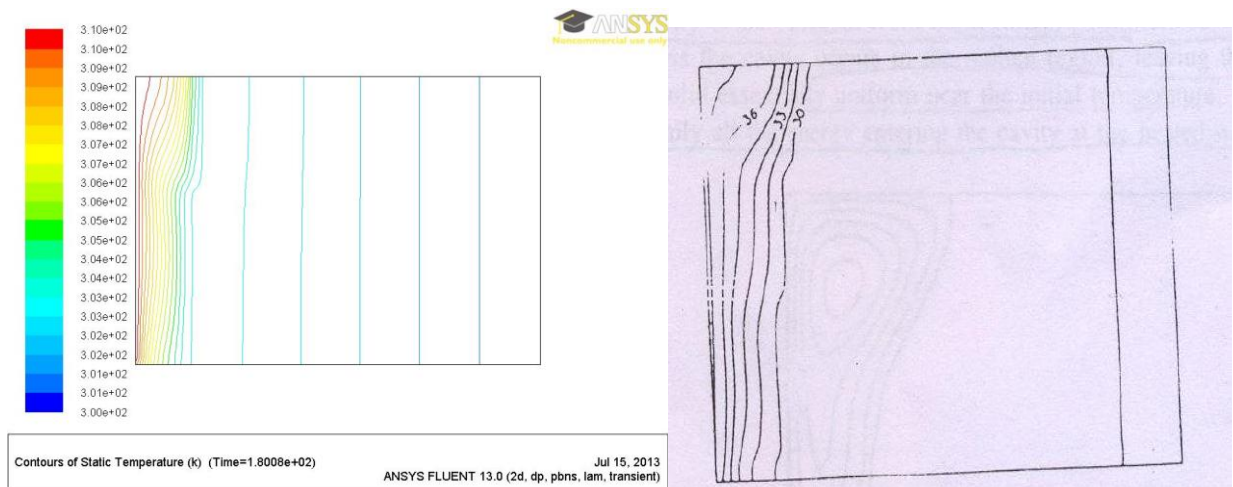


Figure 23 : Comparison of temperature contours from the code implemented (left) and the standard results (right)

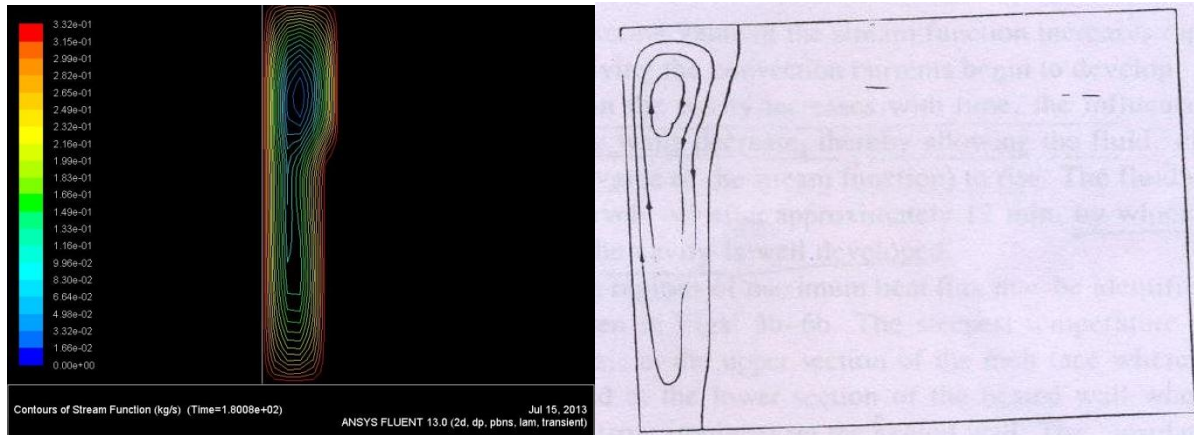


Figure 24 : Comparison of stream function

In the second step, the default energy equation in FLUENT is not solved. Rather, an energy equation in terms of temperature is solved by incorporating temperature as a User Defined Scalar variable. The transport equation is solved and the appropriate unsteady, flux and source terms are added. In all cases, the density, thermal conductivity and viscosity are treated as constants. There is no motion of solid particles in the melt thus $\mathbf{u}_s = \mathbf{0}$.

The results obtained are compared with the reference results and with the previous case where energy equation was solved as per FLUENT.

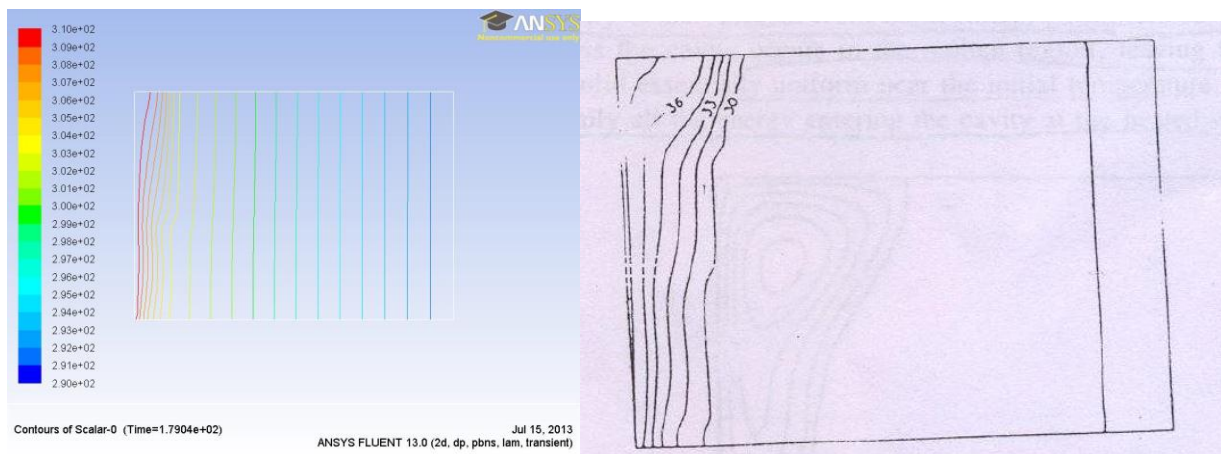


Figure 25 : Comparison of result from UDS Transport equation with reference solution

A comparison is made between the two results obtained. It can be seen that they agree closely.

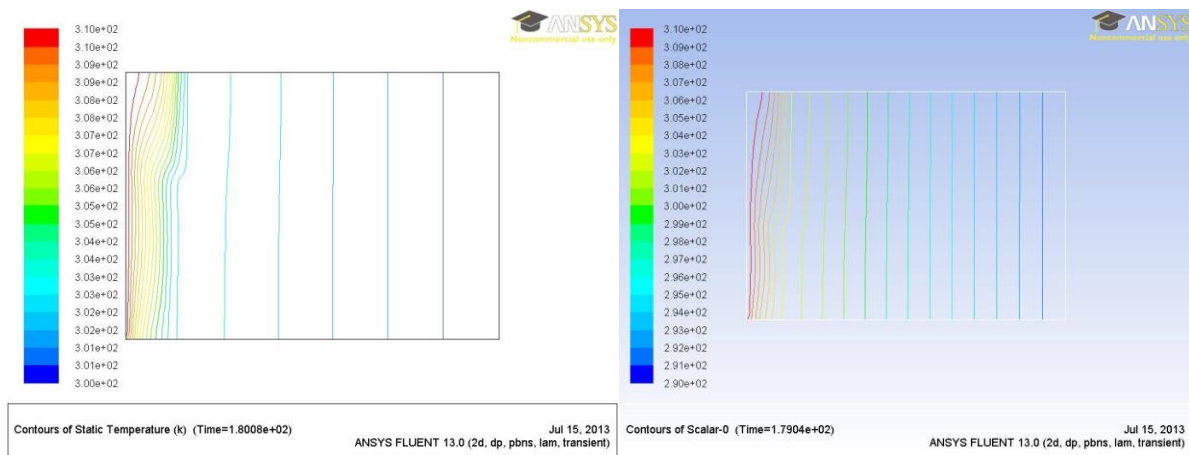


Figure 26 : Comparison of temperature plots obtained from FLUENT (left) and UDS Transport Equation (right)

8. THERMOPHYSICAL PROPERTIES OF SALTS:

In Concentrated Solar Power Plants (CSP), salts are used as phase change materials. The most common salt is NaNO_3 and KNO_3 mixture. It is also known as Solar Salt, it is in the ratio of 50% NaNO_3 by mole and 50% KNO_3 by mol. [12].

The properties of such salts vary with temperature. The data for these properties has been compiled from National Standards Reference Data produced by Ganz et. al [13] and by DLR, Germany [14]. These properties have been summarized in the form of table.

Sodium Nitrate:

Melting Point: 579 K

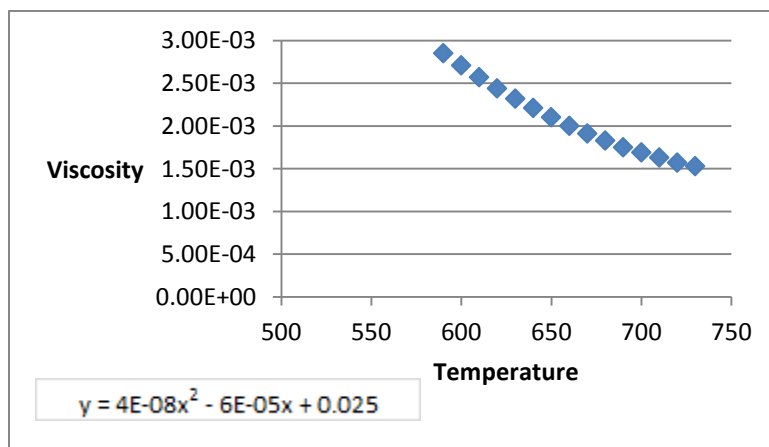


Figure 27: Variation of Viscosity with temperature in Sodium Nitrate

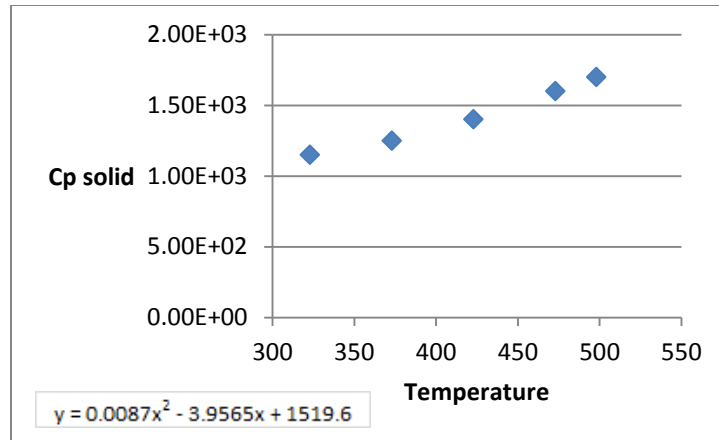


Figure 28: Variation of Specific heat in solid state with temperature in Sodium Nitrate

For solid thermal conductivity, the fit obtained from [13] is in the form of $a+bT$ where $a = 0.396057$
 $b = 0.000336$.

Data	Sources	Type	Value	Unit
Density	DLR	Constant	1908	kg/m ³
Cpliquid	DLR	Constant	1.66E+03	J/Kg-K
Cpsolid	NIST	Variable		J/Kg-K
Viscosity	NIST	Variable		kg/m-s
Latent Heat	NIST	Constant	182127	J/Kg
Liquid Cond	DLR	Constant	0.514	W/m-K
Solid Cond	NIST	Variable		W/m-K

Table 3 : Thermophysical Properties of Sodium Nitrate

Potassium Nitrate:

Melting Point: 610 K

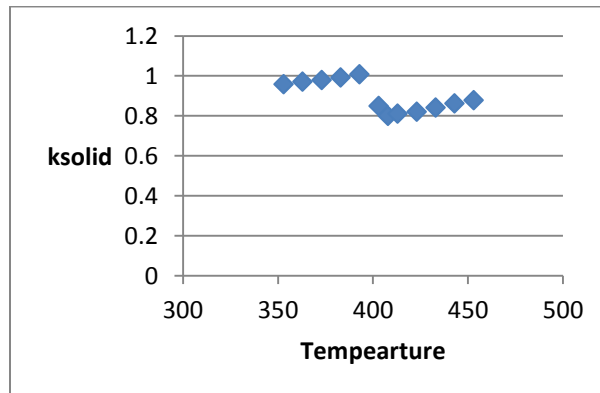


Figure 29: Variation of Thermal conductivity in solid state with temperature in Potassium Nitrate

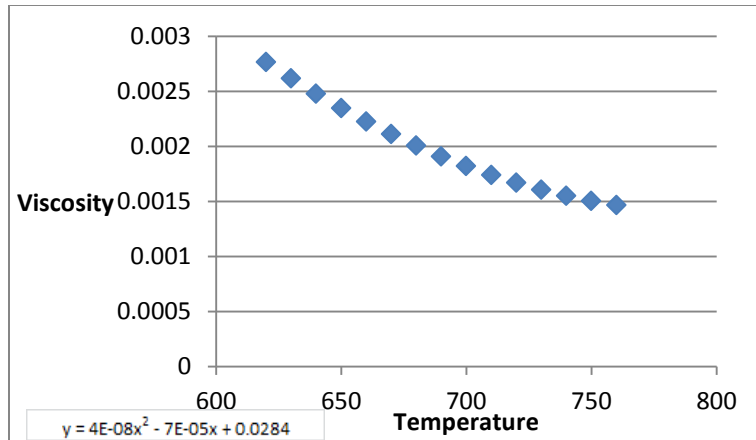


Figure 30: Variation of Viscosity with temperature in Potassium Nitrate

Data	Sources	Type	Value	Unit
Density	NIST	Constant	1863	kg/m ³
Cpliquid	NIST	Constant	1.34E+03	J/Kg-K
Cpsolid	NIST	Constant	1.34E+03	J/Kg-K
Viscosity	NIST	Variable		kg/m-s
Latent Heat	NIST	Constant	99901	J/Kg
Liquid Cond	NIST	Constant	0.459	W/m-K
Solid Cond	NIST	Variable		W/m-K

Table 4: Thermophysical Properties of Potassium Nitrate

Sodium and Potassium Nitrate (Eutectic):

Melting Temperature : 495 K [15]

Data	Sources	Type	Value	Unit
Density	NIST	Constant	1850	kg/m ³
Cpliquid	DLR	Constant	1.49E+03	J/Kg-K
Cpsolid	DLR	Constant	1.34E+03	J/Kg-K
Viscosity	NIST	Variable		kg/m-s
Latent Heat	NIST	Constant	108000	J/Kg
Liquid Cond	DLR	Constant	0.457	W/m-K
Solid Cond	DLR	Constant	0.455	W/m-K

Table 5: Thermophysical Properties of Sodium Nitrate and Potassium Nitrate Eutectic Mixture

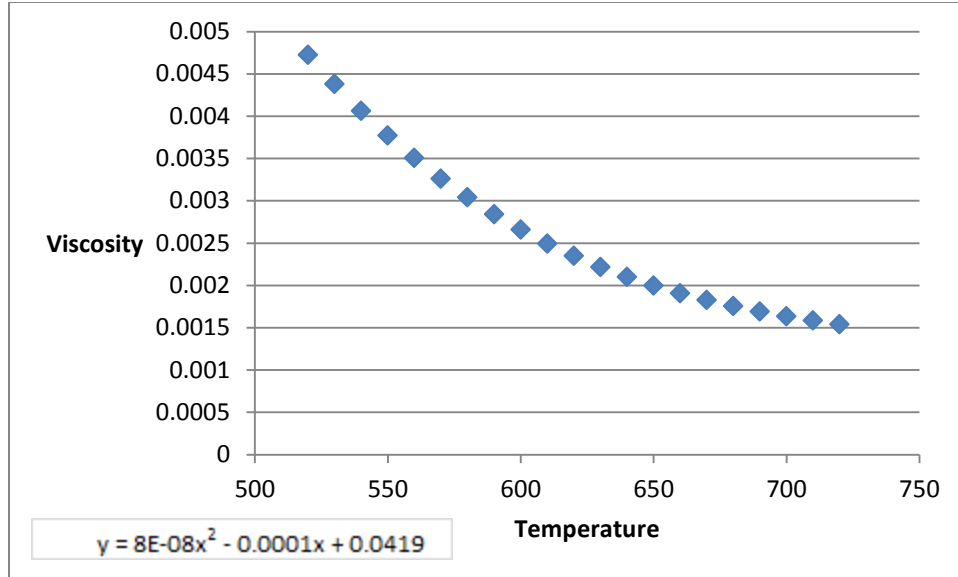


Figure 31 : Variation of Viscosity with temperature in Sodium and Potassium Nitrate eutectic mixture

9. CONCLUSION:

To study the viability of using Sodium and Potassium based salts as heat storage medium, the charging (time taken to melt) and discharging times (time taken to solidify) are of utmost importance. A code was implemented in FLUENT using user defined function to find study melting and solidification. The charging and discharging times can be found out from them. The Thermophysical properties of Solar salt were compiled using reference data from [13, 14]. Further, a code was implemented to account for variation in viscosity and thermal conductivity with temperature. In the future, the effect of corrosion on storage material at high temperatures can be investigated. Also, the variation in specific heat can be accounted for. The results obtained from simulation can be used to design the storage system. Finally, an experiment should be done to validate the results obtained from simulation.

REFERENCES:

- [1] - Review on thermal energy storage with phase change materials and applications, Atul Sharma et al. , Renewable and Sustainable Energy Reviews ,Elsevier Publishing
- [2] - Design of Molten-Salt Thermocline Tanks for Solar Thermal Energy Storage, S. V. Garimella, Z. Yang, S. M. Flueckiger, Taylor and Francis group
- [3] - Hotspots of solar potential in India, T.V. Ramachandra et. al , Renewable and Sustainable Energy Reviews, Elsevier Publishing
- [4] - <http://idealab.talkingpointsmemo.com/2012/04/areva-details-new-india-project-biggest-solar-steam-plant-in-asia.php>
- [5] - Abhat A. Low temperature latent heat thermal energy storage: heat storage materials. Solar Energy 1981;30(4):313–32
- [6] - Buddhi D, Sawhney RL.In: Proceedings on thermal energy storage and energy conversion; 1994.
- [7] – Numerical Heat Transfer and Fluid flow, S.V. Patankar (1980), Hemisphere Publishing corporation
- [8] – The Enthalpy Porosity Technique for Modeling Convection-Diffusion Phase change: Application to melting of pure metal, Voller et. al
- [9] - Modeling of solid phase sedimentation during directional solidification in a side cooled cavity Prodyut Chakraborty and Pradip Dutta, International Journal of Numerical Methods for Heat & Fluid Flow
- [10] - A Generalized Enthalpy Update Scheme for Solidification of a Binary Alloy with Solid Phase Movement, Prodyut Chakraborty and Pradip Dutta, The Minerals, Metals & Materials Society and ASM International 2011
- [11]- FLUENT UDF MANUAL 12.1, ANSYS Inc.
- [12] - Thermodynamic Properties Of Molten Nitrate Salts, Joseph G. Cordaro1 et. al, Sandia National Laboratories, Livermore, CA
- [13] - Physical Properties Data compilation relevant to Energy storage II: Molten Salts : Data on single and multi-component system, G.J. Ganz, C.P. Allen, N.P. Bansal
- [14] - Sodium Nitrate For High Temperature Latent Heat Storage, T. Bauer, Institute for Technical Thermodynamics, DLR,Germany
- [15] - Overview of PCMs for concentrated solar power in the temperature range 200 to 350 °C, T.Bauer et.al, Advances in Science and Technology, Trans Tech Publications

APPENDIX A

ABBREVIATIONS:

Symbol	Meaning
T_i	Initial temperature
T_f	Final Temperature
T_m	Melting Temperature
C_p	Mixture specific heat
C_{psolid}	Specific heat of solid phase
$C_{pliquid}$	Specific heat of liquid phase
m	Mass
f_l	Liquid fraction
λ	Under-relaxation factor
ρ	Density
k_s	Solid phase thermal conductivity
k_l	Liquid phase thermal conductivity
k	Thermal conductivity of mixture
L	Latent Heat
H	Heat of fusion
g	Acceleration due to gravity
β	Thermal Expansion coefficient
μ	Viscosity
P	Pressure
A	Constant
C	Morphological constant
UDF	User Defined Function
UDS	User Defined Scalar

APPENDIX B:

CODE

```
/* Validation problem from Incropera
  C_UDMI(c,t,0) -- Latent heat content (delta H)
  C_UDMI(c,t,1) -- Latent heat content for last timestep
  C_UDMI(c,t,2) -- Liquid fraction
  C_UDMI(c,t,3) -- Liquid fraction for last timestep
  C_UDMI(c,t,4) -- 0 indicates solid zone
                  1 indicates liquid zone
  C_UDSI(c,t,t1) -- Temperature
*/
/* USE OF UDSI TO WRITE TRANSPORT EQN IN TERMS OF TEMPERATURE*/

#include "udf.h"
#include "sg_mem.h"
#include "mem.h"
#include "sg.h"
```

```

/* ===== USER INPUTS START ===== */
#define Cmor      1.0e+07  /* Morphological Constant */
#define Tiny      0.001    /* Small number to avoid div by zero */
#define lamda      0.01    /* Relaxation factor for latent heat update */
#define TMELT      579     /* Melting point in Kelvin */
#define Liquid_Cond      0.514 /* Liquid phase conductivity in W/m-K */

#define Latent_Heat      182127.0 /* Latent heat in J/kg */
#define beta_thermal      40e-6 /* Thermal expansion coefficient in 1/K */
#define solid_specific_heat      1.6e3 /* Specific heat in J/kg.K */
#define liquid_specific_heat      1.4e3 /* Specific heat in J/kg.K */

enum
{
    t1, N_REQUIRED_UDS
};

/* ===== USER INPUTS END ===== */

/*===== INITIALIZING =====*/
/* Identify the cells in the mushy zone (initial state is either solid or liquid – write for that) */

DEFINE_INIT(initialize,d)
{ cell_t c;
  Thread *t;
  int i;

  thread_loop_c(t,d) /* Loop over all cell threads in domain */
  {
    if (FLUID_THREAD_P(t))
    {
      begin_c_loop(c,t) /* Loop over all cells in a cell thread*/
      {
        for(i=0; i<sg_udm; ++i) C_UDMI(c,t,i) = 0.0;
      }
      end_c_loop(c,t)
    }
  }
}

/* ===== VISCOSITY TERM ===== */
DEFINE_PROPERTY(cell_viscosity,c,t)
{
  real mu_lam,temp;
  temp = C_UDSI(c,t,t1);
  mu_lam = (4.00E-08*temp*temp) - (6.00E-05*temp) + 0.025;
  return mu_lam;
}

/* ===== UNSTEADY STATE TERM ===== */
DEFINE_UDS_UNSTEADY(my_unsteady,c,t,i,apu,su)
{
  real timestep,vol,rho,temp_old;
  timestep = RP_Get_Real("physical-time-step");
  vol = C_VOLUME(c,t);

```

```

    rho = C_R_M1(c,t);
    *apu = -rho*vol/timestep; /*implicit part*/
    temp_old = C_STORAGE_R(c,t,SV_UDSI_M1(t1));
    *su = rho*vol*temp_old/timestep; /* explicit part*/
}
/*===== DEFINING SOURCE TERMS=====*/
/* NOTE MINUS SIGNS*/
/* X-Momentum Source Term */
DEFINE_SOURCE(x_source, c, t, dS, eqn)
{
    real con, source, lfrac;
    lfrac = C_UDMI(c,t,2);
    con = -Cmor*(1.0-lfrac)*(1.0-lfrac)/((lfrac*lfrac*lfrac) + Tiny);
    source = con * C_U(c,t);
    dS[eqn] = con;
    return source;
}
/* Y-Momentum Source Term (instead of Teut write Tmelt) */
DEFINE_SOURCE(y_source, c, t, dS, eqn)
{
    real con, source, lfrac, thermal;
    lfrac = C_UDMI(c,t,2);
    con = -Cmor*(1.0-lfrac)*(1.0-lfrac)/((lfrac*lfrac*lfrac) + Tiny);
    source = con * C_V(c,t);
    /* Boussinesque terms */
    thermal = C_R(c,t)*9.81*beta_thermal*(C_UDSI(c,t,t1)-TMELT);
    source += thermal;
    dS[eqn] = con;
    return source;
}
/* Energy Source Term (second one is ignored as no velocity in solid zone)*/

DEFINE_SOURCE(eng_source, c, t, dS, eqn)
{
    real source, timestep, lfrac,specific_heat;
    timestep = RP_Get_Real("physical-time-step");
    lfrac = C_UDMI(c,t,2);
    specific_heat = (lfrac*liquid_specific_heat)+((1-lfrac)*solid_specific_heat);
    source= -C_R(c,t)*(C_UDMI(c,t,0)-C_UDMI(c,t,1))/(specific_heat*timestep);
    dS[eqn]=0.0;
    return source;
}
/* Diffusivity*/
DEFINE_DIFFUSIVITY(diff_coeff,c,t,i)
{
    real cond,specific_heat,lfrac,diff,temp, Solid_Cond;
    lfrac = C_UDMI(c,t,2);
    temp = C_UDSI(c,t,t1);
    Solid_Cond = 0.39605744 + (0.00033557*temp);
    cond = (lfrac*Liquid_Cond)+ ((1-lfrac)*Solid_Cond);
    specific_heat = (lfrac*liquid_specific_heat)+((1-lfrac)*solid_specific_heat);
    diff = cond/specific_heat;
    if(i==t1) return diff;
    else return 3.21e-4;
}
/* Defining Adjust function. Here enthalpy will be updated*/
DEFINE_ADJUST(my_adjust, d)
{
    real timestep, ap, ap0,lfrac,specific_heat;
    int n;
    cell_t c;

```

```

Thread *t;
face_t f;
timestep = RP_Get_Real("physical-time-step");
if(N_UDS < N_REQUIRED_UDS)
    Error("Not enough user defined scalar allocated");
thread_loop_c(t,d) /* Loop over all cell threads in domain */
{
if (FLUID_THREAD_P(t))
{
begin_c_loop(c,t)
{
/* Store the latent heat content for previous timestep
This will be required for energy equation source term */
if(first_iteration)
{
C_UDMI(c,t,1) = C_UDMI(c,t,0);
C_UDMI(c,t,3) = C_UDMI(c,t,2);
}
/* Identify the cells in the mushy zone (my code – note brackets) */

if(C_UDSI(c,t,t1) < TMELT) C_UDMI(c,t,4) = 0.0; /* Solid region */
else C_UDMI(c,t,4) = 1.0; /* Liquid region */
/* Update Latent heat content (Define inverse function first then proceed. Inverse function is TMELT)*/
ap0 = C_R(c,t)*C_VOLUME(c,t)/timestep;
lfrac = C_UDMI(c,t,2);
specific_heat = (lfrac*liquid_specific_heat)+((1-lfrac)*solid_specific_heat);
if (NNLLP(THREAD_STORAGE(t,SV_UDSI_AP))) ap = -C_STORAGE_R(c,t,SV_UDSI_AP);
else ap = 0.0;
C_UDMI(c,t,0) += (ap/ap0)*lamda*specific_heat*(C_UDSI(c,t,t1)- TMELT);

/* Eliminate unphysical values */

if(C_UDMI(c,t,0) > Latent_Heat ) C_UDMI(c,t,0) = Latent_Heat;
if(C_UDMI(c,t,0) < 0.0) C_UDMI(c,t,0) = 0;

/* Calculate liquid fraction */
C_UDMI(c,t,2) = (C_UDMI(c,t,0)) / Latent_Heat;

}
end_c_loop(c,t)
}
}
}

```

APPENDIX C

LIST OF COMMON ERRORS WHILE COMPILING UDF

These are the errors which I encountered while compiling UDF's in FLUENT. I have also given the possible cause and solution.

1. nmake error
'nmake' is not recognized as an internal or external command, operable program or batch file.
Cause: This is the first error while compiling the UDF on a system where Microsoft Visual C++ , Windows Software Development Kernel (SDK) and Microsoft .NET Framework has not been installed.
For 64 bit system – install these three, change the **Path** Environment Variable in the Advanced System Settings as shown in http://www.cfd-online.com/Wiki/Fluent_FAQ.
2. Divergence Detected in AMG solver
This maybe caused when boundary conditions are given incorrectly. If boundary conditions are specified correctly, decrease the under-relaxation factors.
3. Undeclared identifier
error C2065: 't' : undeclared identifier, illegal use of this type as an expression
This is caused when the variable is not declared in the UDF. This is mostly due to negligence. Also in some Macro's like DEFINE_SOURCE, the variables that are passed by FLUENT to the UDF include cell and thread from within a loop thus there is no need to loop over cell and thread again by declaring them explicitly.

www.cfd-online.com contains a wealth of information regarding FLUENT. It has a special section dealing with UDF Programming. Since the site is 19 years old, chances are high that most of the problems encountered are already posted and solved in the forum.