MODELING AND DESIGN OF REACTOR FOR HYDROGEN PRODUCTION USING NON-STOICHIOMETRIC OXIDE

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ARDA YILMAZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
CHEMICAL ENGINEERING

JANUARY 2017

Approval of thesis:

MODELING AND DESIGN OF REACTOR FOR HYDROGEN PRODUCTION USING NON-STOICHIOMETRIC OXIDE

submitted by **ARDA YILMAZ** in partial fulfillment of the requirements for the degree of **Master of Science in Chemical Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Halil Kalıpçılar _____
Head of Department, **Chemical Engineering**

Assoc.Prof. Dr. Serkan Kıncal _____
Supervisor, **Chemical Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Deniz Üner _____
Chemical Engineering Dept., METU

Assoc. Prof. Dr. Serkan Kıncal _____
Chemical Engineering Dept., METU

Prof. Dr. Derek K. Baker _____
Mechanical Engineering Dept., METU

Asst. Prof. Dr. Harun Koku _____
Chemical Engineering Dept., METU

Assoc. Prof. Dr. Niyazi Alper Tapan _____
Chemical Engineering Dept., Gazi University

Date: January 27, 2017

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: ARDA YILMAZ

Signature :

ABSTRACT

MODELING AND DESIGN OF REACTOR FOR HYDROGEN PRODUCTION USING NON-STOICHIOMETRIC OXIDES

Yılmaz, Arda
M. Sc. Department of Chemical Engineering
Supervisor: Assoc. Prof. Serkan Kıncal

January 2017, 160 pages

Nowadays countries investigate to improve alternative energy technologies such as solar power, biomass, wind energy, hydrogen etc. Hydrogen gas is very useful energy carrier and fuel cells produce electricity through hydrogen gas. Hydrogen production technologies are also investigated by many researches due to its high cost production. Thermochemical production way is one of the hydrogen production methods. Solar energy is also clean, renewable and alternative energy source. It is used for heating reaction systems but modeling of solar system requires optimization in terms of heating need of reaction and operation temperature.

Main purpose of this study is to model and design optimum reactor system in terms of heat, mass and momentum transport phenomena via statistical approach, JMP, COMSOL and MATLAB programs. In this reactor system, hydrogen gas is produced in monolith reactor from steam through solar energy and metal oxide catalyst. In front side of reactor, quartz glass takes place for solar irradiation. Backside of reactor is assumed well insulated because this side is closed and reactor channels connect to gas storage place via valve and vacuum system throughout this side. Reactor channel walls are coated with metal oxide catalyst. There is an insulation layer on the outside

of reactor for decreasing energy loss. Artificial experiment (design of experiment-DOE) runs are set via JMP program to determine significant parameters for thermal and kinetic model. After that, thermal, mass-momentum transport simulation models, which are based on significant parameters, are configured on COMSOL. Hydrogen conversion value is obtained on MATLAB by using rate expressions of real experiment and temperature profiles of COMSOL results. Also, model validation studies are configured on COMSOL.

In mass-momentum transport model, neglecting effects of mass transfer and momentum transfer on temperature profiles is verified due to low temperature differences for both reduction and oxidation reactions. Hydrogen conversion is found as 0.7. Hydrogen concentration toward end of the channel is higher because of high reaction rate. In kinetic model, when heating time is shorter than 3 min cordierite is the best material but when heating time is more than 3 min, silicon carbide is the best material in terms of oxygen conversion due to thermal conductivity. Surface area for solar flux and reactor length are very significant parameters for analysis of channel shape effect on oxygen conversion. In first and second simulations including main and second order effect except channel shape of thermal model and statistical approach, optimum conditions of reactor system are silicon carbide as reactor material, high CPSI (cell per square inch), averaged 300 sun solar flux, thin wall thickness for minimum temperature difference. According to final statistical analysis including all effects, optimum conditions of reactor system are high CPSI, high solar flux, square channel model, cordierite material, low wall thickness and optimum inner insulation thickness. In this optimum reactor model, oxygen production rate is 0.15-0.20 min^{-1}, heating time is 1-2 mins and all temperature differences are 50-200 º C. Model validation is carried out for solar flux and temperature profiles of reactor at steady-state. Solar energy is determined 330 W and temperature profiles overlap each other by tuning some physical parameters.

Keywords: Solar energy, hydrogen gas production via thermochemical method, thermal mass and momentum transport modeling on COMSOL, statistical analysis via JMP, monolith reactor

ÖZ

STOKİYOMETRİK OLMAYAN OKSİTLER KULLANILARAK HİDROJEN ÜRETİM REAKTÖRÜNÜN SÜREÇ MODELLEMESİ VE TASARIMI

Yılmaz, Arda
Yüksek Lisans, Kimya Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Serkan Kıncal

Ocak 2017, 160 sayfa

Bugünlerde ülkeler güneş enerjisi, biyokütle, rüzgar enerjisi ve hidrojen gibi alternative enerji teknolojilerini geliştirmek için araştırmalar yapıyorlar. Hidrojen gazı çok kullanışlı bir enerji taşıyıcısıdır ve yakıt pilleri hidrojen gazı yardımıyla elektrik üretir. Birçok araştırmacı tarafından yüksek maliyetli üretiminden dolayı hidrojen üretim teknolojileri araştırılmaktadır. Termokimyasal üretim yolu önemli hidrojen üretim yöntemlerinden birisidir. Aynı zamanda güneş enerjisi de temiz, yenilenebilir ve alternatif bir enerji kaynağıdır. Güneş enerjisi, reaksiyon sistemini ısıtmada kullanılır fakat güneş enerjisi modeli reaksiyonun ısı ihtiyacı ve operasyon sıcaklığına göre optimizasyona ihtiyaç duyar.

Bu çalışmanın ana amacı ısı,kütle ve momentum olarak optimum reaktör sistemini JMP, COMSOL, MATLAB programları ve istatistiksel yaklaşımla modellemek ve tasarlamaktır. Bu reaktör sisteminde hidrojen gazı su buharından güneş enerjisi ve metal oksit katalizör kullanılarak monolit reaktörde üretilmektedir. Reaktörün ön yüzeyinde quartz cam bulunmaktadır. Arka yüzeyi ise mükemmel izolasyon mekanizmasına sahipmiş gibi değerlendirilmiştir. Reaktörün kanallarının duvarları katalizör malzeme ile kaplıdır ve ısı kaybını azaltmak amacıyla reaktörün çevresini

saran izolasyon malzemeleri kullanılmıştır. Sanal deney setleri, termal ve kinetik model için önemli parametreleri belirlemek amacıyla JMP programında kurgulanmıştır. Termal ve kütle-momentum taşınım modelleri COMSOL programında yapılandırılmıştır. Hidrojen dönüşüm değerleri gerçek deneylerin hız denklemleri ve COMSOL sonuçlarının sıcaklık profilleri kullanılarak MATLAB üzerinde elde edilmiştir. Aynı zamanda model doğrulama çalışması da COMSOL üzerinde yapılandırılmıştır.

Transport modelde kütle ve momentum transferinin sıcaklık profillerine etkisinin ihmal edilmesi iki modelde de düşük sıcaklık farkları olması sayesinde teyit edilmiştir. Hidrojen dönüşüm değeri 0.7 olarak bulunmuştur. Kanalın sonuna doğru hidrojen konsantrasyonu yüksek reaksiyon hızından dolayı daha fazladır. Kinetik modelde ısı iletkenlik katsayıları yüzünden oksijen üretimi bakımından ısınma süresi 3 dakikadan az olursa kordierit en iyi malzemedir fakat ısınma süresi 3 dakikadan fazla olursa silikon karbür en iyi malzemedir. Güneş akısı için yüzey alanı ve reaktör uzunluğu kanal şeklinin oksijen dönüşümü üzerindeki etkisinin analizi için çok önemli parametrelerdir. Termal model ve istatistiksel yaklaşım kısmının ana ve ikincil etkenlerini içeren fakat kanal şekli etkisini içermeyen ilk ve ikinci simulasyonlarında optimum reaktör modeli silikon karbür malzeme, yüksek CPSI (kanal sayısı), ortalama 300 sun civarı güneş akısı, ince et kalınlığı gibi özelliklere sahiptir. Tüm etkenlerin yer aldığı son istatistiksel analize göre optimum reaktör modeli yüksek CPSI, yüksek güneş akısı, kare kanal şekli modeli, kordierit reaktör malzemesi, ince et kalınlığı ve optimum iç izolasyon kalınlığı gibi özelliklere sahiptir. Bu optimum reaktör modelinde oksijen dönüşüm değerleri 0.15-0.20 dakika^{-1}, ısınma süresi 1-2 dakika ve sıcaklık farklılıkları da 50-200 º C olarak elde edilmiştir. Model doğrulama işlemleri güneş akısının ve yatışkın koşullarda çalışan reaktörün sıcaklık profillerinin doğrulanması üzerinden gerçekleştirilmiştir. Güneşten alınan enerji 300 W olarak bulunmuş ve sıcaklık profilleri bazı fiziksel parametreler üzerinden ayarlamalar, değişiklikler yapılarak örtüştürülmüştür.

Anahtar Kelimeler: Güneş enerjisi, hidrojen gazı üretimi, COMSOL programında termal ve taşınım modelleme, JMP ile istatistiksel analiz, monolit reaktör

To my parents

ACKNOWLEDGEMENTS

Firstly, I am grateful to my supervisor Assoc. Prof. Dr. Serkan Kıncal for his guidance, encouragement and helps on all parts of this study. His great vision, understanding, jokes and optimistic talks motivate me to go through this study.

I would like to state my sincere thanks to Enver Mert Uzun, Şahin Anıl Aybek, Sercan Kemal Büyükyılmaz, Mehmet Şentürk, Hüseyin Ufuk Akdağ, Ceren Şengül, M. Nur Aşkın, Can Yıldırım and Efe Seyyal for their great friendship, endless support, patience and understanding, intellectual and emotional talks. Also, I would like to thank my girlfriend, Başak Kütükcü for her eternal love, encouraging me both in my good and hard times, making me happy and always being with me.

I would like to thank to my project mate: Necip Berker Üner, Celal Güvenç Oğulgönen and Atalay Çalışan for their technical support and helps. Especially, Necip helped me many times for configuration of modeling part on COMSOL program. Without his programming experience, vision and big effort I could not do modeling study of my thesis on time.

Finally, I would like to express my deepest gratitude to my parents, Hüseyin Yılmaz and Ersin Yılmaz. Without their unlimited love and continuous support throughout my life, I could not be who I am now.

TUBITAK-ARDEB is acknowledged for the project ‘Stokiyometrik Olmayan Oksitler Kullanılarak Hidrojen Üretim Süreç ve Teknolojilerinin Geliştirilmesi-213M006’.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

Today, energy requirement increases day by day in the world. Also, industrial investments are raised to fulfill basic human needs. These requirements lead to increasing energy consumption. Many countries attach importance to improve alternative energy technologies such as solar power, wind energy, marine energy and hydrogen. Hydrogen gas is center of that topic because it is used in petroleum processes, oil and fat hydrogenation, fertilizer production, metallurgical and electronic applications and finally energy production [1]. Hydrogen production area is open for improvements. Many research groups in many countries investigate about feasible hydrogen production methods. There are some common hydrogen production methods such as steam reforming, coal gasification, biomass gasification or reforming, photo biological, microbial biomass conversion, electrochemical, photochemical and finally thermochemical.

Solar energy is formed from nuclear fusion reaction in the sun. It is a clean, renewable and alternative energy source. Areas of using solar energy in the world are increased day by day. This energy is used for the power demand of home, automobiles, space satellite, providing hot water and some new generation aircraft. Solar energy is also used for heating catalytic reactions which occur at high temperature. Configuration of this system needs well optimized focusing solar energy in terms of reaction heat needs and operation temperature. For instance, parabolic channels are used for 400-600 ˚C temperature band and parabolic concentrators (mirrors) are used for high temperature band.

In this study, the main aim is to model and design reactor in terms of thermal, kinetic and mass-momentum transport analysis by using statistical analysis and JMP, MATLAB and COMSOL programs. For this modeling system, hydrogen gas is produced in monolith reactor from steam by using solar energy and non-

stoichiometric metal oxide. Optimization of reactor is planned. Design of artificial experiment is configured in JMP program to determine important parameters of reactor design in terms of thermal model. Thermal and mass-momentum transport models are configured in COMSOL program to obtain the results of artificial experiments. Hydrogen conversion data are calculated from temperature profiles of COMSOL models in MATLAB program by using rate expressions of real experimental data. Finally, reactor prototype properties are determined in light of these studies.

1.1 Hydrogen Production Methods

1.1.1 Steam Reforming Method

Steam reforming method is the most common hydrogen production method. Generally, methane is used as reactant for this reaction because availability of methane is higher than the other reformer gases due to natural gas. Steam reforming is an endothermic reaction. Methane reacts with steam which has high temperature and pressure [2]. Carbon monoxide and hydrogen gases are produced by this reaction. Carbon monoxide is by product. After reforming reaction, it reacts with steam and carbon dioxide, hydrogen are produced. It is called water gas shift reaction.

Steam reforming reaction: $CH_4 + H_2O \rightarrow CO + 3H_2$ (1.1)

Water gas shift reaction: $CO + H_2O \rightarrow CO_2 + H_2$ (1.2)

1.1.2 Coal Gasification Method

Coal is so complicated material and it can be converted to different type substances. Hydrogen is one of these substances. Coal gasification is the primary reaction in industrial hydrogen production but that process is a little complex because of by-products and reaction mechanism [3]. Coal gasification reaction for hydrogen production is shown below.

Coal gasification reaction: $C_nH_m + nH_2O \rightarrow nCO + (n + m/2)H_2$ (1.3)

1.1.3 Biomass Gasification or Reforming Method

Biomass is a potential renewable, organic energy resource and it can be converted to heat, coal, bio-oil, methanol, ethanol, and hydrogen [4]. It is provided from agriculture crop residues, forest residues and animal wastes. Gasification is occurred at high temperature without combustion. Biomass reacts with controlled oxygen and steam. After that reaction, carbon monoxide, carbon dioxide and hydrogen are produced. Water gas shift reaction occurs similarly like methane reforming. Moreover, biomass can be converted into liquid bio-fuels. Then, hydrogen gas can be produced from liquid bio-fuels via reforming reaction like methane reforming.

Gasification reaction: $C_6H_{12}O_6 + O_2 + H_2O \rightarrow CO + CO_2 + H_2$ (1.4)

Ethanol reforming reaction: $C_2H_5OH + H_2O \rightarrow CO + H_2$ (1.5)

1.1.4 Photo Biological Method

Hydrogen production via photo biological method could be potentially environmentally acceptable energy production method because hydrogen gas is renewable using the primary energy source, sunlight, and does not liberate carbon dioxide during combustion [5]. Algae and cyanobacteria are used as hydrogen producer in photo biological systems. They use sunlight to split water into oxygen and hydrogen ions. Main disadvantages of this method are low hydrogen production rate and interaction possibility among hydrogen and oxygen gases. In Figure 1.1, photo biological method is illustrated.

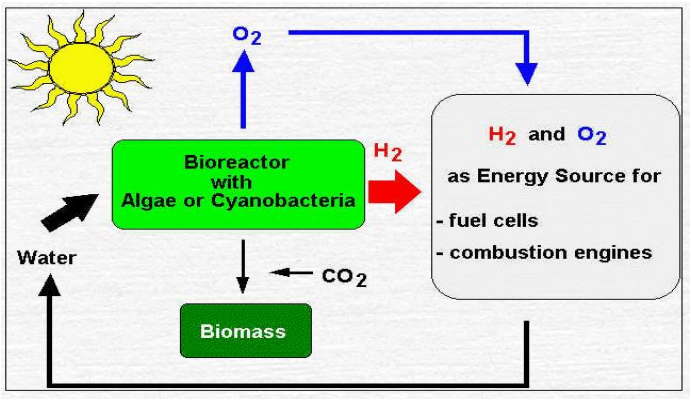

Figure 1.1 Schematic drawing of photo biological method [6]

1.1.5 Microbial Biomass Conversion Method

Hydrogen production by anaerobic microbial communities has drawn attention because hydrogen is important alternative source for decreasing environmental effect and organic waste streams can be used as substrate for hydrogen production [7]. Microorganisms break down bonds of organic wastes and release hydrogen gas. This process is also called dark fermentation. In Figure 1.2, microbial biomass conversion method is illustrated. In figure, water with organic wastes is cleaned by microorganisms. Microorganisms use these organic wastes for hydrogen production and produced hydrogen gas is used in fuel cell system for electricity.

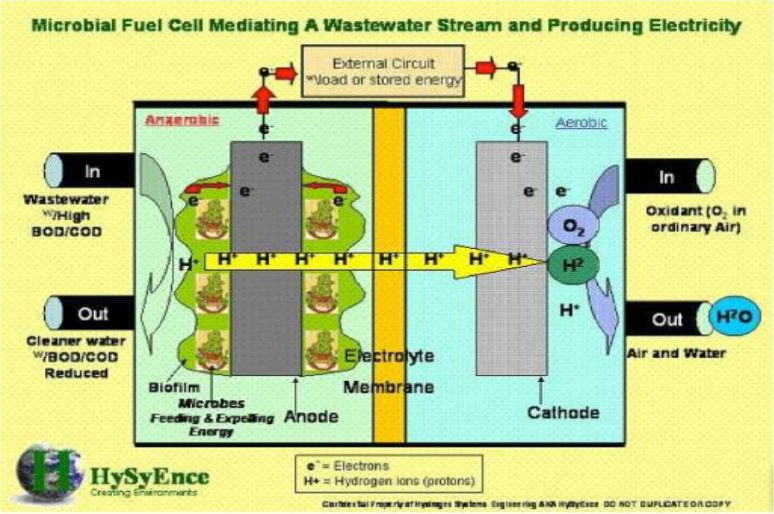

Figure 1.2 Schematic drawing of microbial biomass conversion method [8]

1.1.6 Electrochemical Method

Hydrogen production by electrolysis has been investigated for more than a century. Hydrogen is split from water by using electricity in electrolysis process. However, electrolysis process is not feasible at now when it is compared with other method such as steam reforming. This reaction takes place in a unit called an electrolyzer [9]. There are different types of electrolyzers.

- Polymer Electrolyte Membrane Electrolyzer
- Alkaline Electrolyzer

- Solid Oxide Electrolyzer

Electrolysis reaction: $2H_2O \rightarrow 2H_2 + O_2$ (1.6)

1.1.7 Photochemical Method

H_2 and O_2 can be split by using semiconducting catalysts in photocatalytic water splitting method. This method has received attention due to the potential of this technology. It may provide great economic and environmental opportunities about the production of clean fuel H_2 from water using solar energy [10]. Photo electrochemical materials absorb and use light energy to dissociate water molecules into hydrogen and oxygen. These materials are similar to photovoltaic solar electricity generators. In photochemical method, semiconductors are dipped in electrolyte. In Figure 1.3, photochemical method is shown.

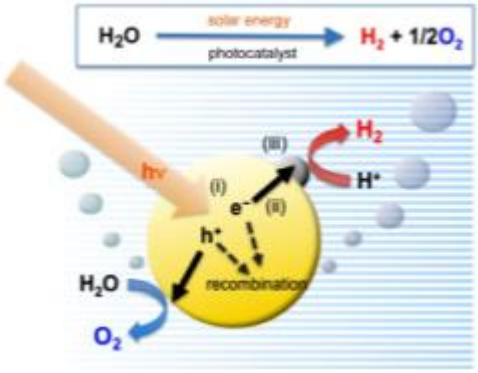

Figure 1.3 Schematic illustration of water spitting over semiconductor photo catalysts [10]

1.1.8 Thermochemical Method

Thermochemical method for hydrogen production is open to development and many researchers investigate currently about this area to improve it. This method consists of water splitting reaction and high temperature solar dissociation reaction (regeneration of metal-oxide catalyst). Regeneration of metal-oxide catalyst occurs at 1500-2000 °C. Hence, high amount heat is required to perform this reaction. Solar energy provided from parabolic dishes or nuclear waste heat is used to obtain this

high amount heat. Metal oxide is selected from between some metals such as Fe, Co, Ce, Pb, Zn, Ti [11]. In solar dissociation step, oxygen is released from metal oxide at high temperature. Then, steam is supplied to reactor and metal oxide captures the oxygen from steam. Finally, pure hydrogen without any separation is obtained in this process.

Solar dissociation reaction: $M_xO_y \rightarrow M_xO_{y-1} + 1/2O_2$ (1.7)

Water splitting reaction: $M_xO_{y-1} + H_2O \rightarrow M_xO_y + H_2$ (1.8)

In this thesis study, thermochemical method (water splitting reaction) is selected for hydrogen production because it has good improvement potential and some important system advantages.

1.2 Reactor Types for Water Splitting Reaction

Water splitting reaction is a complicated reaction system but it has some economic advantages. Redox metal couples are used as catalyst for this reaction. There are many common reactor types for water splitting reaction.

1.2.1 Internally Circulating Fluidized Bed Solar Chemical Reactor

A reactor that can be reached high temperature is needed for reduction of redox metal couples with thermal method. Internally circulating fluidized bed solar chemical reactor works with principle of heating suspension particles by solar energy. These particles are carried by gas flow. It is shown below (Figure 1.4) [12]. Therefore, high efficient heat transfer is provided for these high amount suspension particles. There are two different reactor types for internally circulating fluidized bed solar chemical reactor to use as lab scale.

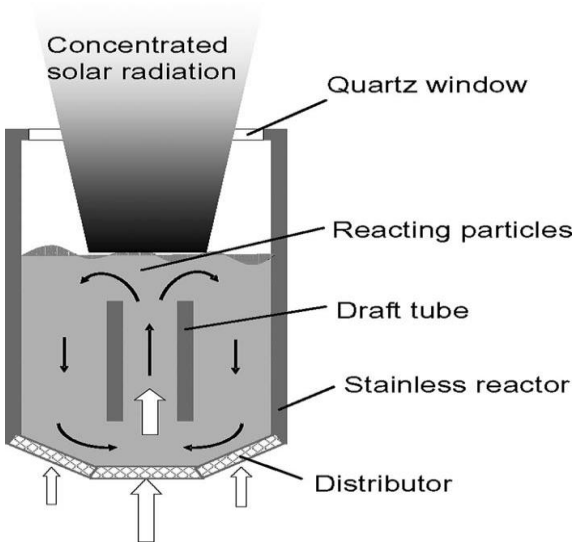

Figure 1.4 Internally circulating fluidized bed solar chemical reactor [12]

First one is reactor with quartz tube; the other is stainless tube reactor with window. Reactor with stainless tube is shown below (Figure 1.5) [13]. Ferrite particles of two reactor types always move upward from flow pipe and downward from annulus to complete circulation. Thus, heat transfer from solar energy is provided efficiently from top region of fluidized bed to bottom region of fluidized bed.

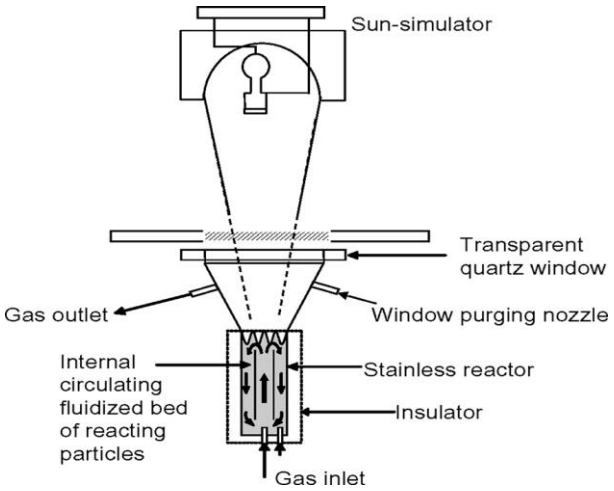

Figure 1.5 Lab scale stainless tube reactor [13]

Also, internally circulating fluidized bed solar chemical reactor has more uniform temperature distribution than other type of fluidized bed [14]. Another advantage of this reactor is to require high gas flow only around the flow pipe. So that, the requirement of total gas flow can be decreased. However, high amount inert gas is needed to prevent window pollution due to gas deposition which is occurred next to window after moving up to fluidized bed of carrier gases. For this reason, it is not obvious which reactor type is the most efficient totally.

Iron oxide is used as redox couple for internally circulating fluidized bed solar chemical reactor system. Generally, this substance is known as ferrite and its chemical formula is Fe_3O_4 but stabilized zirconia with cubic yttrium or calcium is added to ferrite since ferrite cannot provide sufficient reaction activity and performance of two steps reaction cycle mechanism [15]. Chemical formula of this material is (Fe_3O_4/c-YSZ).

For stainless tube reactor with quartz tube and window, metal oxide conversion is observed as %45 at 1000 °C. These values are obtained from results of lab scale experiment [16].

1.2.2 Monolith Reactor (Single and Two Chambers)

Monolith reactor studies are carried out in German Aerospace Center in Köln. Water splitting reaction on metal oxide is occurred by using solar energy in these reactor studies. Two types of reactor is designed for this monolith reactor. First one is single chamber monolith reactor. That single chamber monolith reactor is coated with multi channels ceramic support-active redox ferrite powder as catalytic converter in automobile exhaust systems.

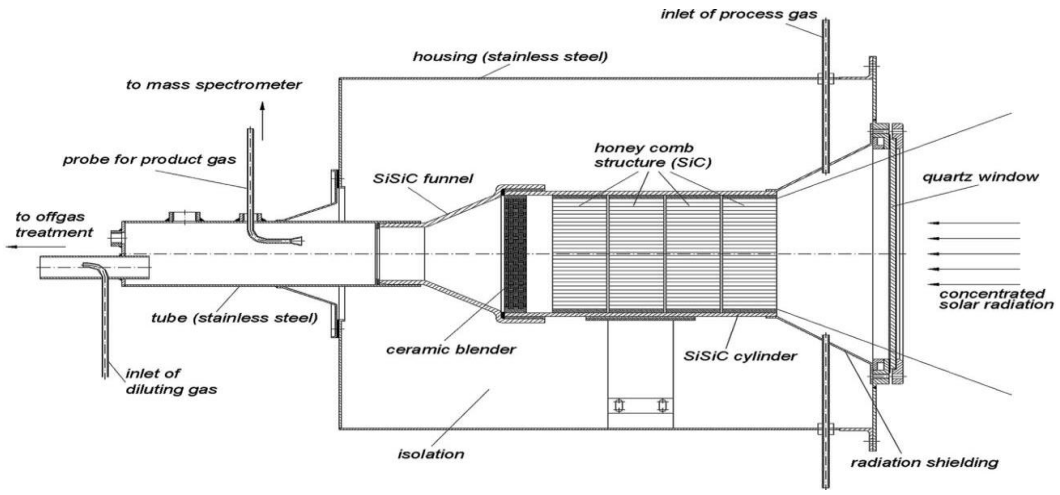

Figure 1.6 Ceramic support multi channels monolith solar reactor [17]

Two reaction steps of hydrogen production occur in same chamber for this reactor type [17]. It is shown above (Figure 1.6). This condition can contribute decreasing temperature in chamber. So that, reaction between hydrogen and oxygen is prevented because oxygen reacts with metal oxide and it is adsorbed. Monolith reactors have one main disadvantage. Ferrite couple used for catalyst is not inert to monolith reactor construction material (silicon carbide) at high temperature [17]. Second one is two chambers monolith reactor. Two different steps are occurred cyclically at same time by adding new monolith reactor in parallel. These steps are oxygen releasing from metal oxide and hydrogen production from steam at high temperature. Two chamber monolith reactor system is shown below (Figure 1.7) [18].

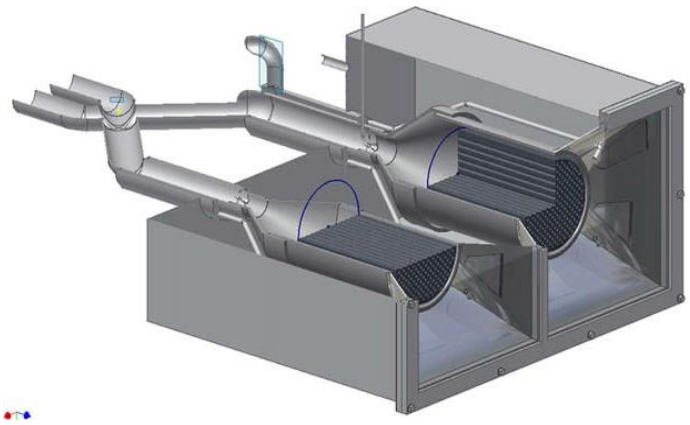

Figure 1.7 Solar reactor system with two monolith reactor chambers [18]

In first chamber, water splitting reaction occurs at 800 °C and in second chamber, regeneration reaction is occurred at up to 1200 °C for these solar reactor system with two monolith reactor chambers.

1.2.3 Counter Rotating Reactor

Mostly, counter rotating reactor studies are carried out in United States. This reactor consists of two circular rings to rotate counter direction and it is called as CR5. This reactor system can produce hydrogen gas continuously due to that rotation ability [19].

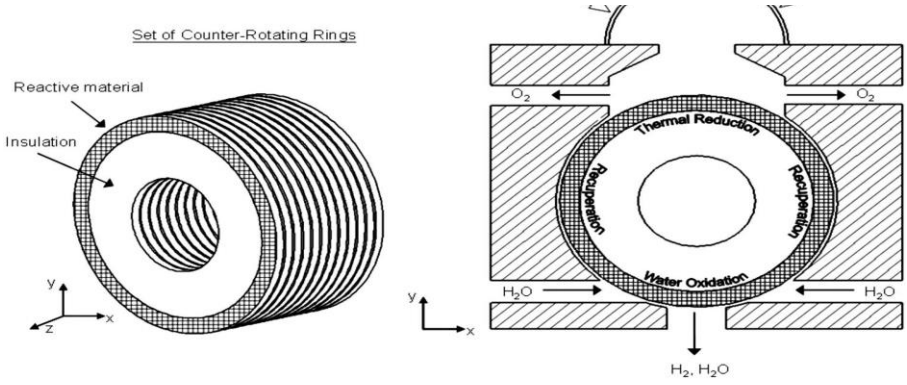

Figure 1.8 Counter Rotating Reactor (CR5) [19]

Counter rotating reactor includes two reaction beds with coated ferrite material to turn reverse direction. So that, heat transfer among reaction cycles can be efficient. This reactor is indicated above (Figure 1.8). Also, that working principle of counter rotating reactor provides to separate hydrogen and oxygen gases by itself. When energy is given as 1 kW to system, system efficiency is %30 for lab scale. Laboratory results indicate that system thermal stability increases by using monolith with cobalt-ferrite-zirconia and thermal cycles of system are occurred successfully [20]. However, rotating mechanism leads system mechanic instability. This problem is main disadvantage of counter rotating reactor systems.

1.2.4 Rotating Cavity Reactor (ROCA), ZIRRUS Reactor and Modified ZIRRUS Reactor

Iron and iron oxide metal couples are used as active material for above reactor type's studies. Also, zinc and zinc oxide material couples is used as catalyst commonly for water splitting reaction [21]. Rotating cavity reactor (ROCA) is one of the most important reactor types which are used this metal couple as active material. ROCA was designed in Paul Scherrer Institute and it is tested with 3500 kW/ m^2 solar flux in this institute [22]. ROCA has efficient heat distribution due to rotating ability. This is the main advantages of ROCA design. Nevertheless, rotating cavity reactor has main disadvantage. Products cannot be withdrawn sufficiently and easily from reactor. In order to eliminate this disadvantage, quenching unit is added to reactor system. Thus, risk of reaction between released zinc gas and oxygen gas is prevented. High amount inert gas is needed for system to separate zinc and oxygen gases due to quenching unit [23].

For this reason, Paul Scherrer Institute designed new ZIRRUS reactor with new approach. Purpose of ZIRRUS design is to maintain the advantages of ROCA due to rotating mechanism and eliminate disadvantages of ROCA and improve the reactor system with new concepts (Figure 1.9). Actually, ZIRRUS is research group name of Paul Scherrer Institute.

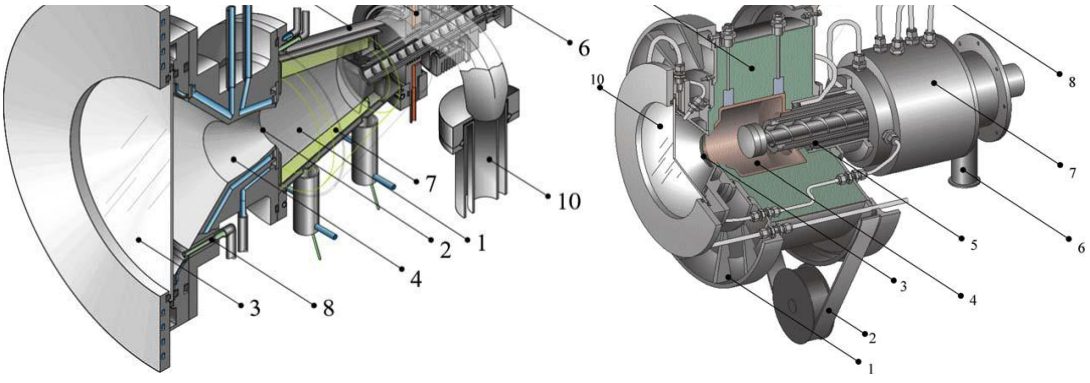

Figure 1.9 (Left) ROCA [23], (Right) ZIRRUS reactors [24]

ZIRRUS reactor is operated for 100 hours and can be heated to 2000 K. Zn / ZnO redox couples are used as active material in these experiments. Although reaction rate increases at high temperature, heat loss increases at that high temperature. In light of these experiments, optimum operational condition should be determined for this system [24].

Steinfeld and Müller state that desorption process of ZnO is occurred at 1700 – 1950 K. Zn and O_2 gases in reactor is cooled at Argon atmosphere by using fast cooling unit. Therefore, re-oxidation of solid product can be obtained as %39 [25].

According to ZIRRUS reactor heat transfer model, maximum reaction rate is 12 g/min and temperature is 2000 K for 9.1 kW heat flux but semi batch operation is not evaluated for this model. Conversion efficiency from solar energy to chemical energy is calculated as %14.8 for this conditions (assuming no react between Zn and O_2)[26].

According to detailed heat transfer analysis of combined radiative heat transfer-reaction kinetic, conversion efficiency from solar energy to chemical energy is calculated as %14 at 1900 K for ZIRRUS reactor [27].

There are some big and small scale modeling studies for steady operation and non-steady operation of ZIRRUS reactor. Those studies show that small scale model has the most heat loss as %30-%40 for 1kW heat energy [28]. In addition, radiative heat loss is important heat loss mechanism for big scale models (50 MW) and this loss depends on focusing solar energy.

ZIRRUS reactors have some mechanic stability problems at more than 2000 K which is needed for desorption of ZnO material. High temperature differences among heating and cooling cycles and oxidative impact of reactor can lead mechanic problems. Modified ZIRRUS reactor and its parts are shown below (Figure 1.10.).

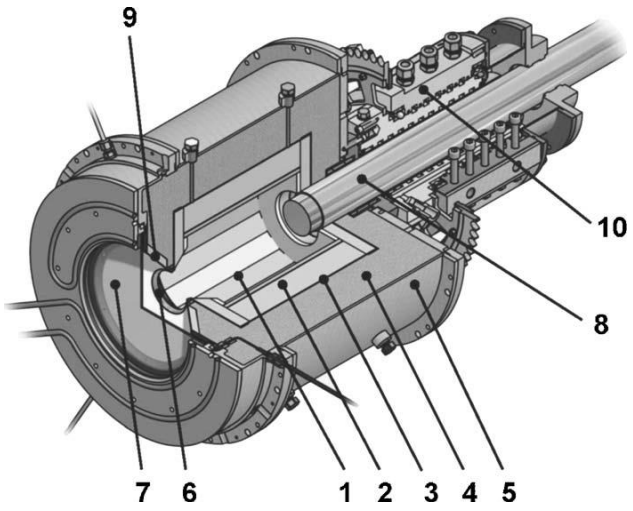

Figure 1.10 Modified ZIRRUS reactor [29]

ZIRRUS reactor is developed and modified ZIRRUS reactor is designed to eliminate some mechanical problems [29]. Fundamental properties of modified ZIRRUS model are new design rotating cylindrical cavity. This structure consists of sintered ZnO with porous tile, %80 Al_2O_3 - %20 SiO_2 isolation material and %95 Al_2O_3 - %5 Y_2O_3 ceramic matrix composite coating material. After this improvement process, not only mechanical and thermal stability issues are solved but also diffusion barrier is provided for product gases [29].

For modified ZIRRUS model, cavity includes circular space. So that, solar heat flux can be entered directly and reached into cavity. Moreover, reactor has dynamic feed and reciprocate toward cavity inside to make uniform thickness ZnO layer [29]. Rotational movements create centrifugal force and that force provides coating ZnO particles on cavity wall. Thus, radiation heat transfer efficiency increases because of this property [29].

Modified ZIRRUS reactor was tested for 23 hours and its temperature is between 1807 and 1907 K. Any mechanical cracking or deformations were not seen. Only condensation of Zn gas was observed but this situation did not lead such a big problems [29].

Some experiments were made at 1600 – 2136 K to analyze thermal performance of modified ZIRRUS reactor model. This model has unsteady state operation of

desorption reaction and three dimensional heat transfer phenomena. According to those experiments, conversion efficiency from solar energy to chemical energy is %56 for 1MW solar energy [30]. Also, ZnO layer depletion and semi batch operation of reactor were considered in those experiments.

1.2.5 Single and Multi-Absorber Reactor, Lab Scale Co-Synthesis Reactor and KIER Reactor

Upon the above reactor types are examined, even though direct solar energy to reactor is important heat source for reaction, reactor glass (quartz glass) is very critical and problematic element regarding operational conditions. This element obstructs increasing reactor system scale at high pressure and dense gas atmosphere [31]. In the light of the research, new cavity mechanism is designed to solve this problem. It includes opaque absorbers as reaction chamber [32]. Absorber reactor has thinner isolation layer on cavity wall than the above other reactor systems. So that, ceramic material is not needed for the basic material of the reactor and heat capacity is decreased. Also, reactor geometry can be adapted for direct absorption processes due to illuminating absorbers directly.

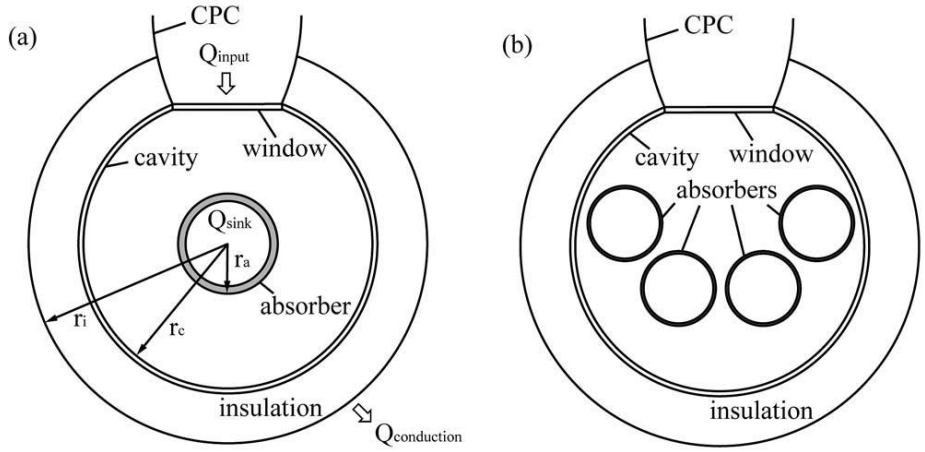

Figure 1.11 Single and multi-absorber reactors [33]

Solar flux was adjusted between 448 – 2125 kW /m^2 for laboratory scale designed 5 kW reactor and whole heat transfer mechanisms were evaluated to configure heat

transfer model of reactor system. Monte Carlo ray tracing algorithm and finite element method is used in configuration of this heat transfer model [31]. Tube number, maximum energy efficiency and maximum absorber temperature are optimized by using results of modeling and experiment. The analysis results show that diffuse reflective cavity can ensure uniform temperature distribution around the absorbers with tube [31].

In one of Haussner and his lab members studies, multi-tube solar reactor is used for desorption reaction of ZnO material [33]. It is shown above (Figure 1.11). 3000-6000 solar flux range, 1-10 tube number, 2 – 20 g /min ZnO flow rate and 0.06 – 1 μm ZnO particle size range is preferred for parametric studies to investigate the effect of solar flux and other parameters on reactor performance. Time of reaction retain is 1 second and final temperature is more than 2000 K, conversion efficiency from solar energy to chemical energy is reported as %29.

According to Wegner [34] and Ernst studies [35], hydrogen production can be occurred as in-situ after Zn hydrolysis which occurs by using Zn nanoparticles and steam.

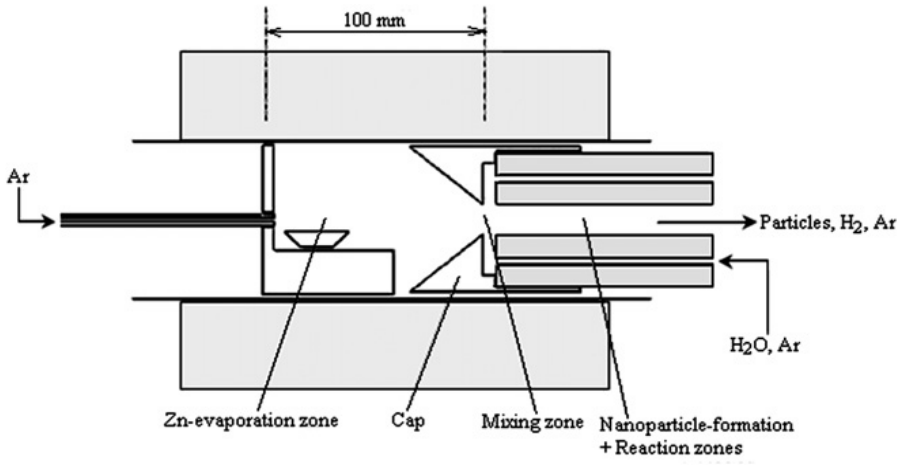

Figure 1.12 Laboratory scale Zn nanoparticle reactor and hydrolysis operational scheme [34]

Those combined processes are performed experimentally in aerosol flow reactor with tube which includes evaporation region of Zn, cooling area of steam and Zn / H_2O reaction area. This reaction system provides continuously required stoichiometric feed of reactant and product withdrawal.

Maximum chemical conversion value from Zn to ZnO is %83 at 1023 K for this configuration. It is indicated above (Figure 1.12) As a result of studies, controlling reactor temperature and production of smaller size nanoparticles are needed to reduce thickness of extra ZnO layer that is passivized on Zn [34].

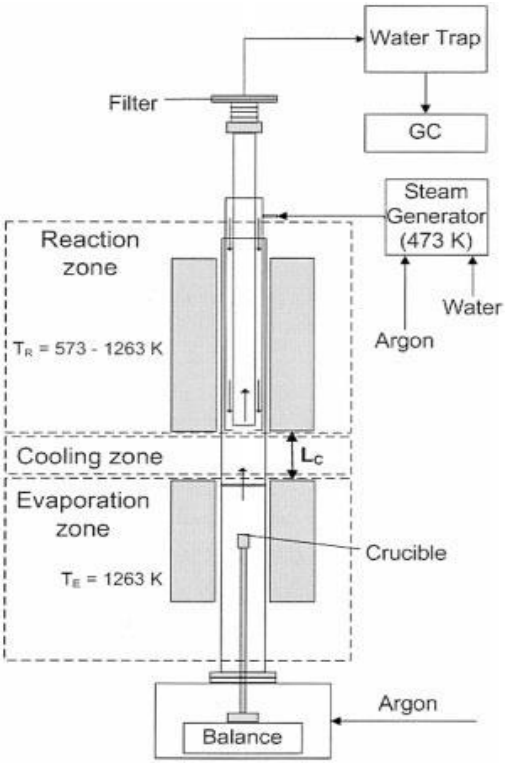

Figure 1.13 H_2 and Zn / ZnO co-synthesis reactor [35]

Moreover, co-synthesis of H_2 and ZnO nanoparticles with steam hydrolysis on hot aerosol reactor is investigated. This process has %90 H_2 conversion value. It is indicated above (Figure 1.13). Average particle sizes of Zn and ZnO nanoparticles are 100 nm and 40 nm, It contains %80 ZnO in terms of weight [35].

H_2 conversion values are obtained among %61 – 79 in synthesis and hydrolysis studies of zinc nanoparticles in tube reactor [36]. Retention time of the particles in flight is less than 1 second and particles are hydrolyzed partially. Common gas deposition on reactor window is observed at 650 K in high hydrogen conversion experimental conditions. The other result of studies is to enhance flow amount of Argon gas for obtaining high cooling performance. If flow rate of coolant fluid increases, retention time of particles in reactor flow field is reduced [36].

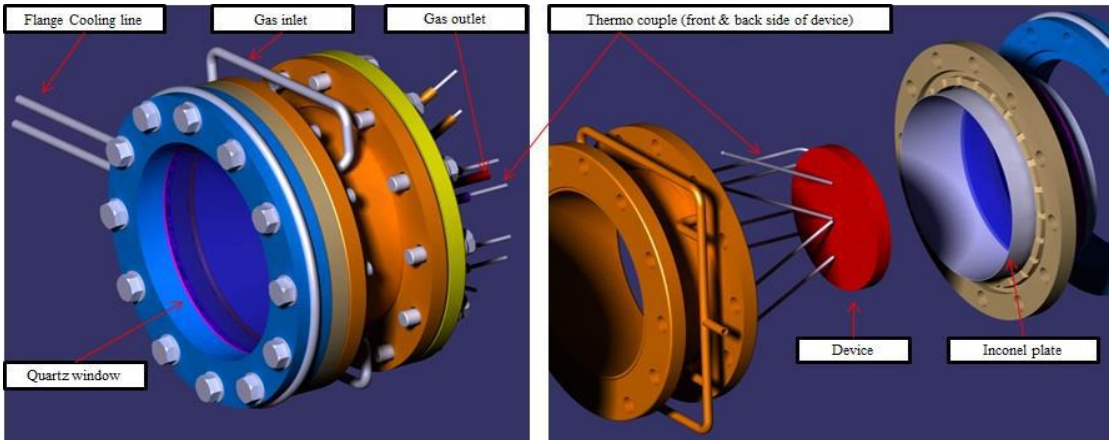

Figure 1.14 KIER 4 cycle reactor [37]

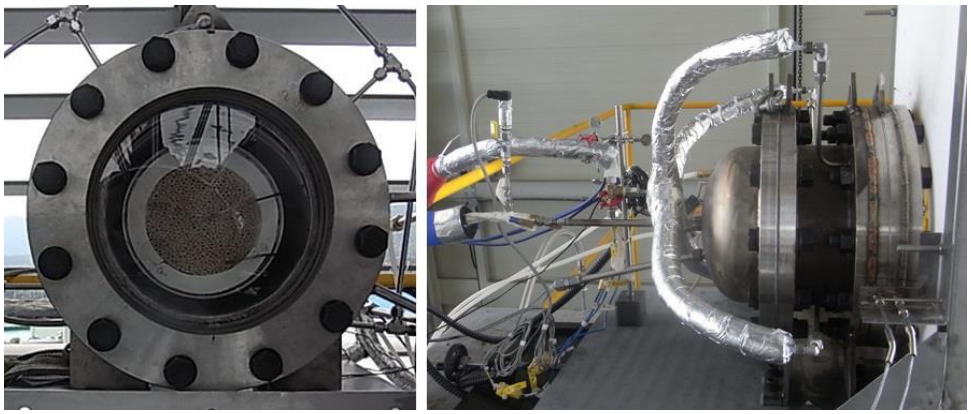

Figure 1.15 KIER 4 cycle reactor-2 [37]

In Kang studies [37] new two steps cycle is tried and GeO_2 / GeO couples is used as redox couples. Advantages of KIER 4 cycle were reported as the low reduction temperature and high thermal efficiency. Reduction temperature of KIER 4 cycle

reactor is between 1400 – 1800 °C for conversion from GeO_2 to GeO. Expansion of KIER is Korea Institute of Energy Research. In this study, reported maximum energy conversion efficiency is calculated approximately %34.6 after analyzing of second law of thermodynamic. KIER reactor is indicated above (Figure 1.14 and Figure 1.15).

1.3 Modeling Studies of Monolith Reactors for Gas-Solid Reactions

Monolith reactors contain catalysts with certain structures or arrangements. There are many different types of monolith reactors, such as honeycomb, foam, and fiber reactors. The monolith channels normally have circular, square or triangular cross-sections. There are different modeling scales of monolith structure, catalyst layer, single channel, multichannel or even entire reactor. If the physical and reaction behaviors in the catalyst layer are only concerned, catalyst layer scale modeling can be selected. At this scale, only the local nature in catalyst layer is described and modeled. For examples, through the interactions between the internal diffusion and reaction, the effectiveness factor of monolith catalyst can be calculated. In order to describe the behaviors of a monolith reactor, single channel scale modeling is applied. At this scale of modeling, it is postulated that every channel in the monolith reactor has similar temperature and flow profiles and can represent the whole reactor. The physics of inside the catalyst are considered. In addition, the mass and heat transfers between the catalyst and the fluid are also considered. However, if model has non-uniform inlet gas distribution, blocked or deactivated channels, modeling a single monolith channel might be insufficient. In this case, all of the channels have to be modeled. Such models are considered at the reactor scale. In order to evaluate accurately the differences in flow and temperature in different channels, multi-channel model can be selected.

Chen and Yang have studied on single channel scale for monolith reactor [38]. For single channel scale, the reaction is considered to be occurring in the entire monolith wall. If the monolith wall is thick enough, internal molecular diffusion effect on the reaction has to be considered. In the monolith channels, there are only transport of mass, heat and momentum. Generally, laminar gas phase flow is present inside the channel since the monolith channel is relatively small [38]. The reactant molecules

diffuse from the channel to the monolith wall surface and further diffuse into the entire monolith wall to undergo reactions. The product molecules diffuse in the opposite direction. For exothermic reactions, heat is transferred through conduction and radiation for extremely high temperatures from the monolith wall to the flowing gas while for endothermic reactions heat is transferred from flowing to the monolith wall. Model equations are listed below.

Momentum balance:

$$\rho(\mathbf{u}\cdot\nabla)\mathbf{u} = \nabla\cdot\left[-\mathrm{P}\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^{\mathrm{T}}) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right] + \mathbf{F}_{\mathrm{b}} \qquad (1.9)$$

u represents velocity vector, P is pressure, $\mathbf{F}_{\mathrm{b}}$ is body force and **I** is unit matrix.

Mass balance:

$$\frac{\partial \mathrm{C_i}}{\partial \mathrm{t}} + \nabla\cdot(-\mathrm{D_i}\nabla\mathrm{C_i}) + \mathbf{u}\cdot\nabla\mathrm{C_i} = \mathcal{R}_{\mathrm{i}}\,,\ \mathbf{N_i} = -\mathrm{D_i}\nabla\mathrm{C_i} + \mathbf{u}\mathrm{C_i} \qquad (1.10)$$

In the above equation, C represents concentration, D symbolizes diffusion coefficient, u is velocity, t is time, R is reaction rate equation and N symbolizes mass flux.

Energy balance:

$$\bar{\rho}_{\mathrm{j}}\overline{\mathrm{C_{p_j}}}\frac{\partial \mathrm{T_j}}{\partial \mathrm{t}} + \bar{\rho}_{\mathrm{j}}\overline{\mathrm{C_{p_j}}}\mathbf{u}.\boldsymbol{\nabla}\mathrm{T_j} = \bar{\mathrm{k}}_{\mathrm{j}}\boldsymbol{\nabla}^2\mathrm{T_j} \qquad (1.11)$$

In the above equation, $\mathrm{C_p}$ represents heat capacity, ρ symbolizes density, u is velocity, t is time, T is temperature and k symbolizes thermal conductivity.

Jahn and Snita have studied on multichannel scale for monolith reactor [39]. For this scale, the reaction is defined to be occurring only at the monolith wall surface without any internal molecular diffusion effect. Similar equations and boundary conditions are used when multichannel scale is compared with single channel scale except catalyst layer boundary condition [39]. Numerical solution of this model scale requires high computer time. Well-founded simplification method is used to obtain a more easily solvable model. Averaging and pseudo-stationary methods are used to simplify the numerical solution in this model scale.

1.4 Purpose of the Study

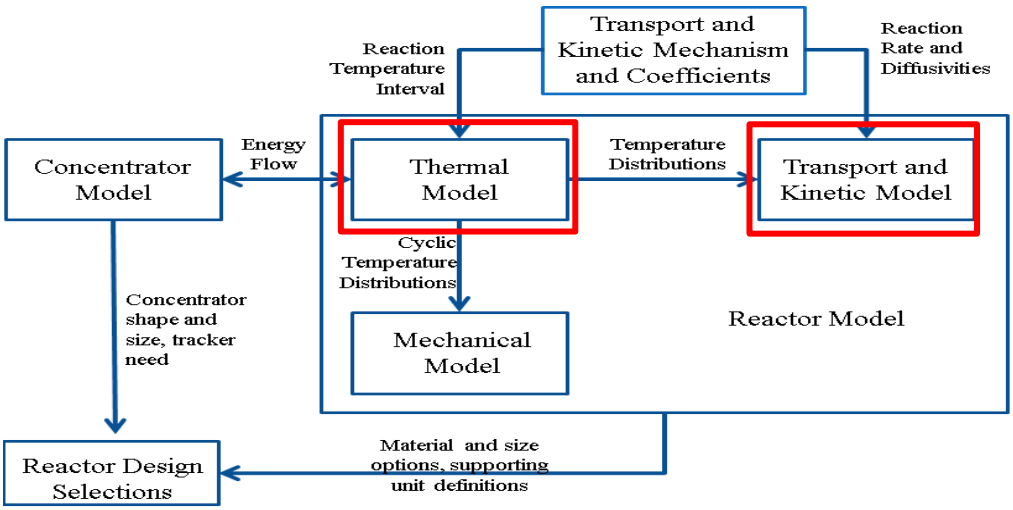

Figure 1.16 Thesis Scopes Chart

The ultimate aim of the project is to decide reactor design selections but this thesis concentrates modeling and design optimum reactor system in terms of heat, mass and momentum phenomena via statistical approach, JMP, COMSOL and MATLAB program. Reaction temperature interval data and reaction rate expressions are taken from UNER Research group. Energy flux of concentrator model is taken from concentrator modeling group. Thermal model is configured on COMSOL by using energy flux and reaction temperature interval data. Then, mass and momentum transport model is configured on COMSOL and MATLAB by using thermal model results and reaction rate expressions. Finally, optimized reactor design selections are identified.

In accordance with this purpose, in this Chapter, general information about hydrogen production method, reactor types for water splitting reaction and model studies of monolith reactor are given. In the second chapter, approaches and configurations of modeling in COMSOL, JMP and MATLAB are discussed. Then, results are presented in Chapter 3, followed by conclusions in Chapter 5. MATLAB codes are presented in the Appendices.

CHAPTER 2

MODELING APPROACH AND DEVELOPMENT

Water splitting monolith reactor system is modeled in three dimensions to analyze effects of several variables on maximum temperature of reactor and heating-cooling time. These variables are material selection for reactor and insulation, channel shape, wall thickness and CPSI (cell per square inch), thickness of insulation material, thickness of thin layer insulation material and solar heat flux for heating reactor. These variables relate to thermal model because thermal modeling is the most important phenomena in this reactor system due to occurring reaction at high temperature. All of variables are evaluated according to statistical approach by using JMP program to optimize reactor system for thermal modeling. Thermal model is configured parametrically through COMSOL program. In addition, kinetic model is configured via MATLAB program by using thermal results data of COMSOL program to project conversion values. Transport model in a single channel is configured through COMSOL program to examine mass, momentum and heat transfer phenomena together. It includes reaction kinetic and diffusion, thermal model and assumed flow model (velocity and pressure).

Solvers of computational programs are defined carefully in terms of optimum accuracy, speediness and parallelism due to complex calculation of radiation heat transfer and many variables. Although these solvers are optimized, these computational programs require high computer capacity. For instance, computer with six cores 2.6 GHz Intel Zeon CPU needs 14 hours calculation time and 60 GB RAM for computation of 200 CPSI 0.5 mm monolith reactor system with unsteady-state operation.

2.1 Statistical Approach

Design of experiment (DOE) is used as alternative method instead of making real experiment on new produced reactors to determine optimum reactor design. Thus,

not only time is used effectively but also money is saved and cost is decreased. Firstly, physical theoretical background is studied and important parameters of this are listed in design of experiment method. Then, simulation runs including different combinations of limited values of parameters are set. These simulation sets are modeled through COMSOL program and model configurations via COMSOL program are run. After that, obtained results data from COMSOL are evaluated by using statistical analysis on JMP program. Therefore, impacts of all parameters on thermal model are analyzed and important parameters are determined.

In this statistical approach, main effects, second order effects and nonlinear effects of parameters are considered by setting simulation runs. Firstly, eight simulation runs for each of the three materials (total twenty-four runs) are set and configured for consideration of main effects. Then, twenty-two simulation runs for each material (total sixty-six runs) are set and configured for consideration of main effects and second order effects. These runs include all parameters except channel shape. Finally, one hundred-sixty-two simulation runs including channel shape parameter are set and configured for consideration of main, second order and nonlinearity effects. These simulations results are analyzed in terms of temperature distribution among channels, temperature distribution along the length of one channel, heating and cooling times. MATLAB is used for this analysis. Significant parameters are determined according to statistical analysis results. These parameters were tabulated in Table 2.1 and 2.2 below.

Table 2.1 Main and second order effects of significant parameters for temperature distribution and heating time

Effect	**Parameters**
Main Effects	Material
	Wall Thickness
	CPSI
	Solar Flux
	Operation Temperature (Reaction)

Table 2.1 Main and second order effects of significant parameters for temperature distribution and heating time (continued)

Second Order Effects	Solar Flux x Operation Temperature
	Wall Thickness x Operation Temperature
	Wall Thickness x Solar Flux
	CPSI x Wall Thickness
	CPSI x Operation Temperature

Table 2.2 Final DOE including main, second order and nonlinearity effects of significant parameters for temperature distribution and heating time

Parameters	**Low Value**	**High Value**
Reactor Material	Stainless Steel, Cordierite, Silicon Carbide	
Channel Shape	Square,Triangle,Hexagon	
Channel number	20 CPSI	200 CPSI
Wall Thickness	0.5 mm	1 mm
Length	Total Reaction Surface 89 cm^2	
Inner Insulation Thickness	5 mm	10 mm
Outer Insulation Thickness	2 mm	
Solar Flux	250 Sun	350 Sun
Operation Temperature	900˚ C	1100˚ C
Temperature Difference of Two Cycles	200˚ C	

2.2 Thermal Model Configuration

Optimized monolith reactor model is configured and computational solutions of this model are done by using COMSOL program which uses finite element method. Front surface area of reactor model is fixed as 7 cm^2 instead of total channel number to limit reactor model sizing. Different geometric shapes are drawn as square, triangle and hexagon. Different types of materials are selected as cordierite, silicon carbide and stainless steel. CPSI (cell per square inch) is defined to specify tube number. This parameter is significant for industrial design of monolith reactor. Definition of variables, drawing geometries, configuration of physics, optimizations of mesh and solver options are explained and shown below parts.

2.2.1 Variables

Firstly, steady state calculation is done by using COMSOL to create basis for unsteady state calculations. Model includes three cycles time such as first heating, first cooling, and final heating in unsteady state calculations. In that part, CPSI, channel geometry, wall thickness, thickness of inner insulation, thickness of outer insulation, material types of monolith reactor, solar flux, reduction and oxidation temperature are determined as primary important parameters. Limit values are defined for these variables to identify significant parameters among them. Cordierite, silicon carbide and stainless steel are selected for material types of monolith reactor. Moreover, some different variables are defined to configured geometry, mesh and physics. They are listed in below Table 2.3.

Table 2.3 Sample table of model variables

Parameter	Value/Definition	Description
CPSI	20-200	Cells per square inch
a_sq	r*sqrt(2)	Length of a cell edge
r	(sqrt((CPSI/0.0254^2)^-1)-dw)/sqrt(2)	Radius of the circumcircle
L	5*(2*r)	Reactor length
dw	0.5-1[mm]	Wall thickness
dins	0.005-0.01[m]	Thickness of the insulation layer
FluxArea	7[cm^2] (fixed)	Incident flux area

Table 2.3 Sample table of model variables (continued)

Parameter	Value/Definition	Description
MonoD	2*sqrt(FluxArea/pi)	Monolith diameter
hair	10[W/(m^2*K)]	Convection coefficient
Tair	298[K]	Air temperature
Tfront	298[K]	Reactor gaseous temperature
T0	500[K]	Initial temperature
flux	250000-350000[W/m^2]	Solar flux
db	dw	Lid thickness
d_coat	1-3[mm]	Thickness of the insulated coating
k_coat	0.01[W/m*K]	Thermal conductivity of the coating
Tred	900-1100[degC]	Reduction temperature
Tox	700-900[degC]	Oxidation temperature

2.2.2 Geometry

Different geometry types of reactor are defined three dimensional parametric in COMSOL program with respect to some variables such as CPSI, wall thickness, reactor length, insulation thickness, flux area. Reactor model includes four domain bodies that are monolith reactor, insulation material, front lid and back lid. These lids are imaginary lids. Front lid is used to distribute radiative heat flux to channel inside by using COMSOL view factor formulas. Those formulas were justified physically via MATLAB codes. Besides, back lid is used as insulator for backside of reactor model. Three different channel shapes are used for this reactor model. They are square, triangle and hexagon. They are shown below in Figure 2.1, Figure 2.2, Figure 2.3 and Figure 2.4.

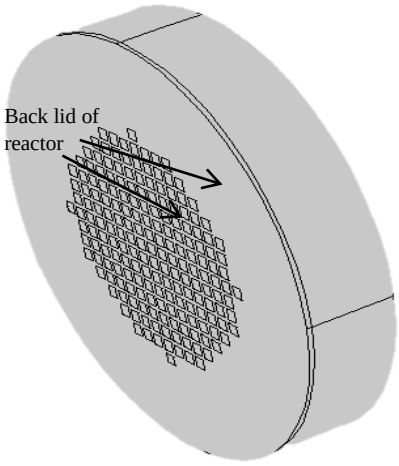

Figure 2.1 Backside of 200 CPSI square model

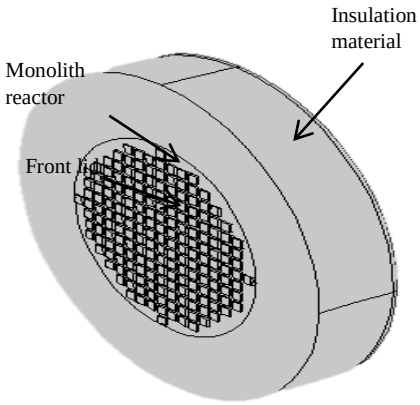

Figure 2.2 Front view of 200 CPSI square model

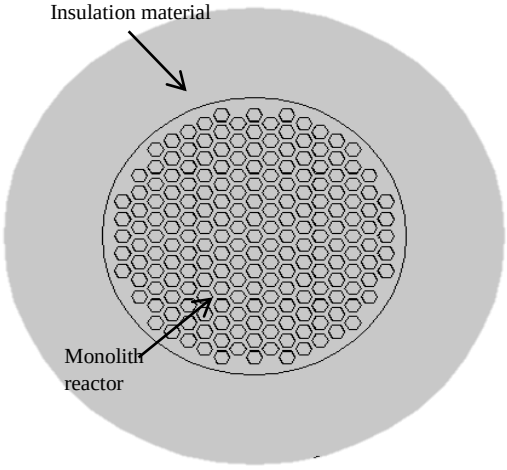

Figure 2.3 Front view of 200 CPSI hexagon model

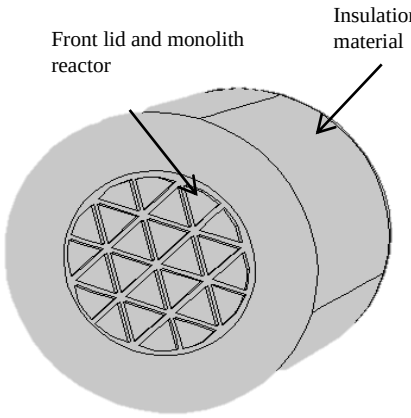

Figure 2.4 Front view of 20 CPSI triangle model

During drawing these geometries, specific nodes of COMSOL geometry part are used such as compose, convert to curve or solid, difference, union, array, copy and extrude etc.

2.2.3 Physics

That reactor model has so complex thermal physical phenomena including radiative flux to monolith reactor channels with view factor, surface to surface radiation in

monolith reactor channels with view factor, conduction, convective and radiative loss to ambient, inward heat flux from the sun to the front face of monolith reactor. In this model, only heat transfer from solid phenomena is used because fluid effects on heat transfer are neglected. System is operated at high temperature (1100 °C) and system fluids have low velocity in channel. Hence, those fluids has laminar flow regime and their heat transfer coefficients might be small. Actually, there is not any flow in first heating or final heating cycle. There is only oxygen production from metal oxide oxygen desorption reaction in those cycles and oxygen gas can spread to reactor channel via primarily diffusion due to very low velocity. In cooling cycle, steam has not regular flow. It is sprayed at the beginning of cycle. Therefore, neglecting of fluid impacts on heat transfer is acceptable. This assumption is justified using the simulations in transport model part. System operates unsteadily. Emissivity values of all material are taken as 1 because reactor system is blackbody. Natural heat transfer coefficient of air is taken constant as 10 (W/m^2*K). Solar flux is assumed as 1000 W/m^2 according to solar irradiance data from solar concentrator. Temperatures of ambient and initial temperature of reactor are assumed as 25 °C and 200 °C. Radiation groups selection of COMSOL program is used in that reactor model to shorten running time of program.

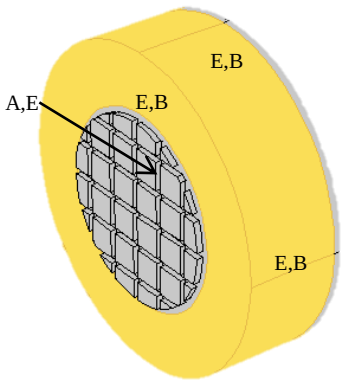

Figure 2.5 Boundary Conditions for 20 CPSI square model

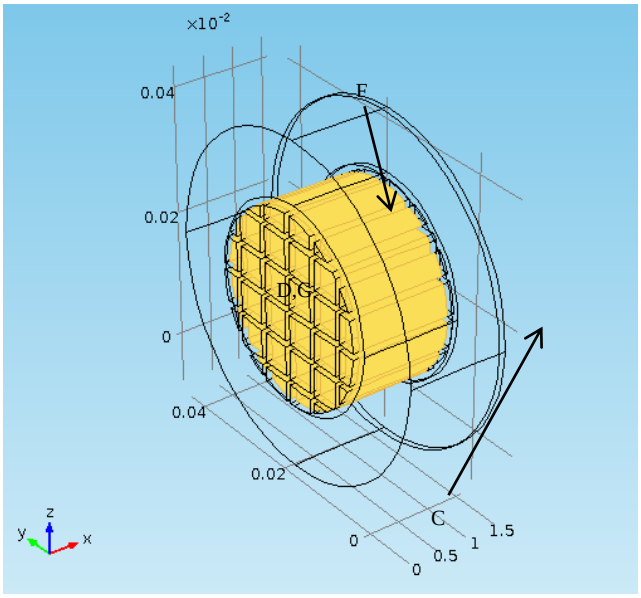

Figure 2.6 Boundary Conditions-2 for 20 CPSI square model

General heat conduction equation shown below was solved with all boundary conditions through COMSOL program:

$$\bar{\rho}_j \overline{C_{p_j}} \frac{\partial T_j}{\partial t} = \bar{k}_j \nabla^2 T_j \tag{2.1}$$

For monolith reactor j = 1, for insulation j = 2 were defined. Physical properties with over bar symbolizes average values for density, heat capacity and thermal conductivity. Boundary conditions of reactor model are shown below:

A boundary $$-\mathbf{n} \cdot \left(-\bar{k}_j \nabla T_j\right) = q_0 \tag{2.2}$$

B boundary $$-\mathbf{n} \cdot \left(-\bar{k}_2 \nabla T_2\right) = h(T_{amb} - T_2) \tag{2.3}$$

C boundary $$-\mathbf{n} \cdot \left(-\bar{k}_1 \nabla T_1\right) = 0 \tag{2.4}$$

D boundary $$-\mathbf{n} \cdot \left(-\bar{k}_1 \nabla T_1\right) = \varepsilon\left(G - \sigma {T_1}^4\right) \tag{2.5}$$

E boundary $$-\mathbf{n}\cdot\left(-\bar{k}_j\nabla T_j\right)=\varepsilon\sigma\left({T_{amb}}^4-{T_j}^4\right) \quad (2.6)$$

F boundary $$T_1=T_2 \quad (2.7)$$

G boundary $$J=J_0 \qquad \varepsilon(T)=\mathbf{n^2}\sigma T^4 \quad (2.8)$$

A boundary condition represents solar flux to front face of monolith reactor from concentrator. B boundary condition represents natural convection at lateral and front face of insulation. C boundary condition represents well insulation at reactor back lid. D boundary condition represents surface to surface radiation at lateral surfaces of channels. E boundary condition is defined as radiative loss at lateral and front face of insulation. F boundary condition is defined as temperature equality at adjacent surfaces. Finally, G boundary condition is defined as heat flux from front lid to inside of channels and emission of channel surfaces.

G term in D boundary condition is surface irradiation term and it is very significant for radiative heating reactor. All surfaces of reactor channels have interaction of heat transfer with the surfaces they see as surface to surface radiation. G term consists of this interaction heat transfer (surface to surface radiation) and emission of ambient temperature. This emission is approximately zero. G term is shown by the following equation:

$$G=G_m+F_{amb}\sigma T_{amb}^4 \quad (2.9)$$

F_{amb} view factor to channel surfaces from surround:

$$F_{amb}=1-\int_S\frac{(-\mathbf{n}\cdot\mathbf{r})(\mathbf{n'}\cdot\mathbf{r})}{\pi|\mathbf{r}|^4}dS \quad (2.10)$$

G_m is radiative heat flux from the other channel surfaces to arbitrary channel surface:

$$G_m=\int_S\frac{(-\mathbf{n}\cdot\mathbf{r})(\mathbf{n'}\cdot\mathbf{r})}{\pi|\mathbf{r}|^4}J'dS \quad (2.11)$$

Variables with apostrophe symbolize normal vector and radiosity of seen surface. Radiosity is shown the following equation:

$$J = \gamma G + \varepsilon \sigma T^4 \tag{2.12}$$

γ is reflectivity coefficient, ε is emissivity coefficient. Correlation among emissivity and reflectivity is $\varepsilon = 1 - \gamma$ due to using opaque body. In addition, emissivity was taken as 1 in this system since channels of reactor system have been painted black.

2.2.4 Mesh Optimization

Various simulations involving the sequential radiative heating and cooling of a cordierite square channel monolith (20 CPSI/1 mm wall thickness) have been run by using different meshes to make sensitivity analysis and optimize mesh parameter.

Firstly, mesh configuration was characterized. Meshes of 5 faces/domains are refined and studied. These are the monolith face mesh, swept body mesh, lid mesh, insulation mesh and insulated back mesh. Except for the main body involving the insulation and the monolith, triangular meshes are drawn on the faces of domains. These are used as in their default “extremely fine” setting (physics: fluid mechanics). The main parameter defining the coarseness of the mesh is the maximum element size. Only this parameter is modified, and the rest, namely the maximum element growth rate, curvature factor and resolution of narrow regions are kept as default. The standardization of the maximum element size ($h_{c_{max}}$) is made as:

$$h_{c_{max}} = \frac{L_c}{m} \tag{2.13}$$

Where L_c is the characteristic length of the face, m is an adjustable parameter and it takes different sets of values for each face in this parametric study. The $h_{c_{min}}$ is given an appropriate small value, again based on L_c. The "m" of each face/domain and some samples are given below.

1) m_1: Fineness of the monolith face mesh. $L_c = d_w$, m_1 takes the following set of values: $1, 1.5, 2$.

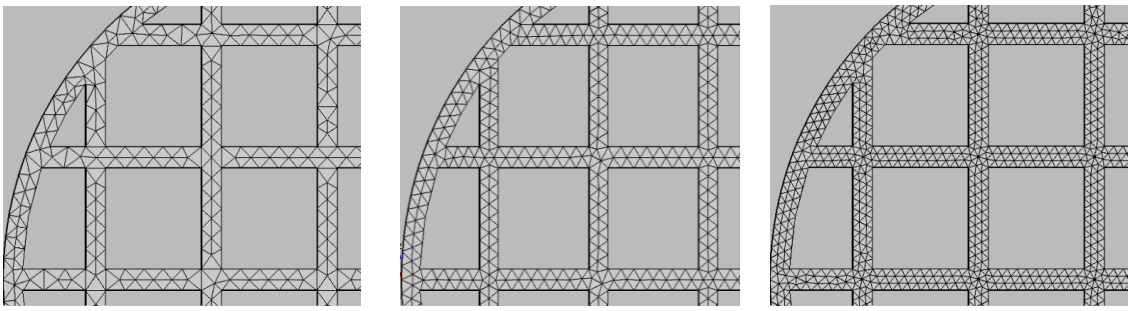

Figure 2.7 Monolith face meshes with $\boldsymbol{m_1 = 1, 1.5, 2}$ respectively

2) m_2: Number of cells in the axial direction for the monolith-insulation domain. This parameter is different than the one in the formulation given above; it simply gives the number of swept elements in x-direction. m_2 takes the following set of values: $10, 15, 20, 25, 30$.

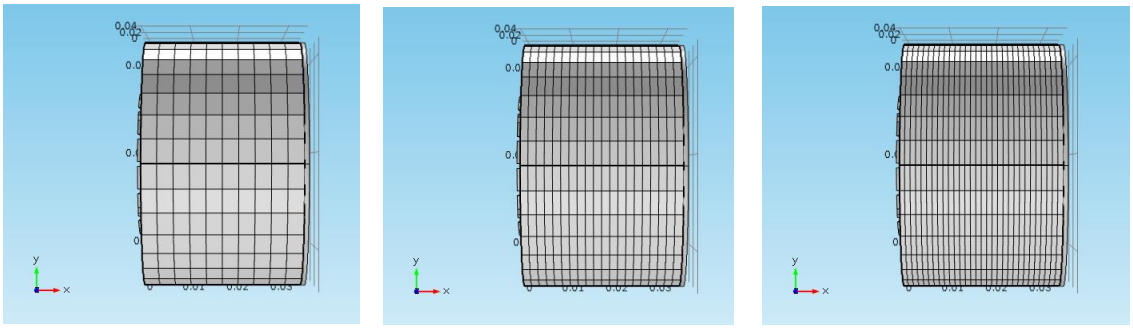

Figure 2.8 Sweep mesh with $\boldsymbol{m_2 = 10, 20, 30}$ respectively

3) m_3: Fineness of the mesh of lids (both prescribed radiosity and back faces of the channels). $L_c = r\sqrt{2}$, m_3 takes the following set of values: $2, 3, 4$. This parameter is a bit limited by m1. Higher values lead to very non-uniform and oblique meshes.

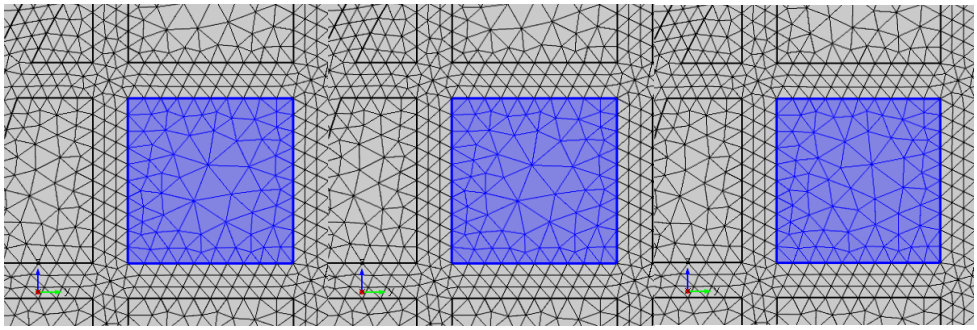

Figure 2.9 Lid mesh with $\boldsymbol{m_3 = 2, 3, 4}$ respectively

4) m_4: Fineness of the insulation mesh. $L_c = d_{ins}$, m_4 takes the following set of values: $2, 4, 6$.

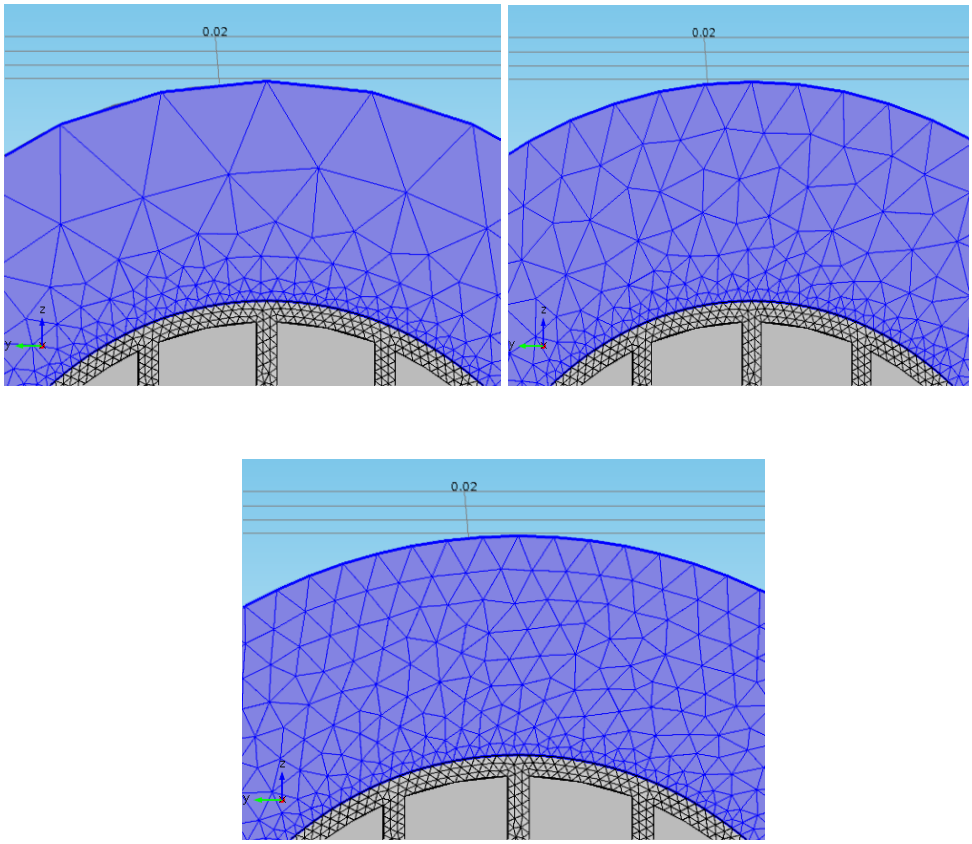

Figure 2.10 Insulation face mesh with $\boldsymbol{m_4 = 2, 4, 6}$ respectively

5) m_5: Number of cells in the axial direction for the back lid. This parameter is also different than the one in the formulation given above; it simply gives the

number of swept elements in x-direction. m_5 takes the following set of values: $1, 3, 5$.

As a result, comparison is based on the cycle duration (heating-cooling-heating), simulation time and RAM needs. The results are given below:

Table 2.4 Cycle duration, simulation time and RAM needs versus mesh parameters

m_1, m_2, m_3, m_4, m_5	Cycle Duration	Simulation Time	RAM (Virtual)
1,10,2	328.11s	10min45s	4.23gb
1,10,3	329.98s	10min46s	4.88gb
1,10,4	335.88s	10min34s	5.42gb
1,15,2	335.13s	25min33s	8.14gb
1,15,3	334.75s	22min46s	9.10gb
1,15,4	334.60s	23min16s	10.09gb
1,20,2	341.38s	40min44s	13.71gb
1,20,3	341.37s	40min54s	15.18gb
1,20,4	341.36s	41min42s	16.63gb
1.5,10,2	322.54s	27min44s	14.38gb
1.5,10,3	322.54s	27min23s	14.24gb
1.5,10,4	322.47s	30min31s	14.55gb
1.5,15,2	329.92s	56min30s	18.55gb
1.5,15,3	329.92s	56min37s	19.02gb
1.5,15,4	329.63s	59min58s	19.18gb
1.5,20,2	323.55s	1h35min14s	24.33gb

Table 1.4 Cycle duration, simulation time and RAM needs versus mesh parameters (continued)

m_1, m_2, m_3, m_4, m_5	Cycle Duration	Simulation Time	RAM (Virtual)
1.5,20,3	323.55s	2h39min6s	23.51gb
1.5,20,4	323.55s	1h40min59s	24.59gb
2,10,2	316.79s	1h6min53s	19.91gb
2,10,3	316.79s	1h7min58s	19.37gb
2,10,4	311.96s	1h7min12s	18.22gb
2,15,2	323.33s	2h7min3s	27.94gb
2,15,3	xxx	Cancelled	xxx
2,15,4	xxx	Cancelled	xxx
2,20,4	323.35s	3h35min25s	31.71gb
2,25,4	323.50s	5h29min1s	49.95gb
2,30,4	323.42s	10h37min35s	74gb (62 physical)
1.5,20,4,2,1	323.55s	1h47min39s	39.05gb
1.5,20,4,2,3	323.65s	2h3min1s	23.75gb
1.5,20,4,4,1	323.52s	1h40min11s	24.77gb
1.5,20,4,4,3	329.75s	2h11min3s	26.98gb
1.5,20,4,4,5	323.57s	2h21min51s	25.11.gb
1.5,20,4,6,3	323.32s	2h10min41s	27.76gb
1.5,20,4,6,5	323.64s	2h15min22s	29.64gb

In first twenty-seven experiment of total thirty-four experiment, the cells are configured with respect to 3 parameters, m_4 is constant as 2 and m_5 is constant as 1. Then, these two parameters are examined after the first three parameters are optimized. Since it was seen that m_3 was an ineffective parameter, these were canceled to save time.

The axial temperature profiles are given in the below figures.

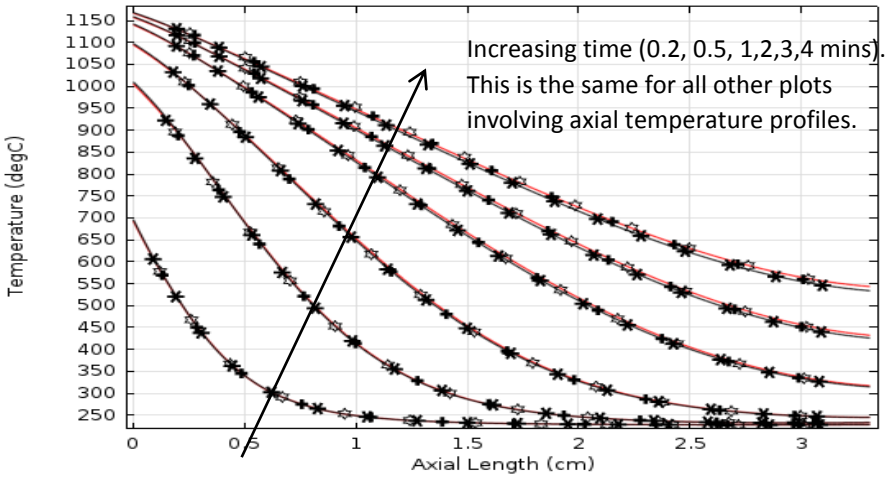

Figure 2.11 Axial temperature profiles with $\boldsymbol{m_4 = 2,\ m_5 = 1}$, red line: $\boldsymbol{m_1 = 1,\ m_2 = 10,\ m_3 = 4}$, black line: $\boldsymbol{m_1 = 1,\ m_2 = 15,\ m_3 = 4}$, asterisk: $\boldsymbol{m_1 = 1,\ m_2 = 20,\ m_3 = 4}$, plus: $\boldsymbol{m_1 = 1.5,\ m_2 = 20,\ m_3 = 4}$, star: $\boldsymbol{m_1 = 2,\ m_2 = 15,\ m_3 = 2}$

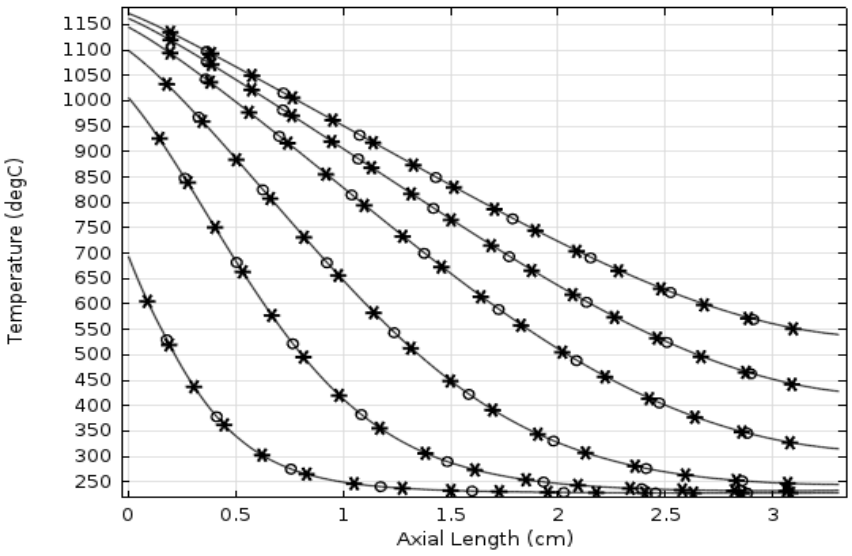

Figure 2.12 Axial temperature profiles with $\boldsymbol{m_1 = 2,\ m_3 = 4,\ m_4 = 2,\ m_5 = 1}$, black line: $\boldsymbol{m_2 = 20}$, circle: $\boldsymbol{m_2 = 25}$, asterisk: $\boldsymbol{m_2 = 30}$

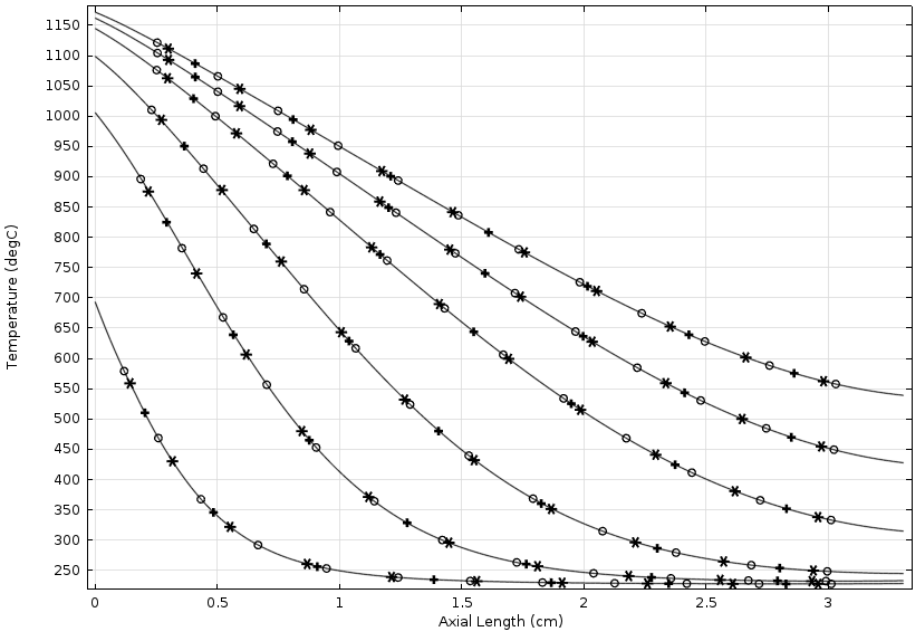

Figure 2.13 Axial temperature profiles with $\boldsymbol{m_1 = 1.5,\ m_2 = 20,\ m_3 = 4}$, black line: $\boldsymbol{m_4 = 2,\ m_5 = 1}$, circle: $\boldsymbol{m_4 = 4,\ m_5 = 1}$, asterisk: $\boldsymbol{m_4 = 2,\ m_5 = 3}$, plus: $\boldsymbol{m_4 = 4, m_5 = 3}$

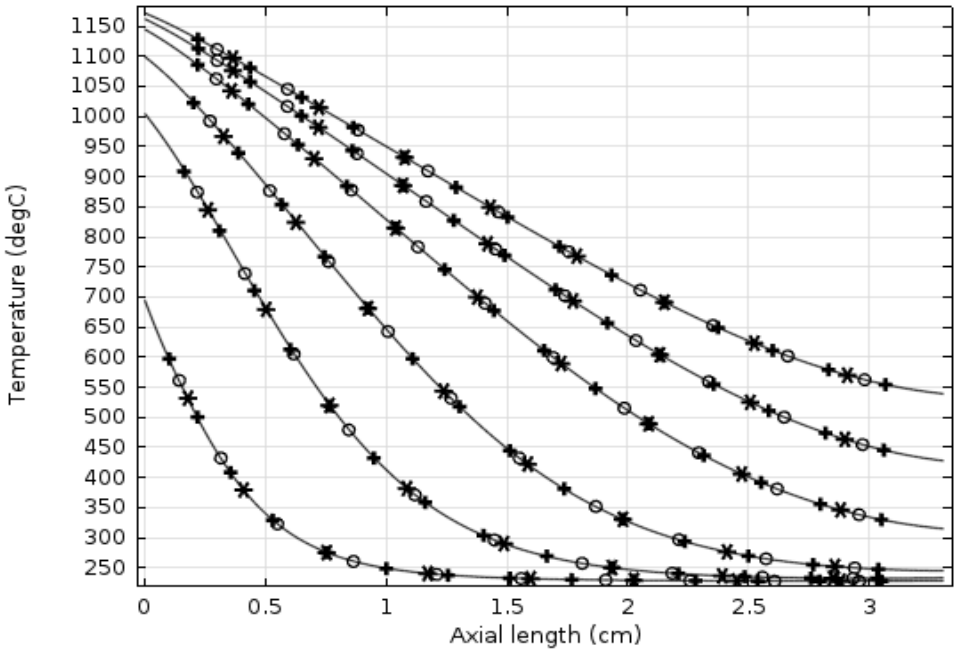

Figure 2.14 Axial temperature profiles with $\boldsymbol{m_1 = 1.5,\ m_2 = 20,\ m_3 = 4}$, black line: $\boldsymbol{m_4 = 4,\ m_5 = 3}$, asterisk: $\boldsymbol{m_4 = 4,\ m_5 = 5}$, circle: $\boldsymbol{m_4 = 6,\ m_5 = 3}$, plus: $\boldsymbol{m_4 = 6, m_5 = 5}$

The above figures show that the results are quite the same, but only in Figure 2.11, the red line with $m_2 = 10$ shows some slight deviation. However, one cannot clearly see the possible deviations at the front face of the monolith. To investigate the temperature at the channel entry, the time trends of temperature on a point (on the corner of the square) at the entrance are given in the following figures.

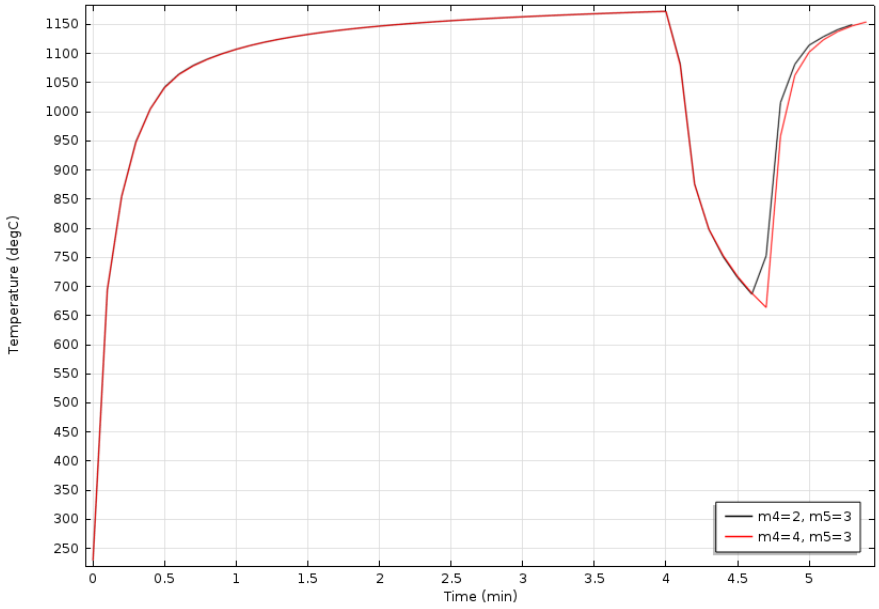

Figure 2.15 Temperature time trend at the channel entrance with $\boldsymbol{m_1 = 1.5, m_2 = 20, m_3 = 4}$

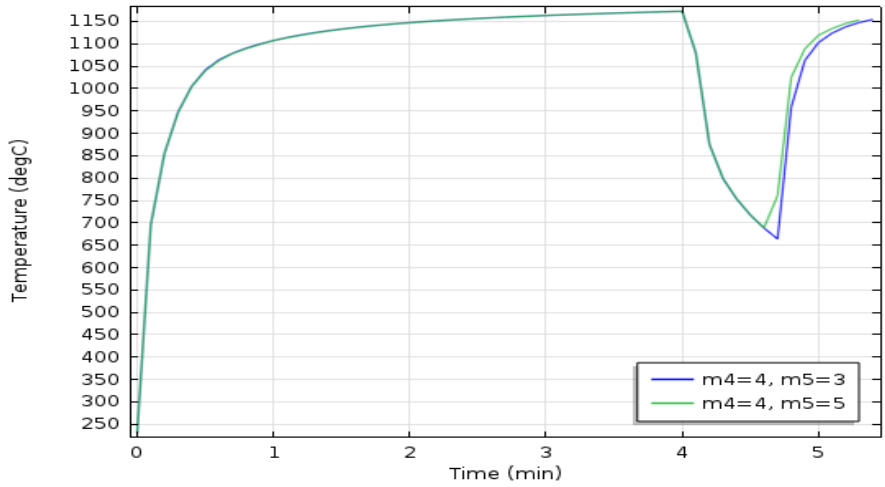

Figure 2.16 Temperature time trend at the channel entrance with $\boldsymbol{m_1 = 1.5, m_2 = 20, m_3 = 4}$

Note that the above figures show differences at the time when heat flux is re-switched on. The difference is of 0.1 min, which is the time step. Small changes in mesh can affect the global variables, which are the average temperature and uniformity here. Even a slight change seems to affect the time of transition into the next heating or cooling period.

Also, the aspect ratio of the meshes seems to be important. In Figures 2.15 and 2.16, the correct profile should be one with a straight temperature rise, not the one with the broken trend. But the broken line represents a situation where the body is more densely meshed. However, the aspect ratio of the denser mesh is larger due to the thin layers that are generated by division of the back lid into 5 pieces.

In the following figure, such effects are more severe, since there are large changes in the mesh. Differences in both the switch-off and switch-on times are seen here. In addition, the time trends are not smooth enough.

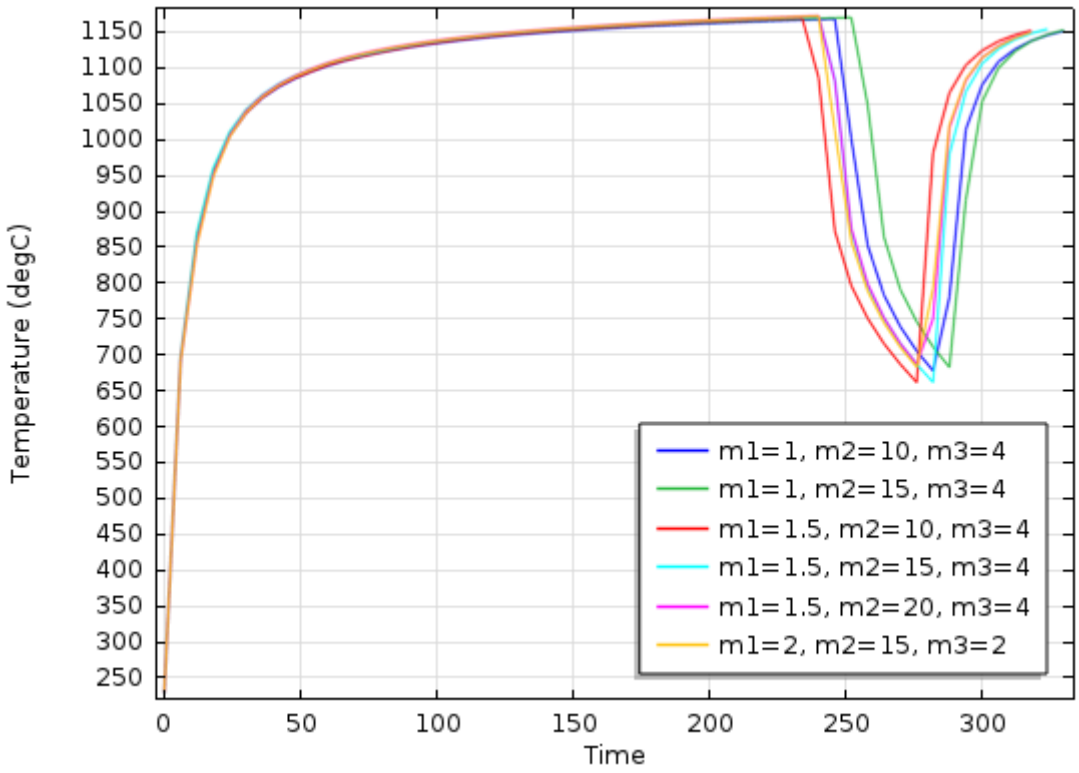

Figure 2.17 Temperature time trend at the channel entrance with $\boldsymbol{m_4 = 2,\ m_5 = 1}$. The orange line is the exact superposition of magenta and yellow

Although there are variables apart from the fineness of the mesh constructed, one may still evaluate a global variable (the total period duration here) and compare the relative importance of m's. This should be acceptable since the periods tend to a certain value as the mesh gets finer. The results are analyzed by using JMP program. It shows that m_1 and m_2 are the most important (decided upon the magnitude of the F ratio).

Below, the average temperatures of the monolith channel surfaces are given. The time trend of the average temperature is not smooth enough and it is not consistent. Also, the average does not reach the set oxidation and reduction temperatures close enough in some of the simulations.

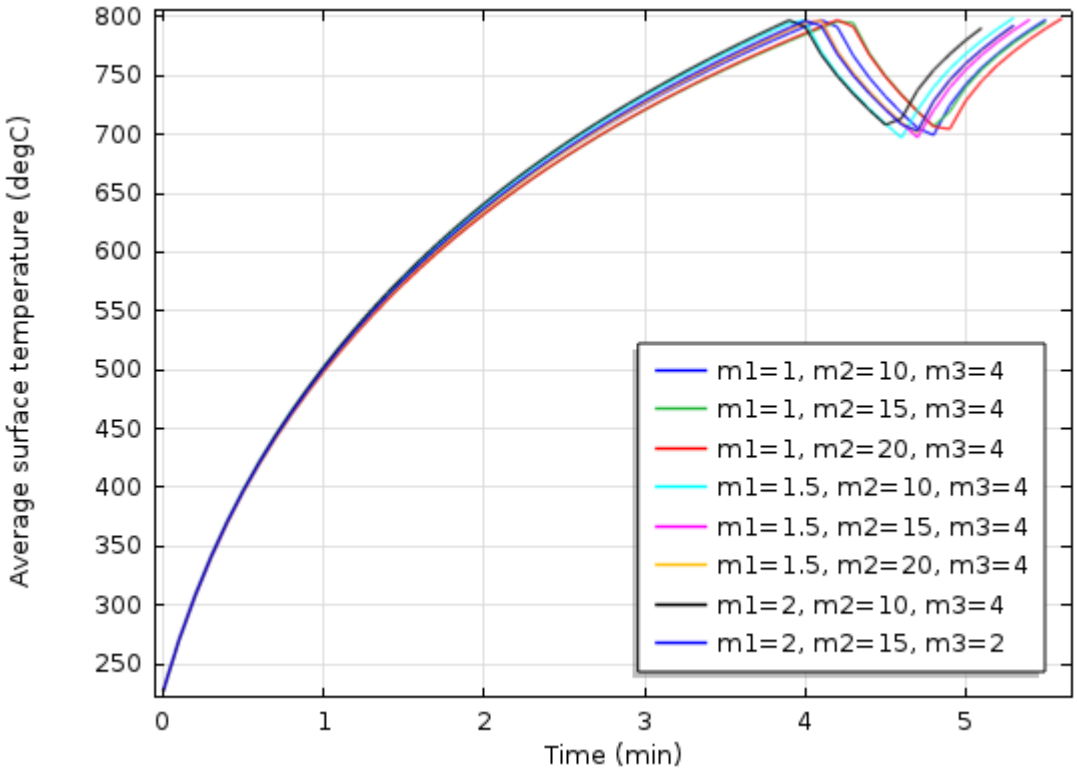

Figure 2.18 Average internal surface temperature with given mesh parameters, with $\boldsymbol{m_4 = 2,\ m_5 = 1}$

The RAM requirement sometimes depends on the previously ran study, thus it may be misleading in some comparisons. However, m_1 and m_2 are found as the most important parameters. The rest is relatively ineffective. Therefore, the optimum set is selected as:

$$m_1 = 1.5, \qquad m_2 = 20, \qquad m_3 = 4, \qquad m_4 = 2, \qquad m_5 = 1$$

These should also apply to hexagonal and triangular channels since the characteristic lengths and geometries are similar. A difference may appear in m_1, but $m_1 = 2$ probably to solve the problem accurately enough. Each simulation is expected to be completed in approximately 2 hours, with ~25 GB RAM.

2.2.5 Solver Optimization

To improve the accuracy, smoothness and to make the solver more sensitive to the switching between heating and cooling periods, some additional manipulations have been done.

The time derivatives are discretized by using backward differentiation formulas (BDF). These are proven to be very stable, but not always accurate. Increasing the order of the minimum BDF order (for example at a point: $\partial T/\partial t = \left(T^{i} - T^{i-1}\right)/\Delta t$ is first order discretization, where "i" denotes the time step) will make the time trends more smooth and realistic, but inherently less stable. However, since the problem is not very nonlinear (not a multiphase or turbulent flow case) stability should not be of any concern. Therefore the minimum (and also the maximum) BDF is order (default is one) is selected as 2. Higher orders are omitted due to the additional computational intensity.

Even though one enters the time step and the end time to the program in a time-dependent simulation; by default, COMSOL takes time steps freely. That is, depending on the physics and the time gradients of the problem, the program can make the time steps larger or smaller than it is actually specified by the user. If it takes larger steps, then it uses interpolation for the desired time data. If it takes smaller steps, it uses the corresponding solutions to obtain the solution at the desired

time and then discards them. This freedom can be limited to a certain degree. In our case, there are implicit events, like the heating-cooling switch-on/shut-down moments. If the solver takes the time steps freely, it may actually miss (actually, react late to) the global constraints corresponding to these events. Therefore, in these simulations, the time steps are taken by program as "intermediately" free. This means, after every two time iterations, the program must use the time step prescribed by the user. It can take larger time steps in between, and decrease it anytime. The decrease in time steps is entirely decided by the program and it cannot be manipulated. It was seen that the maximum time step, should be close to the specified default time step. It was decreased to 0.1 from 0.5 minutes.

With the above maximum solver time step equal to 0.1 min, the overall time step must be decreased. It was set to 0.01 min, instead of 0.1 min.

The event tolerance makes the solver being precautious when an implicit event is approaching; then the solver decreases the time steps to a certain degree. It was reduced to 0.0001 from its default value of 0.01.

The temperature and radiosity, which are the dependent variables in this study, are scaled to a certain extent to make the matrix computations slightly faster. The principle is to divide the dependent variables to their expected magnitude. For temperature, it is taken as $\overline{T} = (T_{red} + T_{ox})/2$, for radiosity it is taken as $\bar{J} = q_s/2$, where q_s is the solar flux.

The overall improvements did not boost up the simulation times significantly. For the optimum set given above $(m_1 = 1.5,\ m_2 = 20,\ m_3 = 4,\ m_4 = 2,\ m_5 = 1)$ the simulation time is increased by 20 minutes, but the overall physical consistency and the smoothness of the simulations seems to be acceptable. The time trend of the surface average temperature and the temperature on an inlet point is given below. One can compare the following figures with the magenta and yellow colored plots of Figures 2.17 and 2.18 respectively.

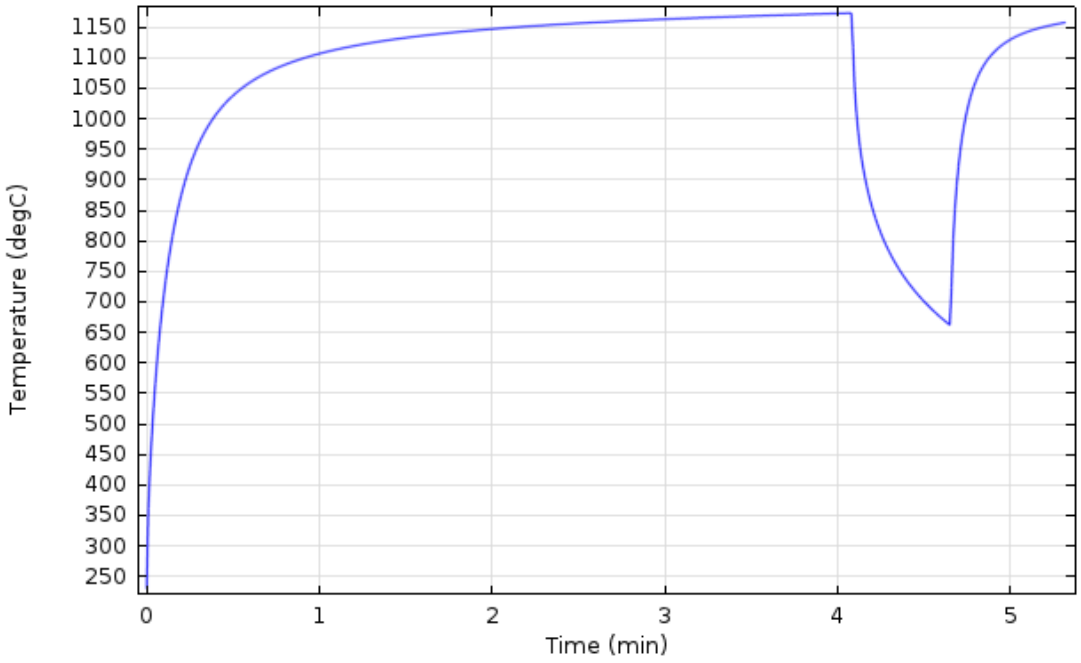

Figure 2.19 Temperature time trend at the channel entrance with optimum mesh and solver parameters

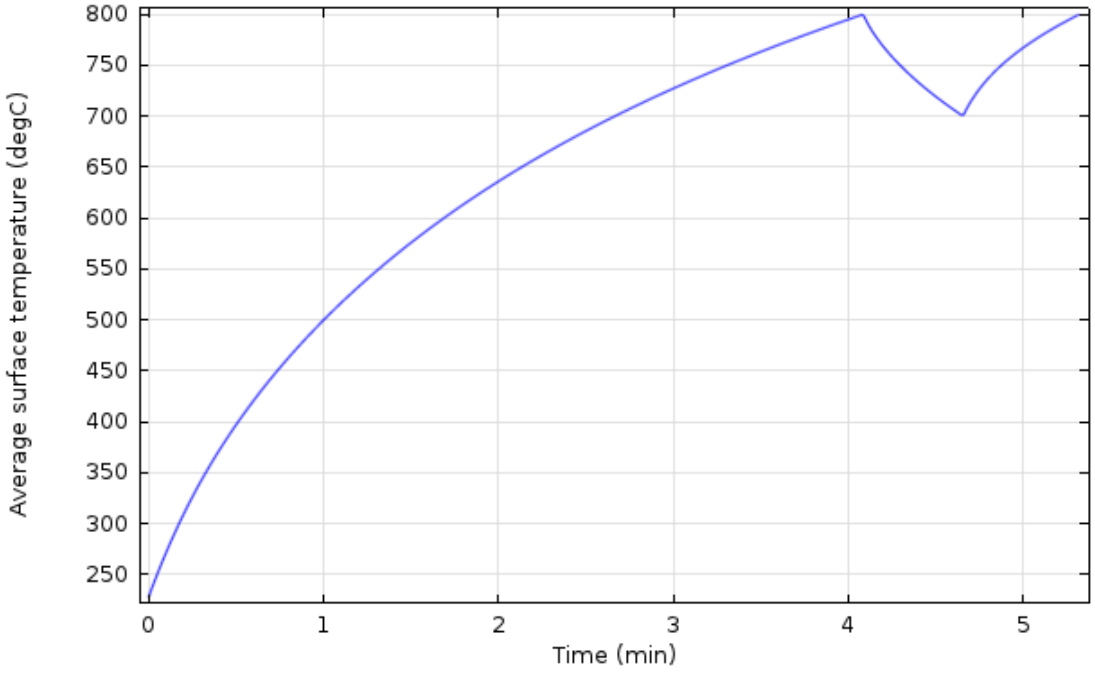

Figure 2.20 Average internal surface temperature with optimum mesh and solver parameters

2.3 Kinetic Model Approach and Development

Aim of this model is to calculate desorbed oxygen and adsorbed oxygen amounts in lateral surface of reactor channels by using MATLAB program thanks to temperature profiles with respect to time which are obtained from thermal analysis by using COMSOL program.

2.3.1 Configuration of Kinetic Model

Methodology of model configuration is to use experimental temperature data and Arrhenius fit equation in order to obtain results which are close to experimental results. Also, model results should have less than %5 error. There are some common models for reaction kinetic expression which includes Arrhenius fit equation. As a result of literature survey, four common models are determined and considered for kinetic model [40].

$$\frac{da}{dt} = A \times \exp(^{-E_a}/_{RT}) 3a^{2/3} \qquad \text{Power Law (P3)} \qquad (2.13)$$

$$\frac{da}{dt} = A \times \exp(^{-E_a}/_{RT}) \qquad \text{Zeroth Order} \qquad (2.14)$$

$$\frac{da}{dt} = A \times \exp(^{-E_a}/_{RT}) \times (1 - a) \qquad \text{First Order} \qquad (2.15)$$

$$\frac{da}{dt} = A \times \exp(^{-E_a}/_{RT}) \times (\frac{1}{2a}) \qquad \text{1-D Diffusion} \qquad (2.16)$$

Basing on fit equation calculations which is made on experimental data by using EXCEL program, Power Law (P3) is convenient for desorption model and Zeroth Order is convenient for adsorption model. These models are updated regularly by using new experimental data.

$$\frac{da}{dt} = 343 \times \exp(\frac{-8761.5}{T(t)}) \times 3a^{\frac{2}{3}} \qquad \text{Desorption Model} \qquad (2.17)$$

$$\frac{da}{dt} = 37.4 \times \exp(\frac{-4845.9}{T(t)}) \qquad \text{Adsorption Model} \qquad (2.18)$$

These model equations related on reaction kinetic are differential equation and integration calculation is needed to solve these equations. These calculations are made by using numeric integration method in MATLAB program. Also, analytic integration with linear approximation method can be used instead of numeric integration method.

2.3.1.1 Desorption Model

Configuration of desorption model has certain significant steps. Firstly, conversion of temperature data unit from Kelvin to Celsius is needed for calculations (Figure 2.21).

```
Kelvin=273*ones(size(T));
TK=T+Kelvin;
    for c=2:length(Data);
            o=length(Data);
            a=time(1,c)-time(1,1);
            b=time(1,c)-time(1,c-1);
            heatingrate=(TK(1,c)-TK(1,c-1))./b;
            Temp(c)=heatingrate.*(time(1,c)-time(1,c-1))+TK(1,c-1);
            fun= @(x) exp(-8761.5./(heatingrate.*(x-time(1,c-1))+TK(1,c-1)));
            integ(c)=quad(fun,time(1,c-1),time(1,c));
            son=(integ(c)*343)+(bizimcon(c-1))^(1/3);
            bizimcon(c)=son^3;
            diff(c)=TK(c)-Temp(c);
            error(c)=(bizimcon(c)-Data(c,3))/Data(c,3);
            if bizimcon(c)>0.1
            break
            end
```

Figure 2.21 Conversion procedure of temperature data unit

Second step is to provide evaluation of all data in MATLAB by using for loop of program. For example, heating rate is difference of temperature data between two time steps. Subtraction of consecutive temperature data is divided by the time step to obtain heating rate value (Figure 2.22).

```
Kelvin=273*ones(size(T));
TK=T+Kelvin;
    for c=2:length(Data);
            o=length(Data);
            a=time(1,c)-time(1,1);
            b=time(1,c)-time(1,c-1);
            heatingrate=(TK(1,c)-TK(1,c-1))./b;
            Temp(c)=heatingrate.*(time(1,c)-time(1,c-1))+TK(1,c-1);
            fun= @(x) exp(-8761.5./(heatingrate.*(x-time(1,c-1))+TK(1,c-1)));
            integ(c)=quad(fun,time(1,c-1),time(1,c));
            son=(integ(c)*343)+(bizimcon(c-1))^(1/3);
            bizimcon(c)=son^3;
            diff(c)=TK(c)-Temp(c);
            error(c)=(bizimcon(c)-Data(c,3))/Data(c,3);
            if bizimcon(c)>0.1
            break
            end
```

Figure 2.22 Definition of heating rate

Temperature function is defined to combine fit equation and temperature data. Then, this function is taken integral with respect to time. In this part, there are two certain methods for this calculation. First one is to calculate conversion values at every small step by evaluating temperature increasing at every small time step. Second step is to calculate total conversion values until evaluating time from beginning time by examining temperature increasing. First method is selected because first method has less error than second method (Figure 2.23).

```
Kelvin=273*ones(size(T));
TK=T+Kelvin;
    for c=2:length(Data);
            o=length(Data);
            a=time(1,c)-time(1,1);
            b=time(1,c)-time(1,c-1);
            heatingrate=(TK(1,c)-TK(1,c-1))./b;
            Temp(c)=heatingrate.*(time(1,c)-time(1,c-1))+TK(1,c-1);
            fun= @(x) exp(-8761.5./(heatingrate.*(x-time(1,c-1))+TK(1,c-1)));
            integ(c)=quad(fun,time(1,c-1),time(1,c));
            son=(integ(c)*343)+(bizimcon(c-1))^(1/3);
            bizimcon(c)=son^3;
            diff(c)=TK(c)-Temp(c);
            error(c)=(bizimcon(c)-Data(c,3))/Data(c,3);
            if bizimcon(c)>0.1
            break
            end
```

Figure 2.23 Definition of temperature function, integral operations and error calculations

%1. Therefore, model is compatible with kinetic expression. So that, it can be applicable for combination of COMSOL thermal model results.

2.3.2 Combination of Kinetic Model and COMSOL Thermal Results

After configuration of model is done, combination of thermal profile in COMSOL and kinetic model is aimed. Thus, conversion value of every small mesh in COMSOL simulation works can be calculated. In addition, import data operations from COMSOL to MATLAB program are added to kinetic model code. It is shown below with Figure 2.33.

```
16 - for id=1:length(files)
17
18 - if strfind(files(id).name,'.txt') >0
19 -    file2read = [path '\' files(id).name];
20 -    runID=files(id).name(1:strfind(files(id).name,'.txt')-1);
21 -    excel2write= [path '\' num2str(runID) '.xlsx'];
22      %import data operations
23 -    [data,delimiter,header] = importdata(file2read,' ',9);
24 -    times = char(data.textdata(9));
25 -    data = (data.data);
26 -          start = strfind(times,'=') + 1;
27 -          stop = strfind(times,'T') - 2;
28 -             stop(1) = [];
29 -             stop = [stop length(times)];
30 -             time = [];
31 -             for j = 1:length(stop)
32 -                time(j) = str2double(times(start(j):stop(j)));
33 -             end
34 -             time = time';
35 -          temp = size(data);
36 -          T = data(:,4:temp(2));
37 -          Kelvin=273*ones(size(T));
38 -          TK=T+Kelvin;
39 -          for o=2:length(time)
40 -             if TK(1,o)-TK(1,o-1)<0
41 -                break
42 -             end
43 -          end
44          % 'o-1' is stopping time of first heating step
45 -          for p=o:length(time)
46 -             if TK(1,p)-TK(1,p-1)>0
47 -                break
```

Figure 2.33 Import data operations

After that, COMSOL data are evaluated to determine different time processes. There are three time processes in COMSOL models. First heating step, first cooling step

and second heating step are system time processes. Part of code which is used for determination of different time steps is shown below in Figure 2.34.

```
T = data(:,4:temp(2));
Kelvin=273*ones(size(T));
TK=T+Kelvin;
for o=2:length(time)
    if TK(1,o)-TK(1,o-1)<0
        break
    end
end
% 'o-1' is stopping time of first heating step
for p=o:length(time)
    if TK(1,p)-TK(1,p-1)>0
        break
    end
end
% 'p-1' is stopping time of first cooling step
```

Figure 2.34 Determination of starting and stopping time processes

Rest part of code is not changed and it is integrated with desorption and adsorption codes. Graph which is shown that average kinetic conversion amount of desorption reaction (oxygen releasing reaction) with respect to first heating time, is taken place in the below Figure 2.35

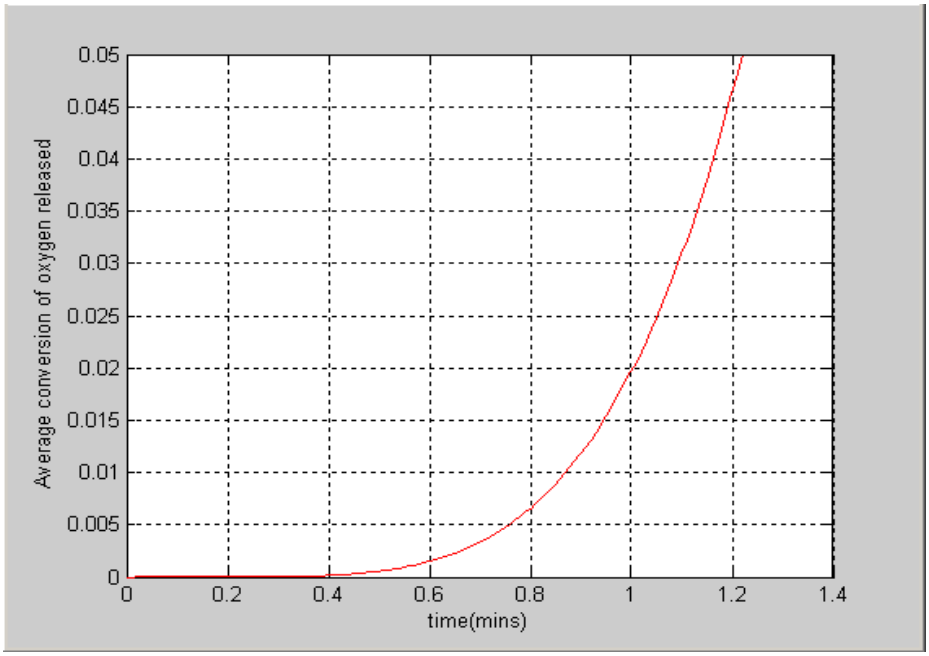

Figure 2.35 Average kinetic conversion increasing amount graph of oxygen releasing reaction during heating time

2.4 Combined Transport Model Configuration

Important assumption was made when thermal model was configured for reactor system. This assumption was that temperature profiles of all reactor channels are not influenced by kinetic, momentum phenomena and mass transfer through diffusion or convection. The most significant heat transfer concepts are radiation and conduction for reactor channels. Thermal-momentum-mass (including kinetic) combined transport model was configured for some purposes. These purposes are validation of this assumption, examination of all physical concepts in single channel and projection of hydrogen production reaction conversion values.

2.4.1 Variables

There are several variables for transport model in single channel. They are tabulated in Table 2.6 below.

Table 2.6 Variables of combined transport model in single channel

Name	Expression	Unit	Description
CPSI	20		Channel number
d	1	[mm]	Wall thickness
sun	1000	[W/m^2]	One solar flux
flux	250*sun	[W/m^2]	Total solar flux
radius	((CPSI/0.0254^2)^(-1)/pi)^(1/2)-d	[m]	Radius
cMeO20	50	[mole/m^3]	Initial concentration of catalyst
v0	0.1	[m/s]	Initial velocity
hea	20	[W/(m^2*K)]	Convective heat transfer coefficient

Table 2.6 Variables of combined transport model in single channel (continued)

Name	Expression	Unit	Description
a	4.95		Aspect ratio
L	a*2*radius	[m]	Length
heat flux	flux*pi*radius^2	[W/m^2]	General heat flux
kdes	1.00E-02	[1/s]	Desorption rate constant
NMeO20	molMeO20/(2*pi*radius*L)	[mole/m^2]	Molar flux of catalyst
molMeO20	0.001	[mole]	Mole of catalyst
massMeO20	100 [g/mole]*molMeO20	[g]	Mass of catalyst

2.4.2 Geometry

Three dimensional studies need high RAM and take a long time. Both three dimensional system and two dimensional axisymmetric system are configured to study two dimensional system instead of three dimensional system by using symmetry because of these reasons. Thus, one cycle is selected from desorption and absorption cycles and same physics and materials are defined in both cycles. After running, their results are compared together to justify dimension assumption. Models geometries are shown below.

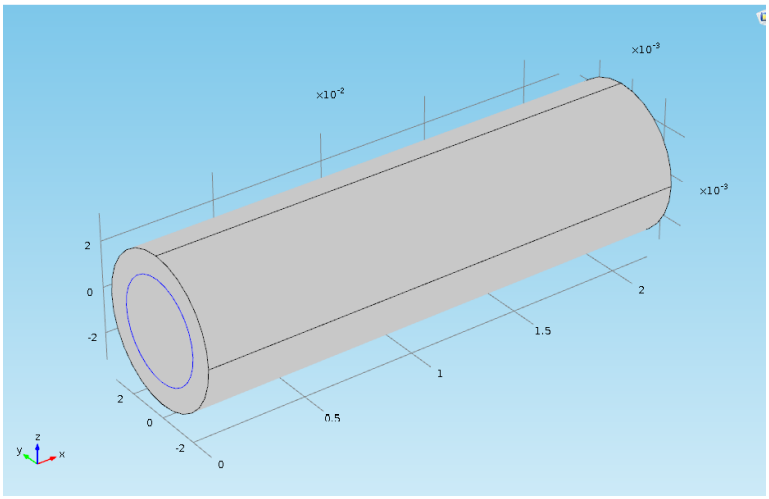

Figure 2.36 Three dimensional model geometry

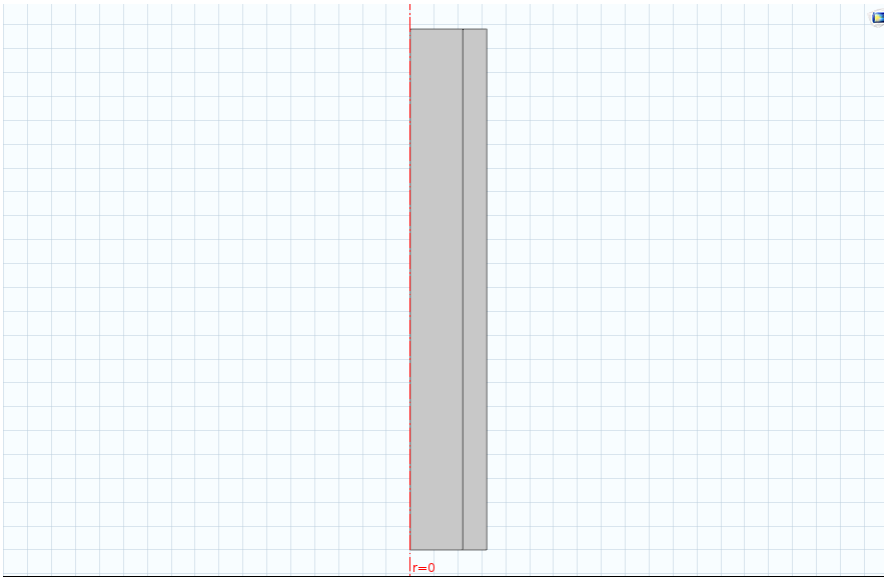

Figure 2.37 Axisymmetric model geometry (two dimensional)

2.4.3 Physics

Desorption cycle is selected for validation of symmetry assumption and physical phenomena is defined for this reaction. All simulations are operated unsteadily (time-dependent) except this validation simulation. Validation simulation of symmetry assumption is operated steadily.

Reduction (desorption) reaction:

$$MeO_{\nu} \rightarrow MeO_{\nu-1} + \frac{1}{2}O_2 \tag{2.19}$$

Oxidation (adsorption) reaction:

$$MeO_{\nu-1} + H_2O \rightleftharpoons MeO_{\nu} + H_2 \tag{2.20}$$

Rate equations for these reactions are indicated below. These equations is determined through kinetic data which are obtained from results of laboratory experiments

$$\mathcal{R} = -k_{des} N_{MeO_2} \tag{2.21}$$

$$\mathcal{R} = -k_{gen} \exp\left(-\frac{4845.9}{T}\right) N_{MeO} \tag{2.22}$$

Unit of reaction rate is mole/m^2s and it equals to diffusion flux. Fluids in reactor system are compressible. Navier-Stokes and continuity equations are used for computation on COMSOL program. Reactions at the metal oxide coating of reactor channels are reduction and oxidation reactions. Reduction reaction is irreversible reaction and oxidation reaction is reversible reaction. Increasing thickness of metal oxide coating can be neglected. Diffusion inside coating can be neglected. Reaction rate equations are elementary, oxygen gas in reduction part and hydrogen gas, steam in oxidation part are assumed as ideal gas. Fluid Flow is assumed as laminar flow.

Navier-Stokes and continuity equation are given below:

$$\rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot \left[-P\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right] + \mathbf{F}_b \tag{2.23}$$

$$\nabla \cdot (\rho\mathbf{u}) = 0 \tag{2.24}$$

u represents velocity vector, P is pressure, $\mathbf{F}_b$ is body force and **I** is unit matrix.

Operational conditions are so significant for selection of equations including convection and diffusion phenomena. In this step, only steam is used in hydrogen production reaction systems. However, one noble gas like argon, helium is used as carrier to clean oxygen and hydrogen gases in reactor channels and prevent mixing of these gases. In this system, helium gas is used as carrier gas. Thus, mole fractions of main three gases are decreased and system becomes dilute system. Fick's law is convenient for dilute systems. Fick's law equation is shown below:

Fick's Law Equation:

$$\frac{\partial C_i}{\partial t} + \nabla \cdot (-D_i \nabla C_i) + \mathbf{u} \cdot \nabla C_i = \mathcal{R}_i\,,\ \mathbf{N}_i = -D_i \nabla C_i + \mathbf{u} C_i \tag{2.25}$$

In the above equation, C represents concentration, D symbolizes diffusion coefficient, u is velocity, t is time, R is reaction rate equation and N symbolizes mass flux.

In heat transfer phenomena in single channel, transfer equations and boundary conditions are defined separately for solid and fluid materials. Even though radiative calculations of COMSOL are used for the other three dimensional models, view factor formulas are used in this model. Mesh problems of front lid is the reason of this selection. So that solar flux can heat directly lateral surfaces of channels. Forced convection of heat transfer is defined between fluid and channels.

Heat transfer equations in model are shown below:

General heat transfer equation for solid and steady-state system:

$$0 = \bar{k}_1 \boldsymbol{\nabla}^2 T_1 \tag{2.26}$$

Radiation equation from channel surfaces to ambient:

$$-\mathbf{n} \cdot \left(-\bar{k}_1 \nabla T_1\right) = \varepsilon\sigma\left({T_{amb}}^4 - {T_1}^4\right) \tag{2.27}$$

Surface to surface radiation equations:

$$-\mathbf{n} \cdot \left(-\bar{k}_1 \nabla T_1\right) = \varepsilon\left(G - \sigma {T_1}^4\right) \tag{2.28}$$

$$J = J_0 \qquad \varepsilon(T) = \mathbf{n^2}\sigma T^4 \tag{2.29}$$

Heat flux equation for monolith front face and lateral surfaces of reactor channels:

$$-\mathbf{n} \cdot \left(-\bar{k}_1 \nabla T_1\right) = q_0 \tag{2.30}$$

Forced heat convection equation between channel surfaces and fluids:

$$-\mathbf{n} \cdot \left(-\bar{k}_1 \nabla T_1\right) = hea(T_1 - T_2) \tag{2.31}$$

General heat transfer equation for fluids and steady-state system:

$$\bar{\rho}_2 \overline{C_{p_2}} \mathbf{u}.\boldsymbol{\nabla} T_2 = \bar{k}_2 \boldsymbol{\nabla}^2 T_2 \tag{2.32}$$

Forced heat convection equation is defined for both solid and fluid module. In the above equations, k represents thermal conductivity, T_1 is solid temperature, T_2 is fluid temperature, T_{amb} is ambient temperature, ε symbolizes emissivity, G-J-q_0 are heat fluxes and hea symbolizes forced convective heat transfer coefficient.

Profile results of these two models are shown below.

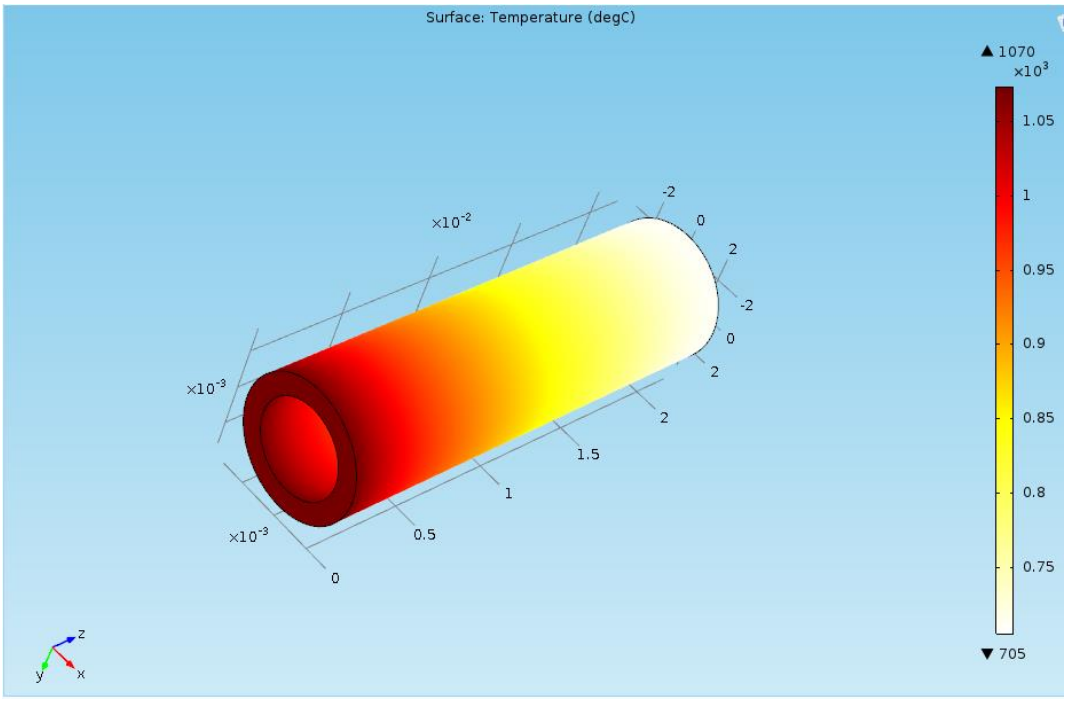

Figure 2.38 Temperature profile of axisymmetric model

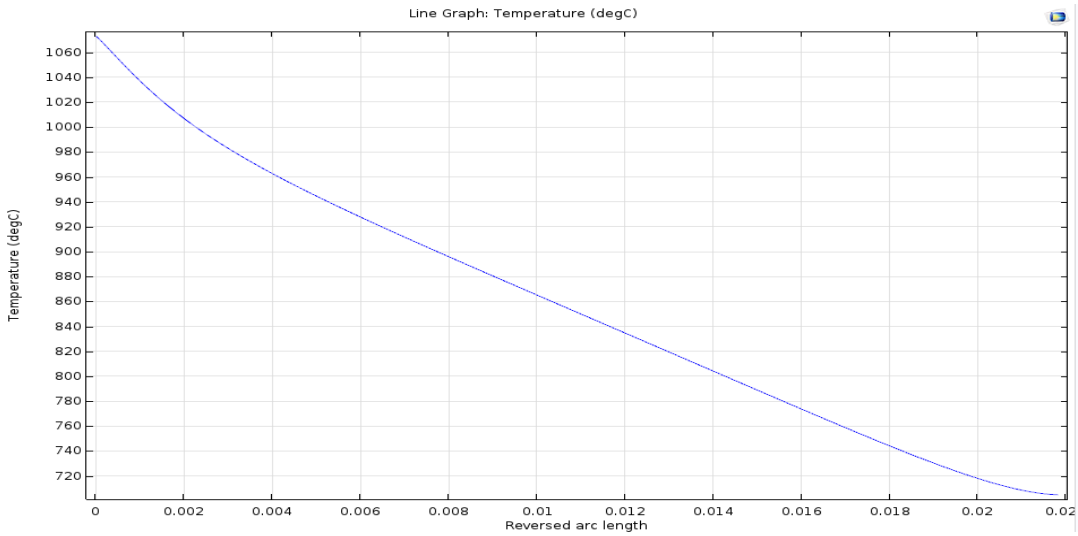

Figure 2.39 Temperature profile of axisymmetric model throughout channel

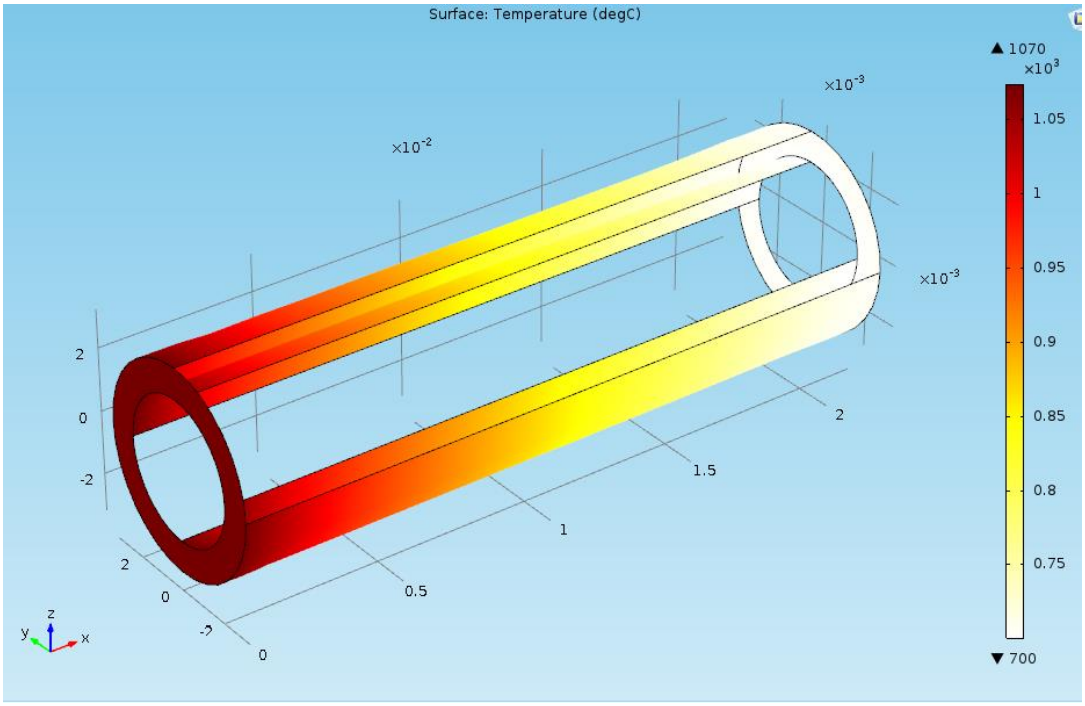

Figure 2.40 Temperature profile of three dimensional model

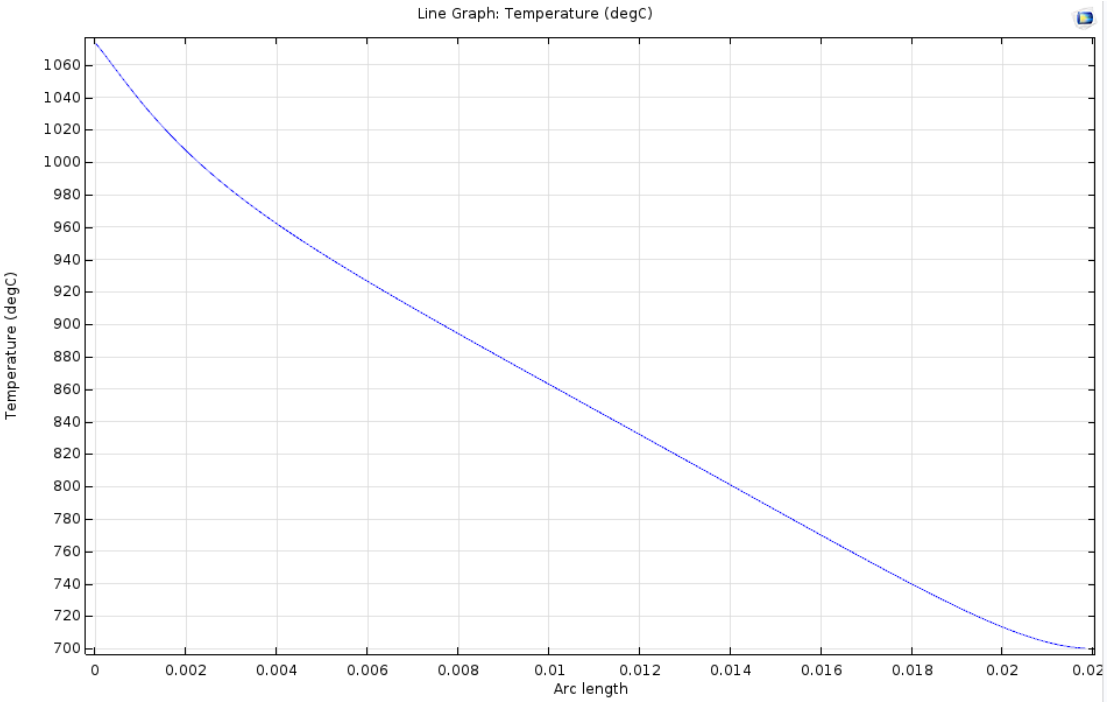

Figure 2.41 Temperature profile of three dimensional model throughout channel

According to both figures and graphs, three dimensional model and axisymmetric model have similar or almost same temperature profiles. Therefore, symmetry assumption is justified and axisymmetric two dimensional model is used for all model configurations.

2.4.4 Mesh and Solver Selection

In this combined transport model, mesh is not configured as user defined mesh. It is configured as physics-controlled mesh and its quality is fine. The time derivatives are discretized by using backward differentiation formulas (BDF) like thermal model. Also, the time steps are taken by program as "intermediately" free like thermal model. Two direct solvers which are defined PARDISO and nested dissection preordering are used for computational. They are segregated. One of the solvers computes thermal and mass transfer models, the other solver computes momentum transfer model. Maximum solver time step is selected as 0.1 min. The event tolerance makes the solver being precautious when an implicit event is approaching; then the solver decreases the time steps to a certain degree. In this study, implicit event is not defined. Hence, the event tolerance is selected as 0.001. Only radiosity, which is the dependent variables in this study, is scaled to a certain extent to make the matrix computations slightly faster. The principle is to divide the dependent variables to their expected magnitude. For radiosity it is taken as $\bar{J} = q_s/2$, where q_s is the solar flux.

CHAPTER 3

RESULTS AND DISCUSSION

3.1 Combined Transport Model Results

For both two cycles (reduction-oxidation), reactor model, which contains mass transfer, momentum transfer and heat transfer, is configured in axisymmetric plane and this model is compared with three dimensional model which includes only thermal model in terms of temperature profiles. First comparison is reduction reaction (desorption), second comparison is oxidation reaction (adsorption).

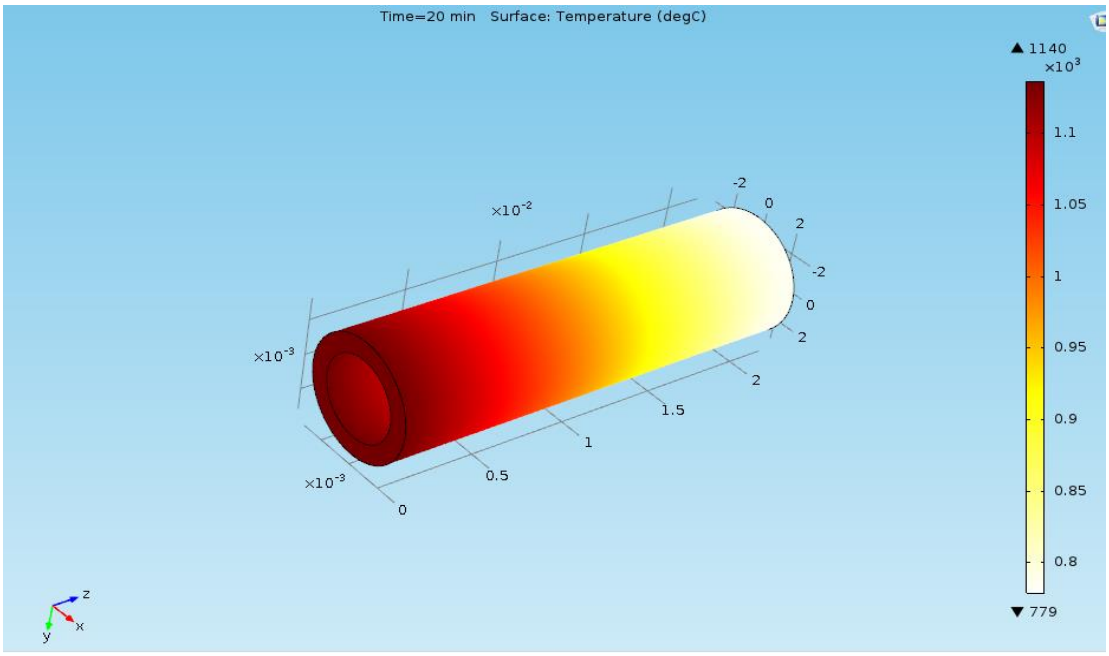

Figure 3.1 Temperature profiles results of reduction reaction model for mass (kinetic)-thermal-momentum phenomena

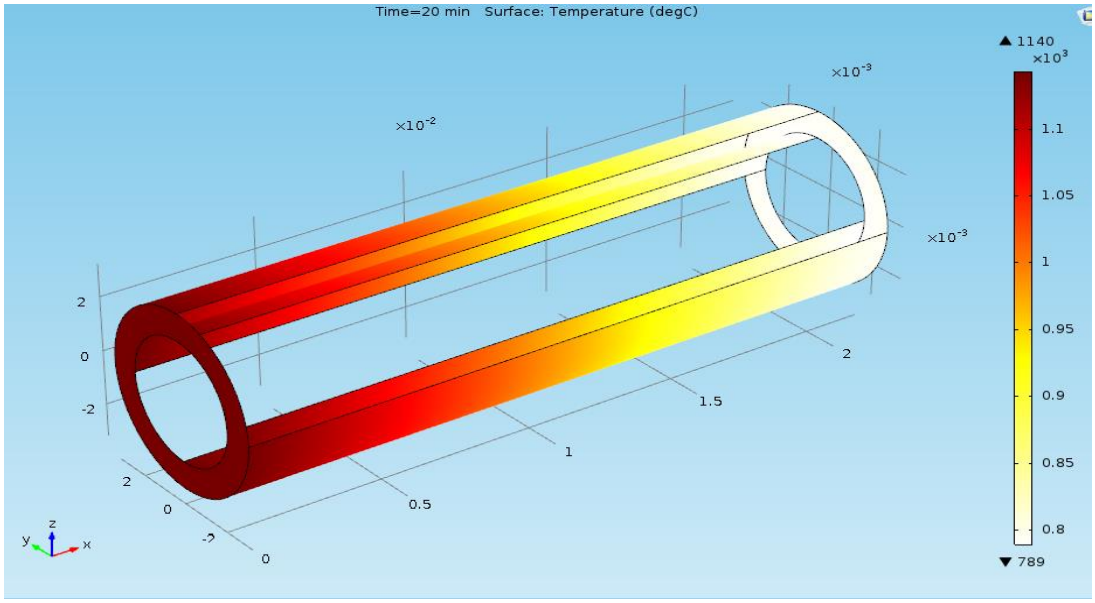

Figure 3.2 Temperature profiles results of reduction reaction model for just thermal model

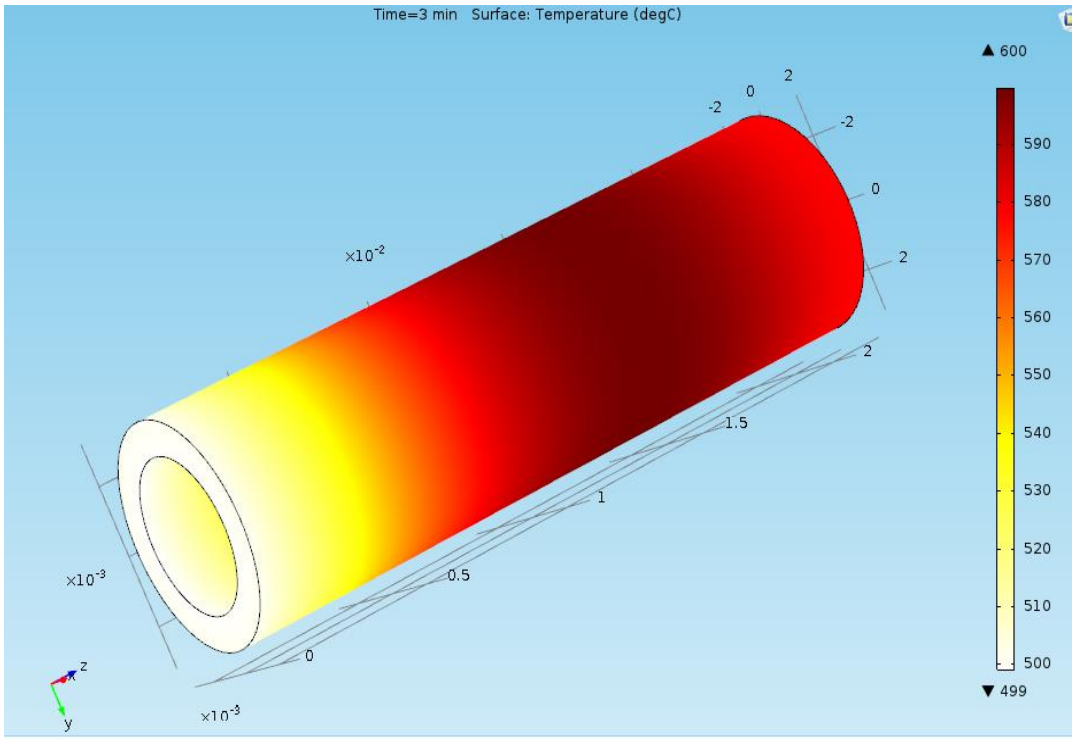

Figure 3.3 Temperature profiles results of oxidation reaction model for mass (kinetic)-thermal-momentum phenomena

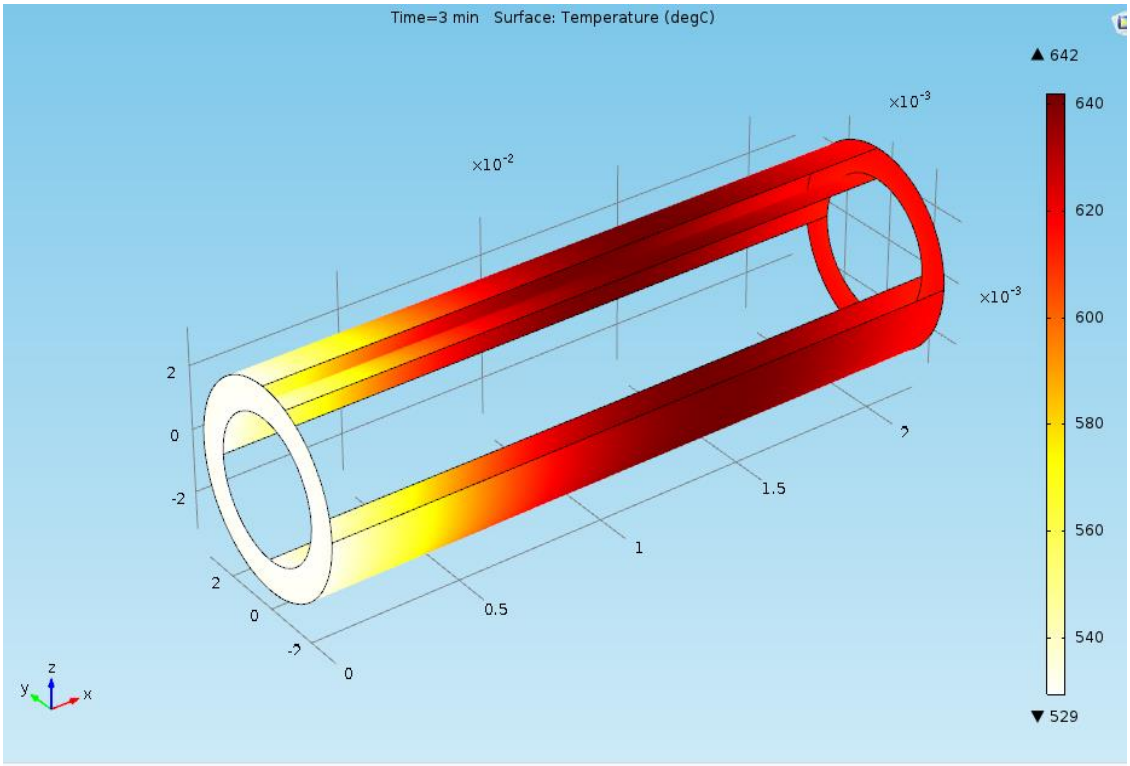

Figure 3.4 Temperature profiles results of oxidation reaction model for just thermal model

As it is shown in the above figures, very low temperature difference is occurred in reduction reaction due to low flow rates of helium and produced oxygen gases and no steam flow inside channel. Maximum temperature difference is observed in backside of channel and this value is obtained as between 0-10 ºC. On the other hand, oxidation part has slightly higher temperature difference than reduction part because of steam flow inside channel. Actually, this temperature difference is also acceptable for model assumptions. Temperature profiles in reactor model are between 500-600 ºC and temperature difference between combined transport model and only thermal model in oxidation part is 30-40 ºC. These values have less than %10 error and impacts of kinetic, mass transfer (diffusion and convection) and momentum transfer (velocity and pressure) on temperature profiles of reactor model can be neglected.

Finally, oxygen and hydrogen production amounts of two reactions, hydrogen conversion of oxidation reaction and molar flux values of metal oxide coating on

channel surfaces are examined. In reactor model system, unit of metal oxide coating amount is taken as mole/m^2 in terms of mole per surface area.

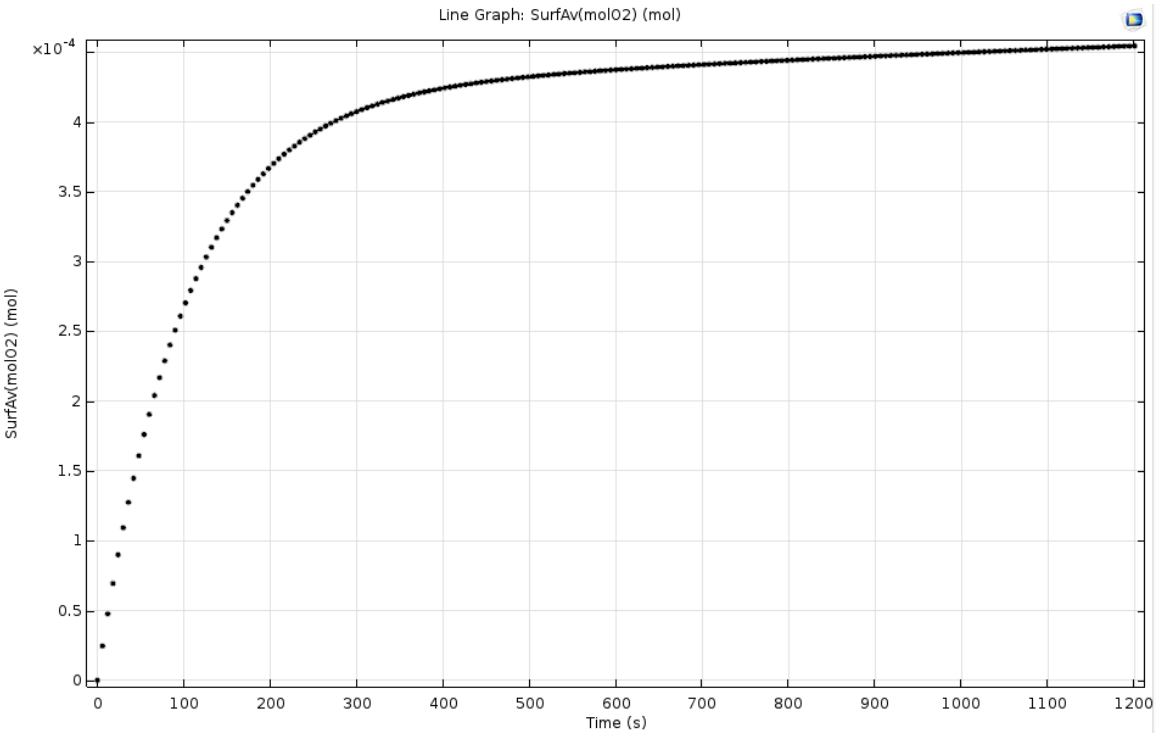

Figure 3.5 Mole releasing amount of oxygen gas from metal oxide coating to channel in reduction reaction part with respect to time

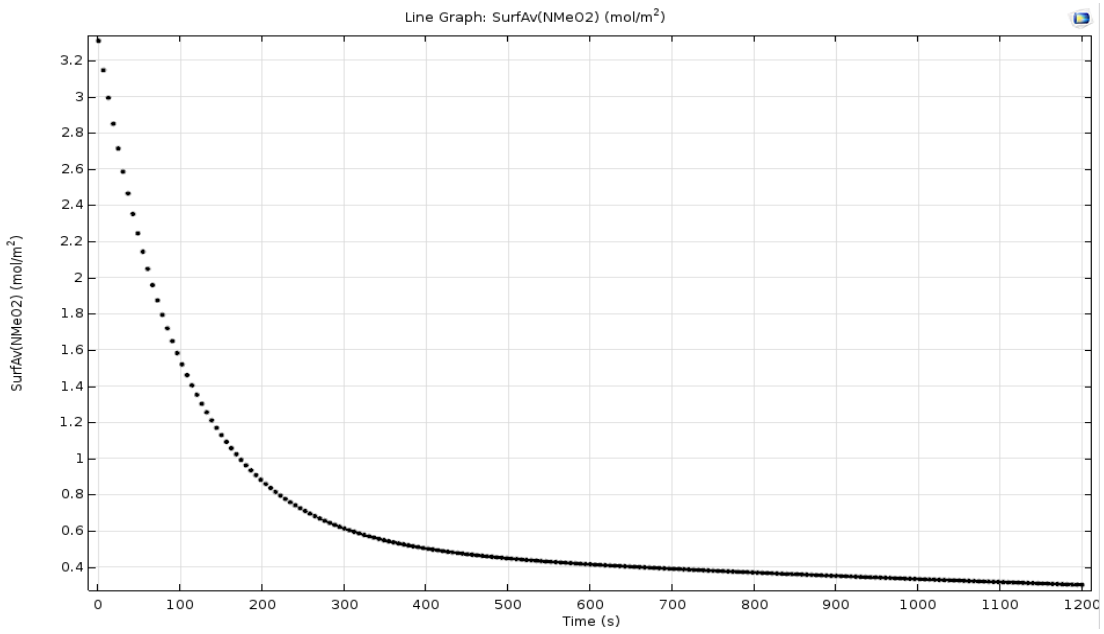

Figure 3.6 Molar flux change amount of metal oxide coating in reduction reaction part with respect to time

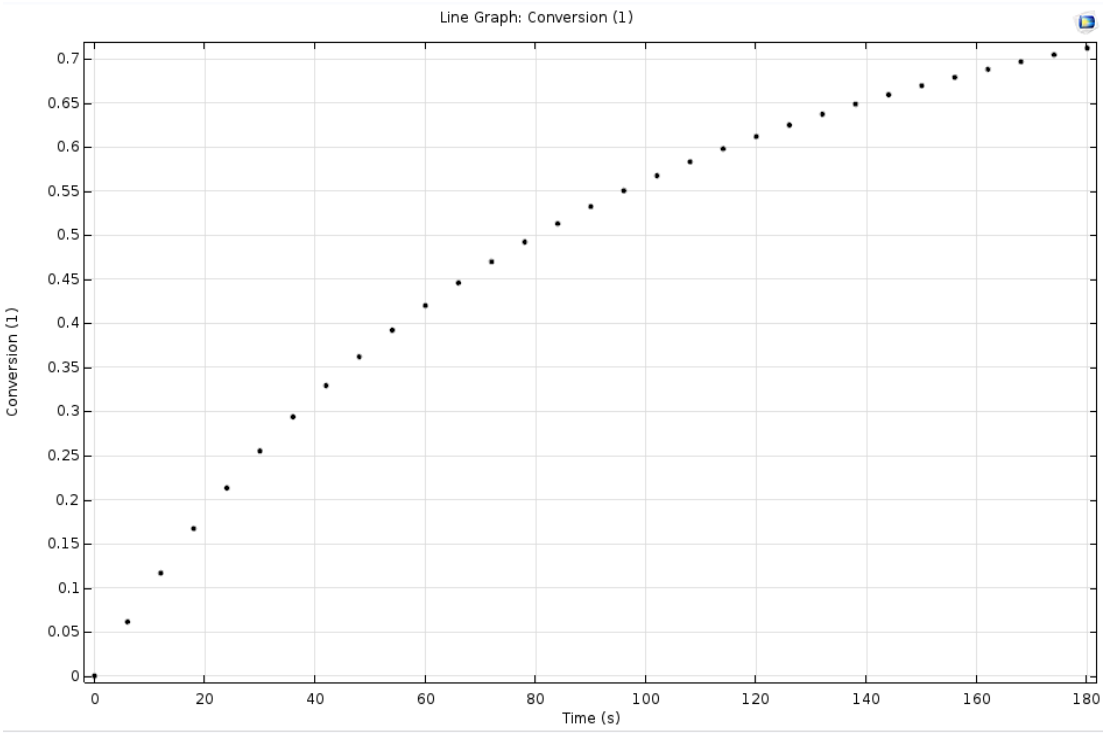

Figure 3.7 Conversion value of produced hydrogen gas from steam-metal oxide coating interface in oxidation reaction part with respect to time

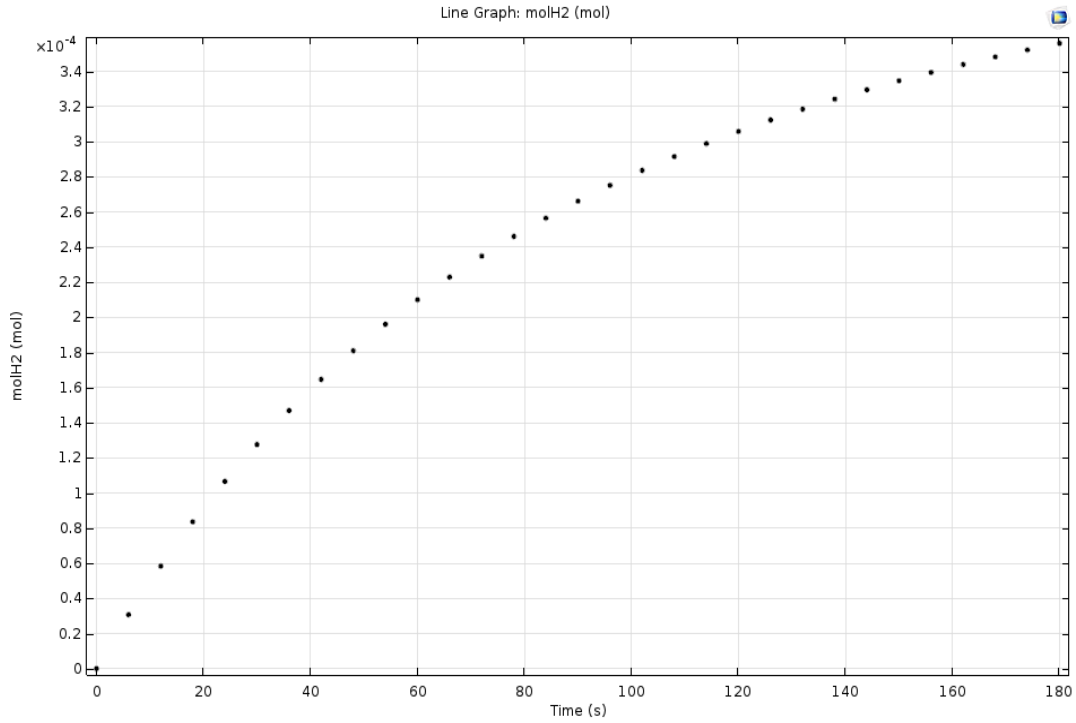

Figure 3.8 Mole amount of produced hydrogen gas from steam-metal oxide coating interface in oxidation reaction part with respect to time

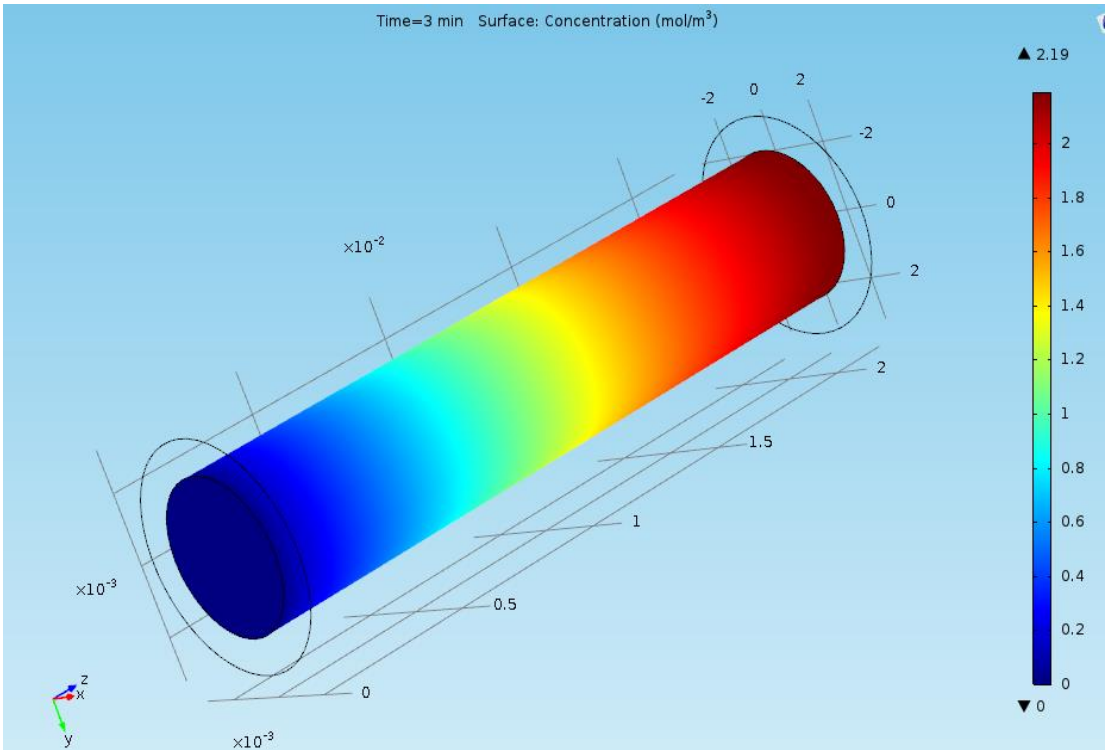

Figure 3.9 At third minute concentration distribution of produced hydrogen gas from steam-metal oxide coating interface in oxidation reaction part

As it is shown in the before figures, oxidation reaction (hydrogen production reaction) occurs faster than reduction reaction (oxygen releasing reaction). Although steam concentration decreases a little amount, it does not change significantly because of higher amount of inlet steam concentration. Hydrogen conversion value can be found as 0.7. Hydrogen production amount in single channel is not high since metal oxide coating amount is approximately 30-40 millimoles (approximately 3-4 grams) but general reactor systems have many channels (more than 100). It is indicated in the above Figure 3.8. Hydrogen concentration is higher toward the end of channel due to high temperature values and high reaction rate, little helium and steam flow effects. In this model system, helium and steam flowrate values are selected low (0.1 m/s) to operate stably and prevent different problems due to turbulence.

3.2 Kinetic Model Results

Kinetic model which includes first and second desorption reaction conversion (oxygen conversion), adsorption reaction conversion (hydrogen conversion) is configured by using MATLAB. These conversion values are calculated with respect to time through rate expression obtained from lab experiments and temperature profiles of COMSOL thermal model results on this program. Actually, this model results is used for improvement of statistical model analysis. Conversion values of kinetic model is outputs of statistical model analysis like heating time and effects of all parameters on conversion values are analyzed to predict real hydrogen-oxygen conversion values in this reactor system conditions. Especially, oxygen conversion value is more important than hydrogen conversion since desorption reaction occurs slowly and oxygen can release at high temperature. Hydrogen production is aim of this study but its reaction occurs fast and it is not needed heating. Thus, if oxygen can be released from metal oxide catalyst, hydrogen production is facilitated.

In this part, channel shape and material effects on oxygen conversion with respect to time are evaluated and shown through kinetic model and EXCEL program. Other parameters effects on oxygen conversion are analyzed detailed in thermal model and statistical approach results part.

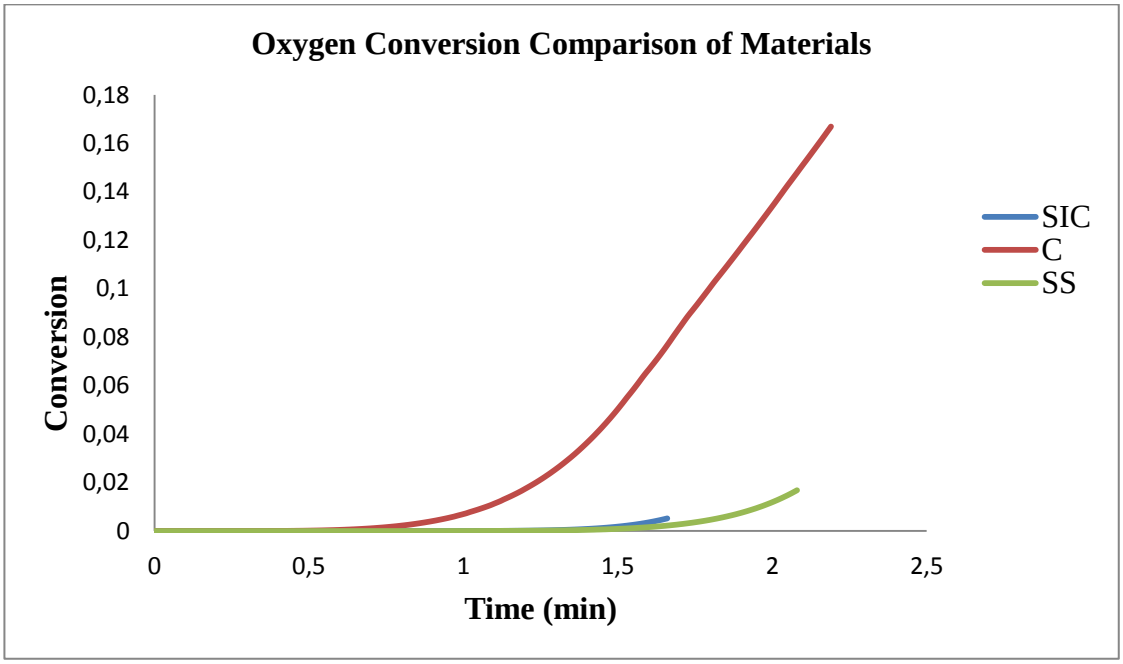

Figure 3.10 Oxygen conversion comparison of materials graph-1

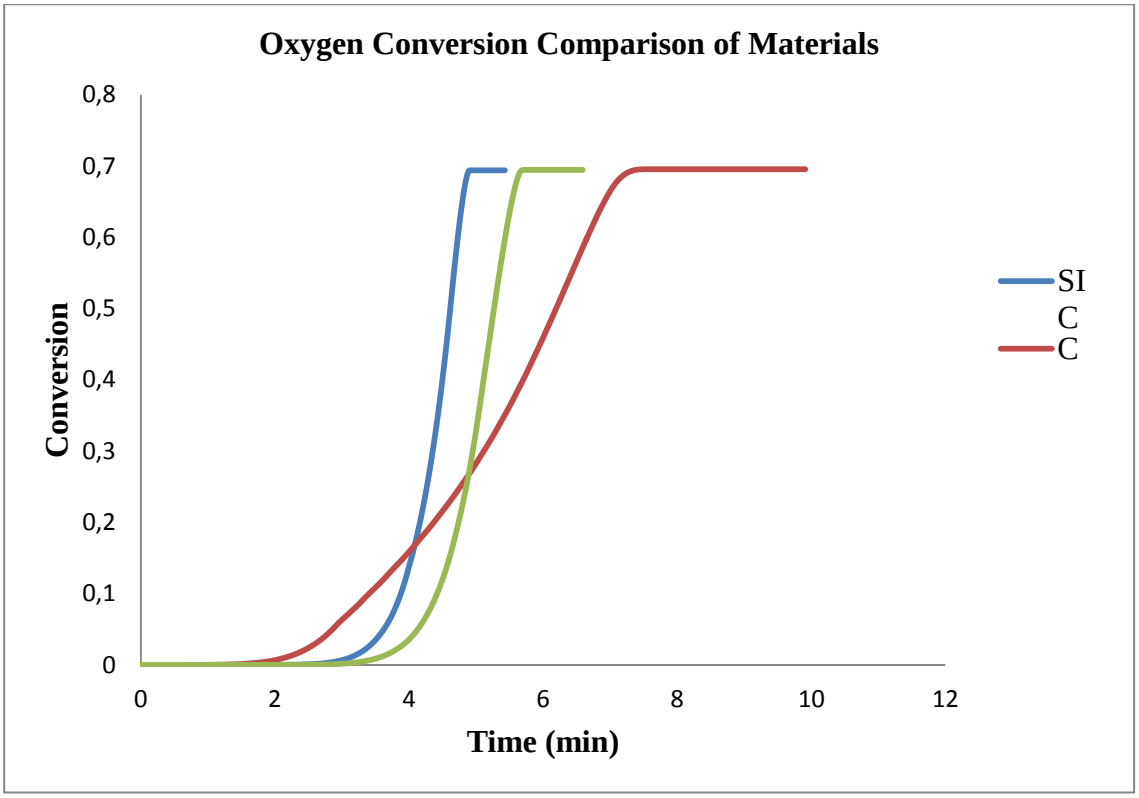

Figure 3.11 Oxygen conversion comparison of materials graph-2

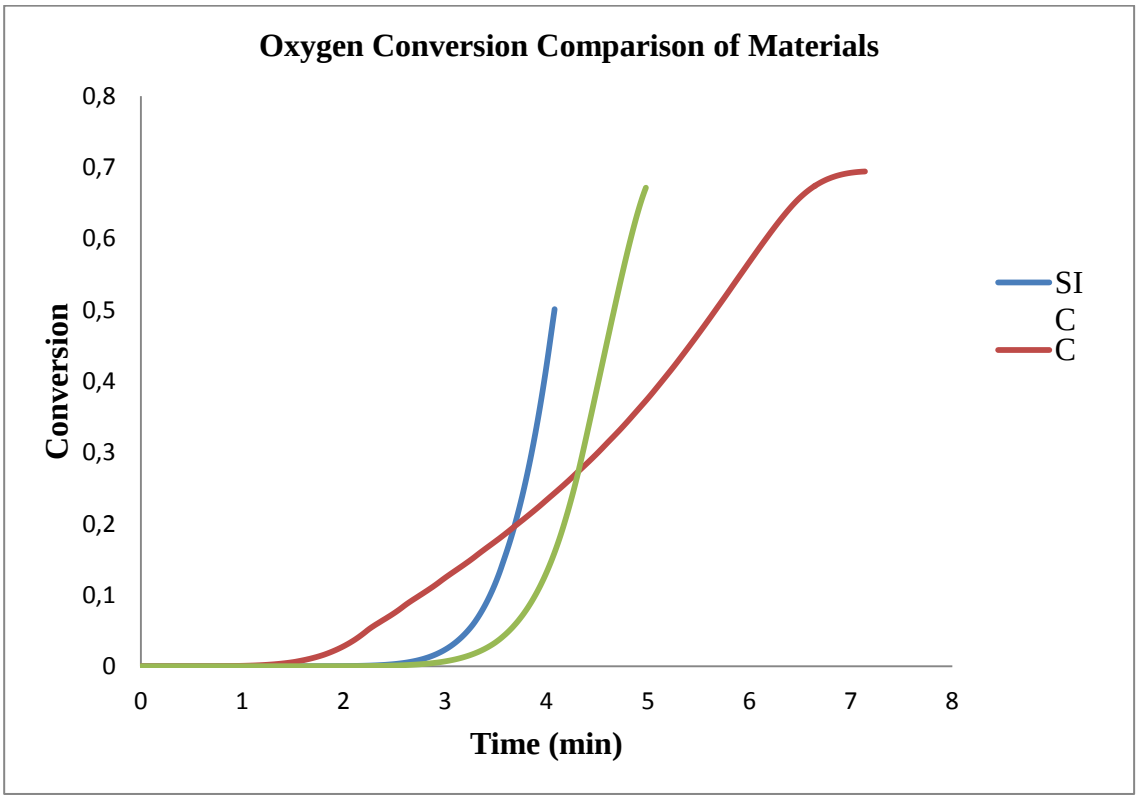

Figure 3.12 Oxygen conversion comparison of materials graph-3

As it is shown in Figure 3.10, cordierite has the highest conversion value in all materials due to low reduction temperature and short time interval of simulation run. Actually, kinetic rate expression depends on temperature basically. When system has low temperature profile, reaction rate value decreases. Silicon carbide and stainless steel have high thermal conductivity. Hence, their reactor systems have low temperature difference. For these systems, all points throughout channel have low temperature values in simulation part of low reduction parameter such as 900-1000 ºC. However, cordierite has low thermal conductivity and some points in front of channel have high temperature such as 1200 ºC, rest points have low temperature. Reaction rate of high temperature area for cordierite can compensate the other areas. Thus, cordierite is the best material for this simulation runs conditions. In Figure 3.11 and 3.12, silicon carbide has the highest conversion value in all materials after certain time. In these simulation runs, reduction temperature parameter is high and solar flux parameter is not high. So all points in silicon carbide channel can have high temperature values and this system has enough time to reach high reaction rate. Maximum conversion value can be obtained in this simulation run (0.7). If reactor system has 3-4 mins heating time, silicon carbide is the best material. If reactor system has shorter than 3 mins heating time, cordierite is the best material. Stainless steel is second good material in some simulation runs which have enough heating time. Generally, cordierite simulations have enough heating time because cordierite reactor reaches slowly reduction temperature due to its low thermal conductivity.

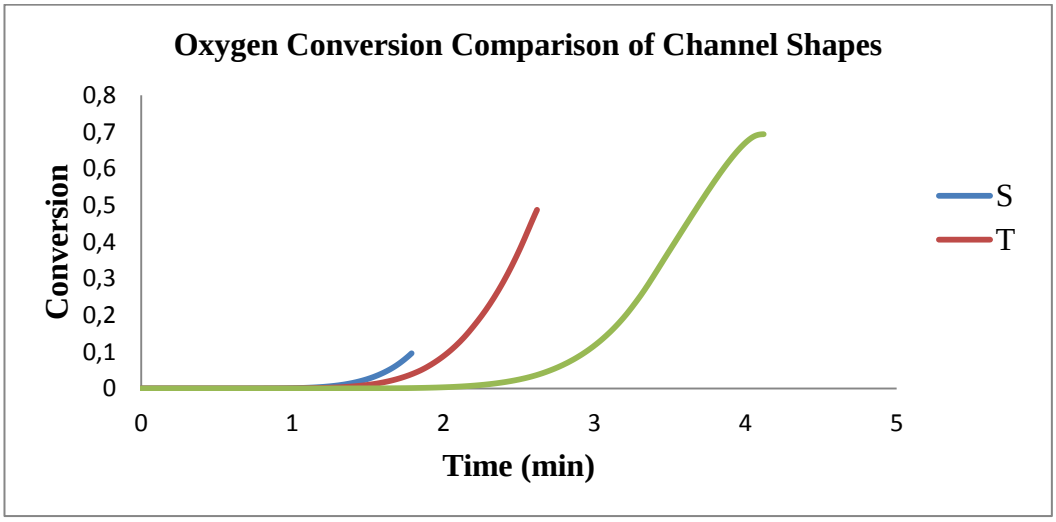

Figure 3.13 Oxygen conversion comparison of channel shapes graph-1

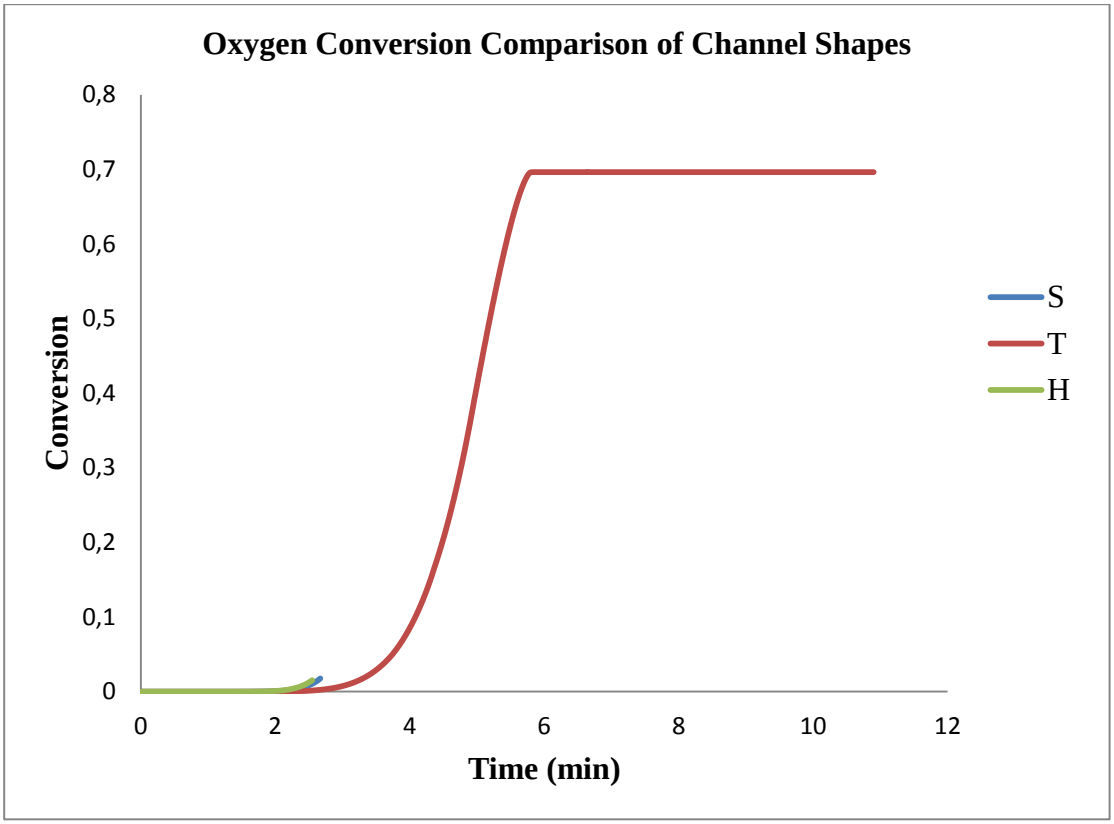

Figure 3.14 Oxygen conversion comparison of channel shapes graph-1

In Figure 3.13, reactor model with square channels has the highest conversion value at the beginning of simulation but reactor model with hexagon channels reaches the highest conversion value at the end of simulation. Square model has high surface area for solar flux (0.0007 m^2), low reactor length (15 mm) and high efficient heat transfer. They lead short heating time and low conversion value. Hexagon model has low surface area for solar flux (0.0004 m^2), high reactor length (26 mm). So it has enough heating time to reach high conversion value. In Figure 3.14, reactor model with triangle channels has the highest conversion value at the end of simulation. In this time, triangle model has low surface area (0.0002 m^2) and high reactor length (14 mm). In before part, triangle model is secondary good system in terms of conversion value. At the beginning of simulation run, hexagon model has the highest conversion value due to high surface area (0.00035 m^2) and low reactor length (13 mm) but it has not enough time to reach maximum conversion value like triangle model. As a result, all shape models have similar conversion values at a certain time such as between 0-2 min.

3.3 Thermal Model and Statistical Approach Results

In statistical approach of this study, main effects, second order effects and nonlinear effects of parameters are considered by setting simulation runs. First twenty-four simulation runs and sixty-six simulation runs are analyzed together in terms of temperature distribution and heating time. Effects of all parameters except channel shape are considered by using JMP program for these studies. Results of these studies are shown below.

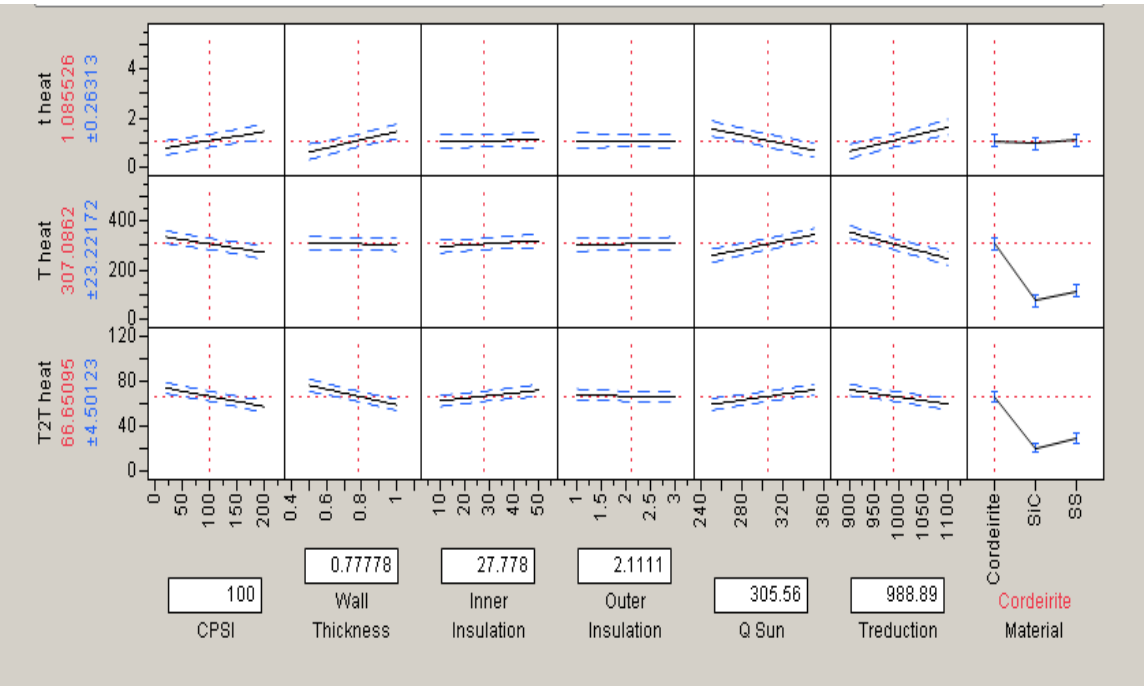

Figure 3.15 Effects of parameters on model temperature distribution and heating time in JMP analysis graph

According to JMP analysis, selection of silicon carbide as reactor material is rational due to high thermal conductivity. High CPSI decreases channel radius and length. In spite of decreasing channel length, heating time raises slightly because decreasing channel radius leads decreasing inside heat flux. Decreasing channel length influences positively temperature distribution and temperature difference decreases. High solar flux provides short heating time but it affects negatively temperature distribution. Reduction reaction temperature affects positively temperature distribution due to enough heating time. High wall thickness leads lower heat flux inside channels because solar flux heats the front face of reactor more. Thus, it decreases temperature difference among channels and increases heating time but it cannot affect significantly temperature difference along single channel. Inner and

outer insulations have not important impact on temperature distributions or time. Optimum conditions of reactor system are silicon carbide as reactor material, high CPSI, averaged 300 sun solar flux, thin wall thickness for minimum temperature difference. For this optimum model, heating time, temperature difference among channels and temperature difference along single channel are approximately 1-2 mins, 20-30 ºC and 50-100 ºC.

Finally, one hundred-sixty-two simulation runs including channel shape parameter are set and configured for consideration of main, second order and nonlinearity effects. In this final step, oxygen production rate is considered as output of statistical analysis like heating time and temperature distribution. Kinetic data from results of real experiment are combined with COMSOL temperature results through MATLAB and oxygen conversion values with respect to time are obtained for all conditions of simulation set. Also, hydrogen conversion values are calculated but its reaction time is restricted with cooling time of reactor and this leads so short reaction time. Despite fast adsorption reaction rate, hydrogen conversion values are slightly less and they are not useful. Thus, hydrogen conversion values are not analyzed. Analysis of only oxygen production rate is more rational because oxygen releasing reaction occurs at high temperature and it has slow reaction rate. The main aim of optimization solar concentrator system and reactor system is to perform this oxygen desorption reaction effectively and fast. Effective oxygen desorption reaction affects high hydrogen production amount directly. In these following figures and tables, all parameters impacts on oxygen production rate, heating time and temperature distributions are shown and tabulated.

Table 3.1 F ratios of important parameters for oxygen production rate

Source	Nparm	DF	Sum of Squares	F Ratio
Treduction	1	1	0.22	347.43
Q (Suns)	1	1	0.03	42.62
Wall Thickness (mm)*Q (Suns)	1	1	0.02	27.17
Q (Suns)*Treduction	1	1	0.02	26.78

Table 3.1 F ratios of important parameters for oxygen production rate (continued)

Material	2	2	0.02	18.85
Wall Thickness (mm)*Treduction	1	1	0.01	11.17
Material*Q (Suns)	2	2	0.01	10.59
Wall Thickness (mm)*Inner Insulation (mm)	1	1	0.01	7.84
Inner Insulation (mm)	1	1	0.004	7.47
Inner Insulation (mm)*Inner Insulation (mm)	1	1	0.004	6.68

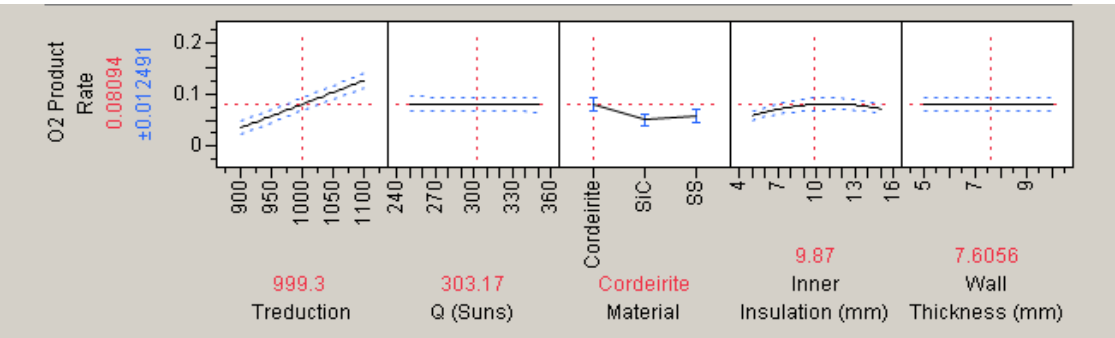

Figure 3.16 Effects of parameters on oxygen production rate in JMP analysis graph

As it is shown in Figure 3.16 and Table 3.1, CPSI and channel shape do not affect significantly to oxygen production rate. Cordierite reactor model has lower thermal conductivity than silicon carbide model but cordierite model has enough reaction time for high solar flux conditions. This concept was analyzed in kinetic model results part. High reaction temperature influences positively oxygen production rate because kinetic expression includes temperature term. Inner insulation has a optimum value for oxygen production rate. In addition, effects of wall thickness and solar flux are not important but these parameters can affect heating time. Thus, effects of these parameters on combined heating time and oxygen production rate can be evaluated.

Table 3.2 F ratios of important parameters for heating time

Source	Nparm	DF	Sum of Squares	F Ratio
Wall Thickness (mm)	1	1	162.17	129.59
Treduction	1	1	156.38	124.97
Q (Suns)	1	1	66.14	52.86
Q (Suns)*Treduction	1	1	35.58	28.43
Wall Thickness (mm)*Treduction	1	1	28.76	22.99
Material	2	2	24.74	9.89
Channel Shape	2	2	24.05	9.61
Wall Thickness (mm)*Inner Insulation (mm)	1	1	10.13	8.09
Wall Thickness (mm)*Q (Suns)	1	1	9.68	7.74
Material*Treduction	2	2	18.21	7.28
Inner Insulation (mm)	1	1	8.91	7.12
Q (Suns)*Q (Suns)	1	1	8.82	7.05
Material*Q (Suns)	2	2	13.93	5.56
Inner Insulation (mm)*Treduction	1	1	6.31	5.04

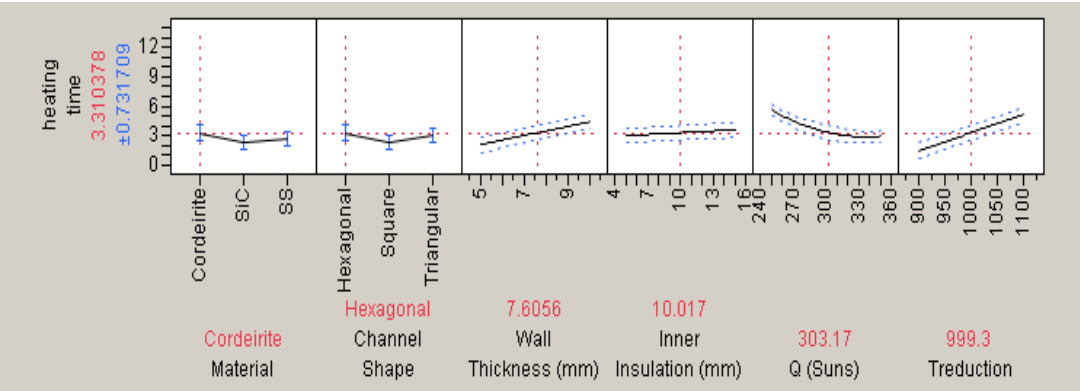

Figure 3.17 Effects of parameters on heating time in JMP analysis graph

Unlike the first analysis, CPSI does not increase heating time. Actually, time scale is different for two analysis, so changing heating time cannot be understand easily for this second analysis. Silicon carbide is the best reactor material for heating time despite low oxygen production rate. Square channel model and low wall thickness affect positively heating time. Solar flux parameter has optimum value for heating time analysis. These value is related to reduction reaction temperature directly proportional. Also, inner insulation has optimum value for general system but it does not affect heating time significantly.

Table 3.3 F ratios of important parameters for temperature difference among channels

Source	Nparm	DF	Sum of Squares	F Ratio
Material	2	2	38413.25	214.86
Wall Thickness (mm)	1	1	10612.96	118.72
Treduction	1	1	5030.73	56.28
Channel Shape	2	2	9887.15	55.30
Q (Suns)	1	1	4392.58	49.14
Channel Shape*CPSI	2	2	4310.80	24.11
CPSI*Wall Thickness (mm)	1	1	1525.87	17.07
Wall Thickness (mm)*Treduction	1	1	714.91	7.99
Material*Treduction	2	2	1278.58	7.15
Inner Insulation (mm)	1	1	618.49	6.92
CPSI	1	1	580.14	6.49
Material*Wall Thickness (mm)	2	2	1105.45	6.18
Wall Thickness (mm)*Q (Suns)	1	1	433.09	4.84
Material*CPSI	2	2	618.47	3.46

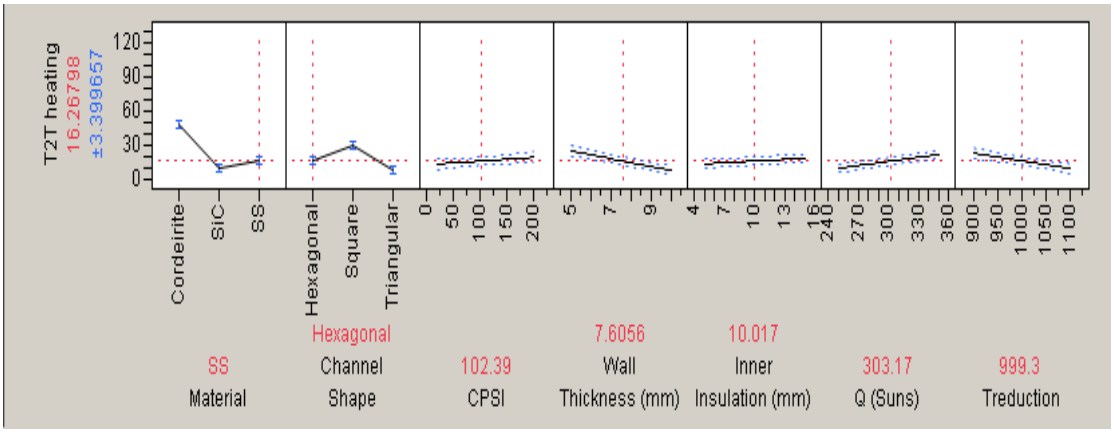

Figure 3.18 Effects of parameters on temperature difference among channels in JMP analysis graph

As it is shown in Figure 3.18 and Table 3.3, silicon carbide reactor model has the lowest temperature difference among channels. CPSI and ınner insulation affect slightly temperature distribution. High wall thickness has positive impact on temperature distribution. However, high solar flux affect and square channel model affect negatively to temperature difference among channels. They have positive effects on heating time.

Table 3.4 F ratios of important parameters for temperature difference along single channel

Source	Nparm	DF	Sum of Squares	F Ratio
Material	2	2	1599308.87	5007.51
Treduction	1	1	476069.36	2981.19
Q (Suns)	1	1	355989.31	2229.24
CPSI	1	1	213976.49	1339.94
Material*Treduction	2	2	250557.99	784.51
Material*Q (Suns)	2	2	110248.44	345.19
CPSI*Treduction	1	1	41040.98	257.0
CPSI*Q (Suns)	1	1	19228.33	120.41
CPSI*Wall Thickness (mm)	1	1	12187.14	76.32
Wall Thickness (mm)	1	1	8334.49	52.19

Table 3.4 F ratios of important parameters for temperature difference along single channel (continued)

Channel Shape	2	2	8629.41	27.02
Inner Insulation (mm)	1	1	3423.83	21.44
Material*CPSI	2	2	5977.87	18.72
Inner Insulation (mm)*Treduction	1	1	1021.26	6.39
Channel Shape*CPSI	2	2	1958.77	6.13
Material*Wall Thickness (mm)	2	2	1800.91	5.64

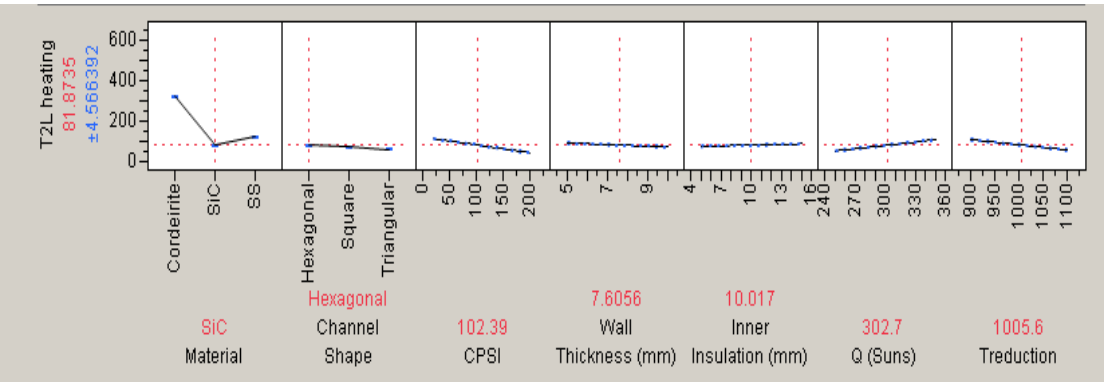

Figure 3.19 Effects of parameters on temperature difference along single channel in JMP analysis graph

Final criteria or output of this analysis is temperature difference along single channel. In Figure 3.19 and Table 3.4, channel shape, wall thickness and inner insulation do not affect significantly to temperature distributions along single channel. Silicon carbide has the lowest temperature difference due to its high thermal conductivity. CPSI parameter decreases temperature difference along single channel since high CPSI models have short reactor length. Solar flux affects negatively. It increases temperature difference for all systems.

In summary, when these four criteria of statistical analysis are evaluated in terms of order of importance, oxygen production rate is the most important output of this

study. Second important output of this study is heating time and finally reasonable temperature distribution is cared for reactor system. In the light of these information, high CPSI, high solar flux, square channel model, cordierite material, low wall thickness and optimum inner insulation thickness are defined as optimum conditions of reactor system. In this optimum reactor model, oxygen production rate is 0.15-0.20, heating time is 1-2 mins and all temperature differences are 50-200 º C. This optimum reactor model can obtain high oxygen conversion while it lose certain surface area of reactor channels due to temperature differences. Overall conversion values of this study are examined. High conversion and less surface model has more oxygen releasing amount than low conversion and high surface model. Hence, higher mole amount of hydrogen gas can be obtained by using this optimum model conditions. Main goal of this thesis is cost effective hydrogen gas production. This optimum reactor model has similar goals.

In order to determine reactor system energy efficiency, heat input and heating loss values are calculated. Firstly, reactor model is run at steady-state to identify and validate heat input and heating loss values. After that, four specific model is selected and reactor model is run at unsteady-state (transient) to determine system energy efficiency. These values are tabulated in Table 3.5.

Table 3.5 Heat input, heating loss, energy efficiency and heating time values of reactor model

Model	Time(min)	Losses (W)	Heat In(W)	Efficiency (%)
SIC-250	5.43	148.78	173.58	14.29
C-350	1.69	130.51	174.96	25.41
SIC-350	1.29	98.79	194.63	49.24
C-250	9.91	164.25	170.73	3.79

As it is shown in Table 3.5, silicon carbide models have greater efficiency value than cordierite models. Also, high solar flux can enhance system efficiency. The most important heating loss is radiative heating loss. If heating time of reactor model is short, this reactor model has high energy efficiency.

3.4 Model Validation

Heat, mass (including kinetic) and momentum transport phenomena are modeled and analyzed in this thesis study, but especially thermal model is focused more than the others since reduction and oxidation reactions occur at high temperature and it makes thermal model more important. Thermal model has a bit complex physics. It is defined for only solid materials but it has radiation, convection, conduction physical phenomena. Thermal model results are obtained through COMSOL but this model needs validation. In order to justify this model with real experiment, experimental setup is established. This setup contains basically dual axis tracking system, two 70 cm diameter parabolic dishes, disc system for flux validation, monolith reactor, some kind of insulation materials and K-type thermocouples. They are shown below in some figures.

Figure 3.20 Dual axis tracking system with two 70 cm diameter parabolic dishes

Figure 3.21 Monolith reactor (left) and disc system (right)

Disc system has 3 cm diameter and 1 cm height. It is made of stainless steel. It is enclosed in insulation material. This disc system is selected to validate solar heat flux because it has so simple geometry. Monolith reactor has 25 square shape reactor channels that have 1 cm side length with wall thickness. It has 10 cm depth and it is made of stainless steel. Also, it is enclosed in insulation material like disc. Glass wool is used as insulation material for all system models.

Model validation study comprises solar flux confirmation and temperature profiles confirmation of monolith reactor model. Disc system is used for solar flux confirmation in this study. Solar flux which comes from sun to concentrator is measured via pyranometer. However, this flux contains direct irradiation, diffuse irradiation and reflected irradiation. Also, concentrator efficiency is not %100. In order to determine the correct solar flux value, lumped temperature model for uniform disc temperature is configured. Inward solar heat flux, general system radiation loss and general system convection loss are considered for physics. Model equation is shown below.

$$\varrho \times V \times C_p \frac{dT}{dt} = F_{in} - A_{rad}\varepsilon\sigma\left(\mathrm{T_1}^4 - \mathrm{T_{amb}}^4\right) - A_{conv}h(T_1 - T_{amb}) \qquad (3.1)$$

Comparison of between model results and experimental results is done. In model part, reactor properties, emissivity and convective heat transfer coefficient are tuned to overlap reactor model results with experimental results. Comparison figures are indicated in below figures.

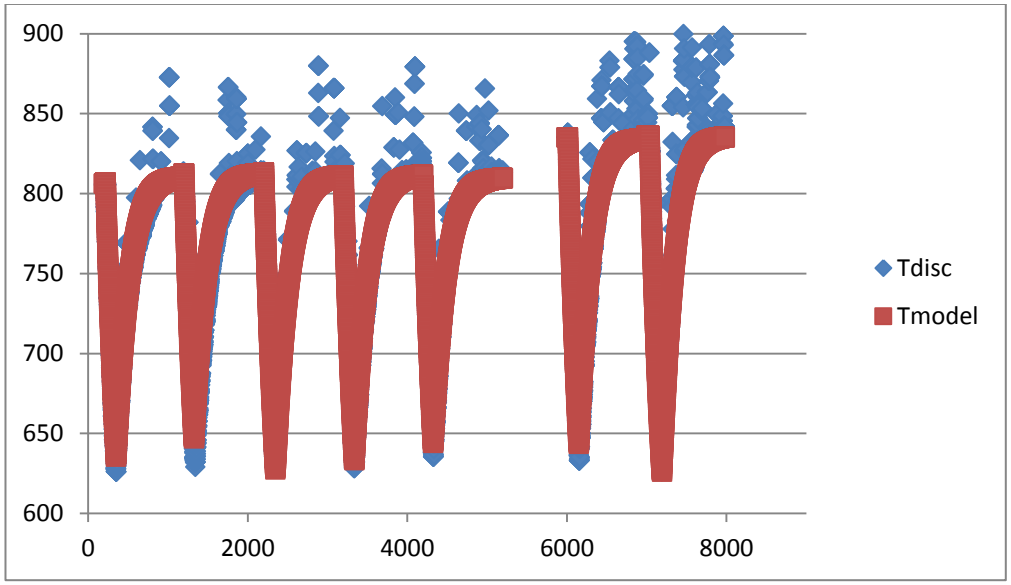

Figure 3.22 Comparison of model temperature profiles and real experimental temperature profiles for 2 min off focus-15 min on focus cycle

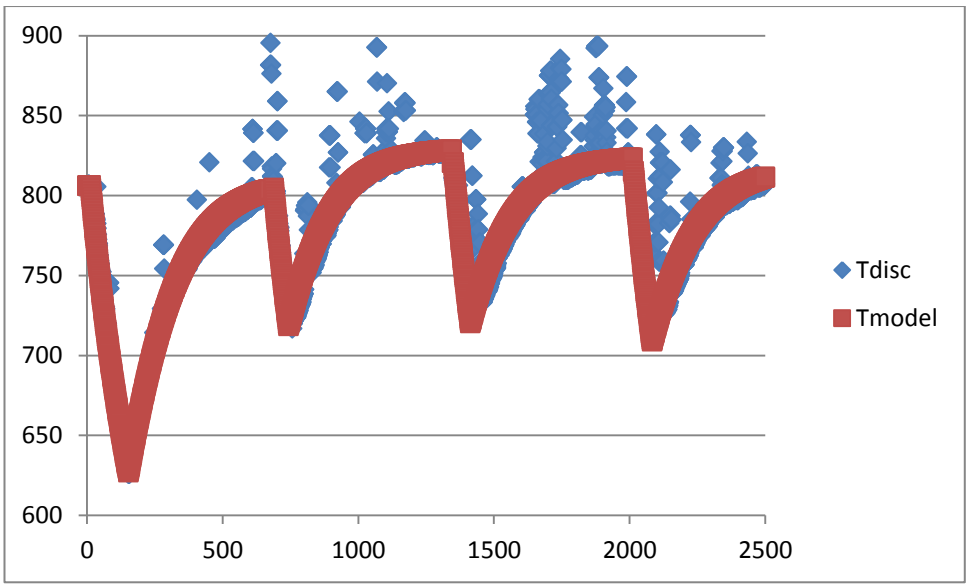

Figure 3.23 Comparison of model temperature profiles and real experimental temperature profiles for 1 min off focus-10 min on focus cycle

As it is seen in the above figures, model temperature profiles and real experimental temperature profiles have similar trend and values. Result of these similar trend and values, solar heat flux values of reactor model are calculated. These values are tabulated in Table 3.6.

Table 3.6 Model and ideal solar heat flux and efficiency values for each cycles

Test Date	Test Step	Q_{in} (W/m2)	T_{amb} (K)	Q_{model} (W)	Q_{ideal} (W)	Efficiency (%)
16-Sep-16	2min-15min-1	1048	305	315	403	78
16-Sep-16	2min-15min-2	1048	310	317	403	79
16-Sep-16	2min-15min-3	1048	304	317	403	78
16-Sep-16	2min-15min-4	1069	309	316	411	77
16-Sep-16	2min-15min-5	1069	308	314	411	76
17-Sep-16	1min-10min-1	1079	308	344	415	83
17-Sep-16	1min-10min-2	1074	300	346	413	84
17-Sep-16	1min-10min-2	1069	305	357	411	87

Model heat flux is taken as real heat flux because their temperature profiles overlap each other. Thus, efficiency values for each cycle are calculated. Average efficiency of all cycles is approximately %80. %20 loss of efficiency is based on diffuse radiation and concentrator efficiency loss. Average solar heat energy is taken as 330 W. This value is used in the temperature profiles confirmation of monolith reactor model.

Monolith reactor geometry is configured in COMSOL to simulate this model. Thermal conductivity of reactor, convection loss inside channel, emissivity of reactor and focus settings are tuned to overlap temperature profiles of COMSOL model with real experimental temperature profiles. Solar heat energy is taken as 330 W from previous part. Sky temperature is calculated instead of ambient temperature but it does not affect significantly. Also, thermocouple position errors are considered in this validation study. Validation graph of monolith reactor model is shown below. Reactor model simulation operates steadily in COMSOL.

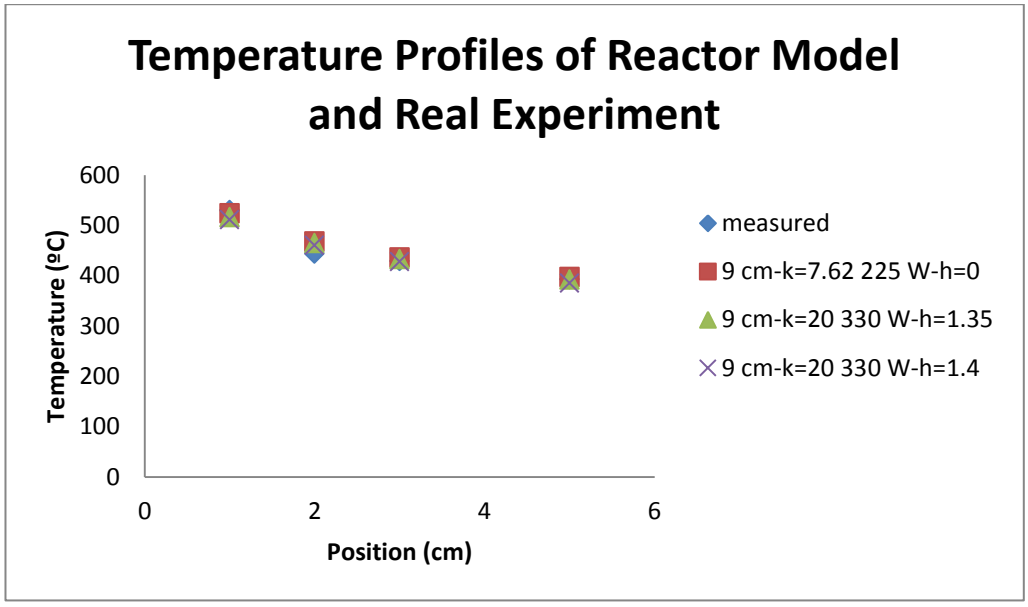

Figure 3.24 Temperature profiles of reactor model (steady) and real experiment

Real experimental temperature values are 530 ºC, 444 ºC, 429 ºC and 393 ºC. Some simulations of reactor model have similar trend and values. For example, model that is heated by 225 W has similar temperature profile. Its temperature values are 524 ºC, 468 ºC, 436 ºC and 397 ºC but its solar heat energy is not realistic. The other model results of validation study have same parameters except convective heat transfer coefficient. Their solar heat energy is 330 W like previous part. In addition, their temperature profiles overlap with measured temperature profiles. Thus, they have acceptable parameters. In this validation study, some analyses are done about these parameters. These parameters are tuned and reasons of new conditions are figured out.

Firstly, thermal conductivity value (k) is less than real value because reactor did not produce. Some metal plates are assembled each other for reactor. This situation leads to lessen thermal conductivity. It equals to 20 W/m*K. Emissivity is taken as 1 for before model simulations but real monolith reactor system does not have maximum efficiency of emissivity despite painting black. Both reactor emissivity and insulation material emissivity are less than previous model configuration. These values equal to 0.9 and 0.7. When real experiment occurs, there are some focusing problems. Normally, solar flux heats all 25 channels of monolith reactor but illuminated certain area is decreased and solar flux is concentrated to 9 cm^2 of reactor front side instead

of 25 cm^2. Experimental setup does not have any vacuum power. Also, front side of monolith reactor is not enclosed in any lens or apparatus. Therefore, this condition leads to convection loss from front side of reactor and inside reactor channel. In order to determine correct convection loss amount, convective heat transfer coefficient (h) is tuned. Correct convection loss amount equals to 40 W. Finally, position of second thermocouple (T2) is not correct position (2cm). Probably, this thermocouple is further away to first thermocouple because all model simulation results cannot close to real temperature value of second thermocouple. It has lower value than model results (444 ºC- 465 ºC).

CHAPTER 4

CONCLUSIONS

In this study, it is aimed to design and optimize water splitting monolith reactor system in terms of thermal, kinetic, mass transport and momentum physics through statistical approach, JMP, COMSOL and MATLAB programs. For this system, hydrogen is produced from steam by using solar energy and oxidation reaction of non-stoichiometric metal oxide.

In accordance with this aim, design of artificial experiment (DOE) is set by using JMP program to determine important parameters of reactor design in terms of thermal and kinetic model. After physical background studies, some of determined parameters are material selection for reactor and insulation, channel shape, wall thickness and CPSI (cell per square inch), thickness of insulation material, thickness of thin layer insulation material and solar heat flux for heating reactor. Not only these parameters relate to thermal model, but also they affect to kinetic model because of temperature dependence of kinetic rate expression. After determining process, thermal model is configured parametrically on COMSOL by using these parameters. Physical conditions are defined, mesh and solver selection are optimized with sensitivity analysis in Chapter 2. Then, kinetic model is configured on MATLAB program through thermal results data of COMSOL program and adsorption-desorption rate equations of real experiments to project conversion values. Model is compatible with real kinetic results since its error is less than %5 in Chapter 2. Transport model in a single channel is configured through COMSOL program to examine mass, momentum and heat transfer phenomena together. It includes reaction kinetic of real experiment data and diffusion, thermal model and assumed flow model (velocity and pressure) in Chapter 2. In this model, two dimensional axisymmetric geometry is used instead of three dimensional geometry due to symmetry. This symmetry assumption is validated via some pre-simulations.

In Chapter 3 for combined transport model on COMSOL, neglecting impacts of mass transfer and momentum transfer on temperature profiles is justified due to low

temperature differences for both two reactions. In desorption part (reduction), maximum temperature difference is observed in backside of channel and this value is obtained as between 0-10 ºC. In adsorption (oxidation) part, maximum temperature difference is 30-40 ºC. Oxidation part has slightly higher temperature difference than reduction part because of steam flow inside channel and no heating flux but both temperature differences is acceptable due to less than %10 error. Then, oxygen and hydrogen production amounts of two reactions, hydrogen conversion of oxidation reaction and molar flux values of metal oxide coating on channel surfaces are examined in Chapter 3. Hydrogen conversion value is obtained as 0.7 and it is observed that hydrogen concentration is higher toward the end of reactor channel owing to high temperature values and high reaction rate, little helium and steam flow effects.

In Chapter 3 for kinetic model on MATLAB, channel shape and material effects on oxygen conversion with respect to time are analyzed. Reduction temperature and short time interval of simulation run are significant parameters for material effect analysis. If heating time is shorter than 3 min, cordierite is the best material. If heating time is more than 3 min, silicon carbide is the best material in terms of oxygen conversion due to thermal conductivity. Silicon carbide needs enough heating time to obtain high conversion but cordierite has enough heating time since it reaches slowly to reduction temperature due to its low thermal conductivity. Surface area for solar flux and reactor length are very significant parameters for analysis of channel shape effect on oxygen conversion in Chapter 3. For 20 CPSI reactor system, square model is the best shape at the beginning of simulation but it has short heating time because of high surface area as 0.0007 m^2 and low reactor length as 15 mm. On the other hand, hexagon model is the best shape at the end of simulation because of low surface area as 0.0004 m^2 and high reactor length as 26mm for 20 CPSI reactor system. For the other model which is 200 CPSI reactor system, hexagon model is the best shape at the beginning of simulation because of high surface area as 0.00035 m^2 and low reactor length as 13 mm. However, triangle model is the best shape at the end of simulation because of low surface area as 0.0002 m^2 and high reactor length as 14 mm. Finally, all of shape models have similar conversion values at a certain time such as between 0-2 min.

In Chapter 3 for thermal model and statistical approach on COMSOL and JMP, all parameters effects on oxygen production rate, heating time, temperature difference among channels and temperature difference along single channel are analyzed. According to first and second statistical analysis not including channel shape effect, optimum conditions of reactor system are silicon carbide as reactor material, high CPSI, averaged 300 sun solar flux, thin wall thickness for minimum temperature difference. For this optimum model, heating time, temperature difference among channels and temperature difference along single channel are approximately 1-2 mins, 20-30 ºC and 50-100 ºC. According to final statistical analysis including all effects, optimum conditions of reactor system are high CPSI, high solar flux, square channel model, cordierite material, low wall thickness and optimum inner insulation thickness. When these optimum conditions are obtained, rational manner is carried out. Oxygen production rate is selected as the most important output of this study. Second important output of this study is heating time and finally reasonable temperature distribution is cared for reactor system. In this optimum reactor model, oxygen production rate is 0.15-0.20, heating time is 1-2 mins and all temperature differences are 50-200 º C. This optimum reactor model can obtain high oxygen conversion while it lose certain surface area of reactor channels due to temperature differences. This optimum reactor model has similar goals with main goal of this thesis.

In last part of Chapter 3, model validation is carried out for solar flux and temperature profiles of monolith reactor. Solar energy is determined 330 W and temperature profiles overlap each other by using tuning physical parameters. For steady-state model, real experimental temperature values are measured as 530 ºC, 444 ºC, 429 ºC and 393 ºC. Model temperature values are obtained as 516 ºC, 465 ºC, 433 ºC and 392 ºC.

To sum up, in this study it was aimed to model and design optimum reactor system in terms of heat, mass and momentum transport phenomena via statistical approach, JMP, COMSOL and MATLAB programs and it was succeeded. Final reactor system has optimum design selection criteria for especially thermal model. So that, this monolith reactor design can be used for water splitting reaction at high temperature and hydrogen gas can be produced because this monolith reactor design can reach

high temperature by using solar heat flux. Also, this modeling and design manners can be used other type of systems which has important thermal model phenomena.

REFERENCES

[1] R. Ramachandran and R. K. Menon, "An overview of industrial uses of hydrogen," Int. J. Hydrog. Energy, vol. 23, no. 7, pp. 593–598, Jul. 1998.

[2] "Hydrogen Production: Natural Gas Reforming | Department of Energy." [Online]. Available: http://energy.gov/eere/fuelcells/hydrogen-production-natural-gas-reforming. [Accessed: 07-Aug-2016].

[3] S. Lin, M. Harada, Y. Suzuki, and H. Hatano, "Hydrogen production from coal by separating carbon dioxide during gasification," Fuel, vol. 81, no. 16, pp. 2079–2085, Nov. 2002.

[4] A. C. C. Chang, H.-F. Chang, F.-J. Lin, K.-H. Lin, and C.-H. Chen, "Biomass gasification for hydrogen production," Int. J. Hydrog. Energy, vol. 36, no. 21, pp. 14252–14260, Oct. 2011.

[5] Y. Asada and J. Miyake, "Photobiological hydrogen production," J. Biosci. Bioeng., vol. 88, no. 1, pp. 1–6, Jan. 1999.

[6] "conference poster 2000." [Online]. Available: http://www.photobiology.com/photoiupac2000/wunschiers/. [Accessed: 08-Aug-2016].

[7] T. Duangmanee, S. I. Padmasiri, J. J. Simmons, L. Raskin, and S. Sung, "Hydrogen Production by Anaerobic Microbial Communities Exposed to Repeated Heat Treatments," Water Environ. Res., vol. 79, no. 9, pp. 975–983, 2007.

[8] "Manovich: Database as a Symbolic Form." [Online]. Available: http://www.personal.psu.edu/svh5507/research%20essay%20with%20pictures.html. [Accessed: 08-Aug-2016].

[9] "Hydrogen Production: Electrolysis | Department of Energy." [Online]. Available: http://energy.gov/eere/fuelcells/hydrogen-production-electrolysis. [Accessed: 08-Aug-2016].

[10] A. A. Ismail and D. W. Bahnemann, "Photochemical splitting of water for hydrogen production by photocatalysis: A review," Sol. Energy Mater. Sol. Cells, vol. 128, pp. 85–101, Sep. 2014.

[11] S. Abanades and G. Flamant, “Thermochemical hydrogen production from a two-step solar-driven water-splitting cycle based on cerium oxides,” Sol. Energy, vol. 80, no. 12, pp. 1611–1623, Dec. 2006.

[12] T. Kodama, S. Enomoto, T. Hatamachi, and N. Gokon, “Application of an Internally Circulating Fluidized Bed for Windowed Solar Chemical Reactor with Direct Irradiation of Reacting Particles,” J. Sol. Energy Eng., vol. 130, no. 1, p. 014504, 2008.

[13] N. Gokon, H. Yamamoto, N. Kondo, and T. Kodama, “Internally Circulating Fluidized Bed Reactor Using m-ZrO[sub 2] Supported NiFe[sub 2]O[sub 4] Particles for Thermochemical Two-Step Water Splitting,” J. Sol. Energy Eng., vol. 132, no. 2, p. 021102, 2010.

[14] N. Gokon, S. Takahashi, H. Yamamoto, and T. Kodama, “Thermochemical two-step water-splitting reactor with internally circulating fluidized bed for thermal reduction of ferrite particles,” Int. J. Hydrog. Energy, vol. 33, no. 9, pp. 2189–2199, May 2008.

[15] T. Kodama, Y. Nakamuro, and T. Mizuno, “A Two-Step Thermochemical Water Splitting by Iron-Oxide on Stabilized Zirconia,” J. Sol. Energy Eng., vol. 128, no. 1, p. 3, 2006.

[16] N. Gokon, S. Takahashi, H. Yamamoto, and T. Kodama, “New Solar Water-Splitting Reactor With Ferrite Particles in an Internally Circulating Fluidized Bed,” J. Sol. Energy Eng., vol. 131, no. 1, p. 011007, 2009.

[17] M. Roeb et al., “Solar Hydrogen Production by a Two-Step Cycle Based on Mixed Iron Oxides,” J. Sol. Energy Eng., vol. 128, no. 2, p. 125, 2006.

[18] M. Roeb et al., “Operational strategy of a two-step thermochemical process for solar hydrogen production,” Int. J. Hydrog. Energy, vol. 34, no. 10, pp. 4537–4545, May 2009.

[19] R. B. Diver, J. E. Miller, M. D. Allendorf, N. P. Siegel, and R. E. Hogan, “Solar Thermochemical Water-Splitting Ferrite-Cycle Heat Engines,” J. Sol. Energy Eng., vol. 130, no. 4, p. 041001, 2008.

[20] J. E. Miller, M. D. Allendorf, R. B. Diver, L. R. Evans, N. P. Siegel, and J. N. Stuecker, “Metal oxide composites and structures for ultra-high temperature solar thermochemical cycles,” J. Mater. Sci., vol. 43, no. 14, pp. 4714–4728, Jul. 2008.

[21] A. Steinfeld, “Solar thermochemical production of hydrogen—a review,” Sol. Energy, vol. 78, no. 5, pp. 603–615, May 2005.

[22] P. Haueter, S. Moeller, R. Palumbo, and A. Steinfeld, “The production of zinc by thermal dissociation of zinc oxide—solar chemical reactor design,” Sol. Energy, vol. 67, no. 1–3, pp. 161–167, Jul. 1999.

[23] D. Gstoehl, A. Brambilla, L. O. Schunk, and A. Steinfeld, “A quenching apparatus for the gaseous products of the solar thermal dissociation of ZnO,” J. Mater. Sci., vol. 43, no. 14, pp. 4729–4736, Jul. 2008.

[24] R. Müller, P. Haeberling, and R. D. Palumbo, “Further advances toward the development of a direct heating solar thermal chemical reactor for the thermal dissociation of ZnO(s),” Sol. Energy, vol. 80, no. 5, pp. 500–511, May 2006.

[25] R. Müller and A. Steinfeld, “H2O-splitting thermochemical cycle based on ZnO/Zn-redox: Quenching the effluents from the ZnO dissociation,” Chem. Eng. Sci., vol. 63, no. 1, pp. 217–227, Jan. 2008.

[26] R. Müller, W. Lipiński, and A. Steinfeld, “Transient heat transfer in a directly-irradiated solar chemical reactor for the thermal dissociation of ZnO,” Appl. Therm. Eng., vol. 28, no. 5–6, pp. 524–531, Apr. 2008.

[27] R. Müller and A. Steinfeld, “Band-approximated radiative heat transfer analysis of a solar chemical reactor for the thermal dissociation of zinc oxide,” Sol. Energy, vol. 81, no. 10, pp. 1285–1294, Oct. 2007.

[28] P. Charvin, S. Abanades, P. Neveu, F. Lemont, and G. Flamant, “Dynamic modeling of a volumetric solar reactor for volatile metal oxide reduction,” Chem. Eng. Res. Des., vol. 86, no. 11, pp. 1216–1222, Nov. 2008.

[29] L. O. Schunk, P. Haeberling, S. Wepf, D. Wuillemin, A. Meier, and A. Steinfeld, “A Receiver-Reactor for the Solar Thermal Dissociation of Zinc Oxide,” J. Sol. Energy Eng., vol. 130, no. 2, p. 021009, 2008.

[30] L. O. Schunk, W. Lipiński, and A. Steinfeld, “Heat transfer model of a solar receiver-reactor for the thermal dissociation of ZnO—Experimental validation at 10kW and scale-up to 1MW,” Chem. Eng. J., vol. 150, no. 2–3, pp. 502–508, Aug. 2009.

[31] T. Melchior, C. Perkins, A. W. Weimer, and A. Steinfeld, “A cavity-receiver containing a tubular absorber for high-temperature thermochemical processing using concentrated solar energy,” Int. J. Therm. Sci., vol. 47, no. 11, pp. 1496–1503, Nov. 2008.

[32] T. Melchior and A. Steinfeld, “Radiative Transfer Within a Cylindrical Cavity With Diffusely/Specularly Reflecting Inner Walls Containing an Array of Tubular Absorbers,” J. Sol. Energy Eng., vol. 130, no. 2, p. 021013, 2008.

[33] S. Haussener, D. Hirsch, C. Perkins, A. Weimer, A. Lewandowski, and A. Steinfeld, “Modeling of a Multitube High-Temperature Solar Thermochemical Reactor for Hydrogen Production,” J. Sol. Energy Eng., vol. 131, no. 2, p. 024503, 2009.

[34] K. Wegner, H. Ly, R. Weiss, S. Pratsinis, and A. Steinfeld, “In situ formation and hydrolysis of Zn nanoparticles for H2H2 production by the 2-step ZnO/Zn water-splitting thermochemical cycle,” Int. J. Hydrog. Energy, vol. 31, no. 1, pp. 55–61, Jan. 2006.

[35] F. O. Ernst, A. Tricoli, S. E. Pratsinis, and A. Steinfeld, “Co-synthesis of H_2 and ZnO by in-situ Zn aerosol formation and hydrolysis,” AIChE J., vol. 52, no. 9, pp. 3297–3303, Sep. 2006.

[36] T. Abu Hamed, L. Venstrom, A. Alshare, M. Brülhart, and J. H. Davidson, “Study of a Quench Device for the Synthesis and Hydrolysis of Zn Nanoparticles: Modeling and Experiments,” J. Sol. Energy Eng., vol. 131, no. 3, p. 031018, 2009.

[37] K.-S. Kang, C.-H. Kim, W.-C. Cho, K.-K. Bae, S.-H. Kim, and C.-S. Park, “Novel two-step thermochemical cycle for hydrogen production from water using germanium oxide: KIER 4 thermochemical cycle,” Int. J. Hydrog. Energy, vol. 34, no. 10, pp. 4283–4290, May 2009.

[38] J. Chen, H. Yang, N. Wang, Z. Ring, and T. Dabros, “Mathematical modeling of monolith catalysts and reactors for gas phase reactions,” Appl. Catal. Gen., vol. 345, no. 1, pp. 1–11, Jul. 2008.

[39] R. Jahn, D. Snita, M. Kubíček, and M. Marek, “3-D modeling of monolith reactors,” Catal. Today, vol. 38, no. 1, pp. 39–46, Oct. 1997.

[40] R. Pitchumani, T. Ishida, N. Gokon, T. Hatamachi, and T. Kodama, “Proceedings of the SolarPACES 2013 International ConferenceKinetics of Thermal Reduction Step of Thermochemical Two-step Water Splitting Using CeO2 Particles: MASTER-plot Method for Analyzing Non-isothermal Experiments,” Energy Procedia, vol. 49, pp. 1970–1979, Jan. 2014.

APPENDICES

A MATLAB CODES

A.1 Combination code of COMSOL Thermal Results and Kinetic Model

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc;
clear all;
close all;
path = 'C:\Users\HP\Desktop\Triangle';
path2 = 'C:\Users\HP\Desktop\TriangleEx';

files = dir(path);

%Parameters
Fa0=1; %Initial mole/cm^2
lmtconversionde=0.7; %conversion limit of desorption
lmtconversionad=0.6; %conversion limit of adsorption
Eadesorb=72843; % J/mol  activation energy of desorption
Eaadsorb=40289; %J/mol   activation energy of adsorption
R=8.314; %gas constant  J/(mol*K)
Fadesorb(1)=0;        % initial desorption value t=0
for id=1:length(files)
```

```
if strfind(files(id).name,'.txt') >0
    file2read = [path '\' files(id).name];
    runID=files(id).name(1:strfind(files(id).name,'.txt')-1);
    excel2write= [path2 '\' num2str(runID) '.xlsx'];
    %import data operations
    [data,delimiter,header] = importdata(file2read,' ',9);
    times = char(data.textdata(9));
    data = (data.data);
        start = strfind(times,'=') + 1;
        stop = strfind(times,'T') - 2;
            stop(1) = [];
            stop = [stop length(times)];
            time = [];
            for j = 1:length(stop)
                time(j) = str2double(times(start(j):stop(j)));
            end
            time = time';
        temp = size(data);
        T = data(:,4:temp(2));
        Kelvin=273*ones(size(T));
        TK=T+Kelvin;
        for o=2:length(time)
            if TK(1,o)-TK(1,o-1)<0
                break
            end
        end
        % 'o-1' is stopping time of first heating step
        for p=o:length(time)
            if TK(1,p)-TK(1,p-1)>0
                break
            end
        end
        % 'p-1' is stopping time of first cooling step
```

```
%first desorption
 for k=1:length(TK)
    conversion(k,1)=0;  %initial conversion value at t=0
       for i=2:o-1
       a=time(i,1)-time(1,1);
       b=time(i,1)-time(i-1,1);
       heatingrate=(TK(k,i)-TK(k,i-1))./b;
       Temp(k,i)=heatingrate.*(time(i,1)-time(i-1,1))+TK(k,i-1);
       fun= @(x) exp(-8761.5./(heatingrate.*(x-time(i-1,1))+TK(k,i-1)));
       integ(k,i)=quad(fun,time(i-1,1),time(i,1));
       son=(integ(k,i)*343)+(conversion(k,i-1))^(1/3);
       conversion(k,i)=son^3;
       diff(k,i)=TK(k,i)-Temp(k,i);
       if conversion(k,i)>0.1
          conversion(k,i)=conversion(k,i)*1.053;
          for j=i+1:o-1;
             a=time(j,1)-time(1,1);
             b=time(j,1)-time(j-1,1);
             heatingrate=(TK(k,j)-TK(k,j-1))./b;
             Temp(k,j)=heatingrate.*(time(j,1)-time(j-1,1))+TK(k,j-1);
             fun= @(x) exp(-8761.5./(heatingrate.*(x-time(j-1,1))+TK(k,j-1)));
             integ(k,j)=quad(fun,time(j-1,1),time(j,1));
             son=(integ(k,j)*343)+(conversion(k,j-1))^(1/3);
             conversion(k,j)=son^3;
             diff(k,j)=TK(k,j)-Temp(k,j);
             if conversion(k,j)>lmtconversionde
                for e=j:o-1
                   conversion(k,e)=conversion(k,j-1);
                end
                break
             end
          end
       break
       end
```

```
            end
        end
%           %adsorption
        for k=1:length(TK)
          conversionad(k,o-1)=0;  %initial conversion value at t=o-1
          for i=o:p-1
            a=time(i,1)-time(1,1);
            b=time(i,1)-time(i-1,1);
            heatingrate=(TK(k,i-1)-TK(k,i))./b;
            Temp(k,i)=TK(k,i-1)-heatingrate.*(time(i,1)-time(i-1,1));
            fun= @(x) exp(-4845.9./(TK(k,i-1)-heatingrate.*(x-time(i-1,1))));
            integ(k,i)=quad(fun,time(i-1,1),time(i,1));
            son=(integ(k,i)*37.4)+conversionad(k,i-1);
            conversionad(k,i)=son;
            diff(k,i)=TK(k,i)-Temp(k,i);
              if  conversionad(k,i)>conversion(k,o-1)
                for w=i:p-1
                    conversionad(k,w)=conversionad(k,i-1);
                end
                break
              end
             if conversionad(k,i)>0.11
              conversionad(k,i)=conversionad(k,i)*1.73;
               for j=i+1:p-1;
                 a=time(j,1)-time(1,1);
                 b=time(j,1)-time(j-1,1);
                 heatingrate=(TK(k,j-1)-TK(k,j))./b;
                 Temp(k,j)=TK(k,j-1)- heatingrate.*(time(j,1)-time(j-1,1));
                 fun= @(x) exp(-4845.9./(TK(k,j-1)-heatingrate.*(x-time(j-1,1))));
                 integ(k,j)=quad(fun,time(j-1,1),time(j,1));
                 son=(integ(k,j)*37.4)+(conversionad(k,j-1));
                 conversionad(k,j)=son;
                 diff(k,j)=TK(k,j)-Temp(k,j);
                  if conversionad(k,j)>conversion(k,o-1)
```

```
                for h=j:p-1
                conversionad(k,h)=conversionad(k,j-1);
                end
                break
            end
        end
        break
    end
end
end
%second desorption
limitconversionde=0.7;
  for k=1:length(TK)
    limitconversiondeger=1-conversion(k,o-1)+conversionad(k,p-1);
    if limitconversiondeger<limitconversionde
      limitconversionde=limitconversiondeger;
    end
    conversionde(k,p-1)=0;  %initial conversion value at t=0
    for i=p:length(time)
    a=time(i,1)-time(1,1);
    b=time(i,1)-time(i-1,1);
    heatingrate=(TK(k,i)-TK(k,i-1))./b;
    Temp(k,i)=heatingrate.*(time(i,1)-time(i-1,1))+TK(k,i-1);
    fun= @(x) exp(-8761.5./(heatingrate.*(x-time(i-1,1))+TK(k,i-1)));
    integ(k,i)=quad(fun,time(i-1,1),time(i,1));
    son=(integ(k,i)*343)+(conversionde(k,i-1))^(1/3);
    conversionde(k,i)=son^3;
    diff(k,i)=TK(k,i)-Temp(k,i);
    if conversionde(k,i)>0.1
      conversionde(k,i)=conversionde(k,i)*1.053;
      for j=i+1:length(time);
        a=time(j,1)-time(1,1);
        b=time(j,1)-time(j-1,1);
        heatingrate=(TK(k,j)-TK(k,j-1))./b;
```

```
                Temp(k,j)=heatingrate.*(time(j,1)-time(j-1,1))+TK(k,j-1);
                fun= @(x) exp(-8761.5./(heatingrate.*(x-time(j-1,1))+TK(k,j-1)));
                integ(k,j)=quad(fun,time(j-1,1),time(j,1));
                son=(integ(k,j)*343)+(conversionde(k,j-1))^(1/3);
                conversionde(k,j)=son^3;
                diff(k,j)=TK(k,j)-Temp(k,j);
                if conversionde(k,j)>limitconversionde
                    for u=j:length(time)
                    conversionde(k,u)=conversionde(k,j-1);
                    end
                    break
                end
            end
        break
        end
        end
    end
    %plotting procedure
for n=1:length(TK)
   xdata(n)=data(n,1);
end
for n=1:(o-1)
   cond(n)=0;
   timed(n)=time(n);
   for g=1:length(TK)
   cond(n)=conversion(g,n)+cond(n);
   end
   cond(n)=cond(n)/length(TK);
   exceldata(n,1)=timed(n);
   exceldata(n,2)=cond(n);
end
for y=(o-1):(p-1)
   cona(y)=0;
   timea(y)=time(y);
```

```
    for r=1:length(TK)
        cona(y)=conversionad(r,y)+cona(y);
    end
    cona(y)=cona(y)/length(TK);
    exceldata(y,3)=timea(y);
    exceldata(y,4)=cona(y);
end
for q=(p-1):length(time)
    conde(q)=0;
    timef(q)=time(q);
    for u=1:length(TK)
        conde(q)=conversionde(u,q)+conde(q);
    end
    conde(q)=conde(q)/length(TK);
    exceldata(q,5)=timef(q);
    exceldata(q,6)=conde(q);
end
 xlswrite(excel2write,exceldata)
end
clear all;
    path = 'C:\Users\HP\Desktop\Triangle';
    path2 = 'C:\Users\HP\Desktop\TriangleEx';
    files = dir(path);

%Parameters
Fa0=1; %Initial mole/cm^2
lmtconversionde=0.7; %conversion limit of desorption
lmtconversionad=0.6; %conversion limit of adsorption
Eadesorb=72843; % J/mol  activation energy of desorption
Eaadsorb=40289; %J/mol   activation energy of adsorption
R=8.314; %gas constant  J/(mol*K)
Fadesorb(1)=0;        % initial desorption value t=0
end
% figure
```

```
% plot(timed,cond,'r',timea,cona,'b',timef,conde,'r')
% xlabel('time(mins)');
% ylabel('Average conversion values');
% legend('Desorbed Oxygen','Adsorbed Hydrogen')
% grid on
% plot(xdata,con,'.')
% xlabel('length to tube');
% ylabel('Total conversion of oxygen released');
% grid on
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```

A.2 Evaluation code of COMSOL Square Model Thermal Results(20-200 CPSI)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%

clc
clear all
close all

path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Square Final DOE Results';
```

```
path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Square Final DOE Results Ex';


DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal Results\Square Final Thermal DOE',1);

files = dir(path);
k = 0;

for i = 1:length(files)

   if strfind(files(i).name,'.txt') > 0
      time = [];
      file2read = [path '\' files(i).name];
      excelID=files(i).name(1:strfind(files(i).name,'.txt')-1);
      excel2write= [path2 '\' num2str(excelID) '.xlsx'];
      runID =i-2;
      %data = dlmread(file2read,'',10,0);
      [data,delimiter,header] = importdata(file2read,' ',9);
      times = char(data.textdata(9));
      data = (data.data);
         start = strfind(times,'=') + 1;
         stop = strfind(times,'T') - 2;
            stop(1) = [];
            stop = [stop length(times)];
            time = [];
            for j = 1:length(stop)
               time(j) = str2double(times(start(j):stop(j)));
            end
            time = time';
         z = data(:,1) * 1000;
         x = data(:,2) * 1000;
            x = x - mean(x);
```

```
    y = data(:,3) * 1000;
       y = y - mean(y);
    temp = size(data);
    T = data(:,4:temp(2));

N = DOE(runID,2);
CPSI = DOE(runID,3)/25.4/25.4;
t = DOE(runID,4);
    switch N
        case 3
            L = sqrt(4/sqrt(3)/CPSI);
            s = L - sqrt(3)/2 * t;
            R = 1/sqrt(3) * s;
            A = sqrt(3)/4 * s^2;
            P = 3*s;
        case 4
            L = sqrt(1/CPSI);
            s = L - t;
            R = sqrt(2)/2 * s;
            A = s^2;
            P = 4*s;
        case 6
            L = sqrt(2/3/sqrt(3)/CPSI);
            s = L - 1/sqrt(3) * t;
            R = s;
            A = 3*sqrt(3)/2 * s^2;
            P = 6*s;
    end

%       figure(runID)
%       plot(x,y,'.')

if DOE(runID,3) == 200
    grid = [s/2+t/2];
```

```
    else
        grid = [0];
    end

        for j = 2:round(max(x)/(s+t))+1
            grid(j) = grid(j-1) + (s+t)
        end
        grid = [-1*fliplr(grid) grid]
%           for j = 1:length(grid)
%               line([grid(j) grid(j)],[min(grid) max(grid)]);
%               line([min(grid) max(grid)],[grid(j) grid(j)]);
%           end
%
%
%       axis square
%       title(num2str(runID))

    tubex = [];
    tubey = [];

    for j = 1:length(x)
        tubex(j) = min(find((x(j)<grid)))-1;
        tubey(j) = min(find((y(j)<grid)))-1;
    end
    T2T_range = [];
    T_range = [];
    T_avg = [];

    for j = 1:length(time)

        Tnow = T(:,j);
        [tubemean tuberange] = grpstats(Tnow,{tubex' tubey'},{'mean','range'});
        T2T_range(j) = range(tubemean);
        T_range(j) = mean(tuberange);
```

```
            T_avg(j) = mean(Tnow);
            n_tubes = length(tubemean);
            exceldata(j,1)=time(j);
            exceldata(j,2)=T2T_range(j);
            exceldata(j,3)=T_range(j);
            exceldata(j,4)=T_avg(j);
            exceldata(1,6)=n_tubes;
        end
        xlswrite(excel2write,exceldata)
    end
    clear all
    path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Square Final
DOE Results';
    path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Square Final
DOE Results Ex';
    DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal Results\Square
Final Thermal DOE',1);
    files = dir(path);
    k = 0;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```

A.3 Evaluation code of COMSOL Triangle Model Thermal Results(20 CPSI)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%

clear all
close all

path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle20 Final DOE Results';
path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle20 Final DOE Results Ex';



DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle20 Final Thermal DOE',1);

files = dir(path);
k = 0;

for i = 1:length(files)

    if strfind(files(i).name,'.txt') > 0
        time = [];
        file2read = [path '\' files(i).name];
        excelID=files(i).name(1:strfind(files(i).name,'.txt')-1);
        excel2write= [path2 '\' num2str(excelID) '.xlsx'];
        runID =i-2;
        %data = dlmread(file2read,'',10,0);
        [data,delimiter,header] = importdata(file2read,' ',9);
        times = char(data.textdata(9));
        data = (data.data);
            start = strfind(times,'=') + 1;
            stop = strfind(times,'T') - 2;
                stop(1) = [];
                stop = [stop length(times)];
```

```
        time = [];
        for j = 1:length(stop)
            time(j) = str2double(times(start(j):stop(j)));
        end
        time = time';
    z = data(:,1) * 1000;
    x = data(:,2) * 1000;
        x = x - mean(x);
    y = data(:,3) * 1000;
        y = y - mean(y);
    temp = size(data);
    T = data(:,4:temp(2));

N = DOE(runID,2);
CPSI = DOE(runID,3)/25.4/25.4;
t = DOE(runID,4);
    switch N

        case 4
            L = sqrt(1/CPSI);
            s = L - t;
            R = sqrt(2)/2 * s;
            A = s^2;
            P = 4*s;
        case 6
            L = sqrt(2/3/sqrt(3)/CPSI);
            s = L - 1/sqrt(3) * t;
            R = s;
            A = 3*sqrt(3)/2 * s^2;
            P = 6*s;
        case 3
            L = sqrt(4/sqrt(3)/CPSI);
            s = L - sqrt(3) * t;
            R = 1/sqrt(3) * s;
```

```
A = sqrt(3)/4 * s^2;
P = 3*s;

if DOE(runID,3) == 20

   % define type 1 center grid
   for q=11:1:12
      tubedata1(q,1)=(q-11)*(s+t*sqrt(3));
      tubedata1(q,2)=R+t;
   end

   for q=11:-1:9
      tubedata1(q,1)=(q-11)*(s+t*sqrt(3));
      tubedata1(q,2)=R+t;
   end

   for q=5:8:13
      tubedata1(q,1)=(-2)*(s+t*sqrt(3));
      tubedata1(q,2)=((q-5)/8)*((3*R)+(3*t))-R-t;
   end

   for q=9:-8:1
      tubedata1(q,1)=(-2)*(s+t*sqrt(3));
      tubedata1(q,2)=((q-9)/8)*((3*R)+(3*t))+R+t;
   end

   for q=6:8:14
      tubedata1(q,1)=(-1)*(s+t*sqrt(3));
      tubedata1(q,2)=((q-6)/8)*((3*R)+(3*t))-R-t;
   end

   for q=10:-8:2
      tubedata1(q,1)=(-1)*(s+t*sqrt(3));
      tubedata1(q,2)=((q-10)/8)*((3*R)+(3*t))+R+t;
```

```
end

for q=7:8:15
   tubedata1(q,1)=0;
   tubedata1(q,2)=((q-7)/8)*((3*R)+(3*t))-R-t;
end

for q=11:-8:3
   tubedata1(q,1)=0;
   tubedata1(q,2)=((q-11)/8)*((3*R)+(3*t))+R+t;
end

for q=8:8:16
   tubedata1(q,1)=(1)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-8)/8)*((3*R)+(3*t))-R-t;
end

for q=12:-8:4
   tubedata1(q,1)=(1)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-12)/8)*((3*R)+(3*t))+R+t;
end
% define type 2 center grid
tubedata2(11,1)=(1/2)*L;
tubedata2(11,2)=(1/2)*(R+t);
tubedata2(12,1)=tubedata2(11,1)+(s+t*sqrt(3));
tubedata2(12,2)=(1/2)*(R+t);
tubedata2(10,1)=tubedata2(11,1)-(s+t*sqrt(3));
tubedata2(10,2)=(1/2)*(R+t);
tubedata2(9,1)=tubedata2(11,1)-2*(s+t*sqrt(3));
tubedata2(9,2)=(1/2)*(R+t);
for q=7:8:15
   tubedata2(q,1)=(1/2)*L;
   tubedata2(q,2)=tubedata2(11,2)-R-t+((q-7)/8)*((3*R)+(3*t));
end
```

```
for q=11:-8:3
   tubedata2(q,1)=(1/2)*L;
   tubedata2(q,2)=tubedata2(11,2)+((q-11)/8)*((3*R)+(3*t));
end

for q=8:8:16
   tubedata2(q,1)=tubedata2(11,1)+(s+t*sqrt(3));
   tubedata2(q,2)=tubedata2(12,2)-R-t+((q-8)/8)*((3*R)+(3*t));
end

for q=12:-8:4
   tubedata2(q,1)=tubedata2(11,1)+(s+t*sqrt(3));
   tubedata2(q,2)=tubedata2(12,2)+((q-12)/8)*((3*R)+(3*t));
end

for q=6:8:14
   tubedata2(q,1)=tubedata2(11,1)-(s+t*sqrt(3));
   tubedata2(q,2)=tubedata2(10,2)-R-t+((q-6)/8)*((3*R)+(3*t));
end

for q=10:-8:2
   tubedata2(q,1)=tubedata2(11,1)-(s+t*sqrt(3));
   tubedata2(q,2)=tubedata2(10,2)+((q-10)/8)*((3*R)+(3*t));
end

for q=5:8:13
   tubedata2(q,1)=tubedata2(11,1)-2*(s+t*sqrt(3));
   tubedata2(q,2)=tubedata2(9,2)-R-t+((q-5)/8)*((3*R)+(3*t));
end

for q=9:-8:1
   tubedata2(q,1)=tubedata2(11,1)-2*(s+t*sqrt(3));
   tubedata2(q,2)=tubedata2(9,2)+((q-9)/8)*((3*R)+(3*t));
```

```
end

%define numbergridx1
sayi=round(max(x)/(s+t*sqrt(3)));
toplamsayi=2*sayi;
ortasayi=1+toplamsayi/2;
numbergridx1(ortasayi)=L/2;
numbergridx1(ortasayi+1)=numbergridx1(ortasayi)+L;
for f=ortasayi-1:-1:1
   numbergridx1(f)=numbergridx1(f+1)-L;
end
%define numbergridx2
sayi=round(max(x)/L)+1;
toplamsayi=2*sayi-1;
ortasayi=(1+toplamsayi)/2;
numbergridx2(ortasayi)=0;
numbergridx2(ortasayi+1)=L;
numbergridx2(ortasayi+2)=2*L;
numbergridx2(ortasayi-1)=-L;
numbergridx2(ortasayi-2)=-2*L;
%define numbergridy1
sayi=round(max(y)/(R+t));
toplamsayi=2*sayi-1;
ortasayi=(1+toplamsayi)/2;
numbergridy1(ortasayi)=0;
for b=ortasayi+1:toplamsayi
numbergridy1(b)=numbergridy1(b-1)+(3/2)*(R+t);
end
for f=ortasayi:-1:1
   numbergridy1(f)=numbergridy1(f+1)-(3/2)*(R+t);
end
%define numbergridy2
sayi=round(max(y)/(R+t));
toplamsayi=2*sayi-1;
```

```
ortasayi=(1+toplamsayi)/2;
numbergridy2(ortasayi)=0;
for b=ortasayi+1:toplamsayi
numbergridy2(b)=numbergridy2(b-1)+(3/2)*(R+t);
end
for f=ortasayi:-1:1
   numbergridy2(f)=numbergridy2(f+1)-(3/2)*(R+t);
end
% determine which type of grid is convenient for all points
for o=1:1:length(z)
   mindistance1=10000;
   for w=1:1:length(tubedata1)
distance1=sqrt((x(o,1)-tubedata1(w,1))^2+(y(o,1)-tubedata1(w,2))^2);
      if mindistance1>distance1
         mindistance1=distance1;
      end
   end
   mindistance2=10000;
   for w=1:1:length(tubedata2)
distance2=sqrt((x(o,1)-tubedata2(w,1))^2+(y(o,1)-tubedata2(w,2))^2);
      if mindistance2>distance2
         mindistance2=distance2;
      end
   end
   if mindistance1<mindistance2
     tubex(o) =2*(min(find((x(o,1)<numbergridx1)))-1);
     tubey(o) =2*(min(find((y(o,1)<numbergridy1)))-1);
   end

   if mindistance2<mindistance1
     tubex(o) =2*(min(find((x(o,1)<numbergridx2)))-1)-1;
     tubey(o) =2*(min(find((y(o,1)<numbergridy2)))-1)-1;
   end
```

```
                end


%       figure(runID)
%       plot(x,y,'.')

        T2T_range = [];
        T_range = [];
        T_avg = [];

        for j = 1:1:length(time)
        Tnow = T(:,j);
            [tubemean tuberange] = grpstats(Tnow,{tubex' tubey'},{'mean','range'});
            T2T_range(j) = range(tubemean);
            T_range(j) = mean(tuberange);
            T_avg(j) = mean(Tnow);
            n_tubes = length(tubemean);
            exceldata(j,1)=time(j);
            exceldata(j,2)=T2T_range(j);
            exceldata(j,3)=T_range(j);
            exceldata(j,4)=T_avg(j);
            exceldata(1,6)=n_tubes;

        end
        xlswrite(excel2write,exceldata)
                    end
            end
    end
    clear all
    path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle20
Final DOE Results';
    path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle20
Final DOE Results Ex';
```

```
   DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal
Results\Triangle20 Final Thermal DOE',1);

   files = dir(path);
   k = 0;
end
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %
```

A.4 Evaluation code of COMSOL Triangle Model Thermal Results (200 CPSI)

```
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %

clear all
close all

path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle200 Final
DOE Results';
path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle200
Final DOE Results Ex';
```

```
DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal
Results\Triangle200 Final Thermal DOE',1);

files = dir(path);
k = 0;


for i = 1:length(files)

   if strfind(files(i).name,'.txt') > 0
      time = [];
      file2read = [path '\' files(i).name];
      excelID=files(i).name(1:strfind(files(i).name,'.txt')-1);
      excel2write= [path2 '\' num2str(excelID) '.xlsx'];
      runID =i-2;
      %data = dlmread(file2read,'',10,0);
      [data,delimiter,header] = importdata(file2read,' ',9);
      times = char(data.textdata(9));
      data = (data.data);
         start = strfind(times,'=') + 1;
         stop = strfind(times,'T') - 2;
            stop(1) = [];
            stop = [stop length(times)];
            time = [];
            for j = 1:length(stop)
               time(j) = str2double(times(start(j):stop(j)));
            end
            time = time';
         z = data(:,1) * 1000;
         x = data(:,2) * 1000;
            x = x - mean(x);
         y = data(:,3) * 1000;
            y = y - mean(y);
         temp = size(data);
```

```
    T = data(:,4:temp(2));

N = DOE(runID,2);
CPSI = DOE(runID,3)/25.4/25.4;
t = DOE(runID,4);
    switch N

        case 4
            L = sqrt(1/CPSI);
            s = L - t;
            R = sqrt(2)/2 * s;
            A = s^2;
            P = 4*s;
        case 6
            L = sqrt(2/3/sqrt(3)/CPSI);
            s = L - 1/sqrt(3) * t;
            R = s;
            A = 3*sqrt(3)/2 * s^2;
            P = 6*s;
        case 3
            L = sqrt(4/sqrt(3)/CPSI);
            s = L - sqrt(3) * t;
            R = 1/sqrt(3) * s;
            A = sqrt(3)/4 * s^2;
            P = 3*s;

            if DOE(runID,3) == 200

                % define type 1 center grid
                for q=72:1:77
                    tubedata1(q,1)=(q-72)*(s+t*sqrt(3));
                    tubedata1(q,2)=R+t;
                end
```

```
for q=72:-1:67
   tubedata1(q,1)=(q-72)*(s+t*sqrt(3));
   tubedata1(q,2)=R+t;
end

for q=83:1:88
   tubedata1(q,1)=(q-83)*(s+t*sqrt(3));
   tubedata1(q,2)=2*R+2*t;
end

for q=83:-1:78
   tubedata1(q,1)=(q-83)*(s+t*sqrt(3));
   tubedata1(q,2)=2*R+2*t;
end
%67-77
%67
for q=67:22:111
   tubedata1(q,1)=(-5)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-67)/22)*((3*R)+(3*t))+R+t;
end

for q=67:-22:1
   tubedata1(q,1)=(-5)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-67)/22)*((3*R)+(3*t))+R+t;
end
%68
for q=68:22:112
   tubedata1(q,1)=(-4)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-68)/22)*((3*R)+(3*t))+R+t;
end

for q=68:-22:2
   tubedata1(q,1)=(-4)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-68)/22)*((3*R)+(3*t))+R+t;
```

```
end
%69
for q=69:22:113
   tubedata1(q,1)=(-3)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-69)/22)*((3*R)+(3*t))+R+t;
end

for q=69:-22:3
   tubedata1(q,1)=(-3)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-69)/22)*((3*R)+(3*t))+R+t;
end
%70
for q=70:22:114
   tubedata1(q,1)=(-2)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-70)/22)*((3*R)+(3*t))+R+t;
end

for q=70:-22:4
   tubedata1(q,1)=(-2)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-70)/22)*((3*R)+(3*t))+R+t;
end
%71
for q=71:22:115
   tubedata1(q,1)=(-1)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-71)/22)*((3*R)+(3*t))+R+t;
end

for q=71:-22:5
   tubedata1(q,1)=(-1)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-71)/22)*((3*R)+(3*t))+R+t;
end
%72
for q=72:22:116
   tubedata1(q,1)=0;
```

```
    tubedata1(q,2)=((q-72)/22)*((3*R)+(3*t))+R+t;
end

for q=72:-22:6
    tubedata1(q,1)=0;
    tubedata1(q,2)=((q-72)/22)*((3*R)+(3*t))+R+t;
end
%73
for q=73:22:117
    tubedata1(q,1)=(1)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-73)/22)*((3*R)+(3*t))+R+t;
end

for q=73:-22:7
    tubedata1(q,1)=(1)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-73)/22)*((3*R)+(3*t))+R+t;
end
%74
for q=74:22:118
    tubedata1(q,1)=(2)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-74)/22)*((3*R)+(3*t))+R+t;
end

for q=74:-22:8
    tubedata1(q,1)=(2)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-74)/22)*((3*R)+(3*t))+R+t;
end
%75
for q=75:22:119
    tubedata1(q,1)=(3)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-75)/22)*((3*R)+(3*t))+R+t;
end

for q=75:-22:9
```

```
    tubedata1(q,1)=(3)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-75)/22)*((3*R)+(3*t))+R+t;
end
%76
for q=76:22:120
    tubedata1(q,1)=(4)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-76)/22)*((3*R)+(3*t))+R+t;
end

for q=76:-22:10
    tubedata1(q,1)=(4)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-76)/22)*((3*R)+(3*t))+R+t;
end
%77
for q=77:22:121
    tubedata1(q,1)=(5)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-77)/22)*((3*R)+(3*t))+R+t;
end

for q=77:-22:11
    tubedata1(q,1)=(5)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-77)/22)*((3*R)+(3*t))+R+t;
end
%78-88
%78
for q=78:22:122
    tubedata1(q,1)=(-5)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-78)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=78:-22:12
    tubedata1(q,1)=(-5)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-78)/22)*((3*R)+(3*t))+2*R+2*t;
end
```

```
%79
for q=79:22:123
   tubedata1(q,1)=(-4)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-79)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=79:-22:13
   tubedata1(q,1)=(-4)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-79)/22)*((3*R)+(3*t))+2*R+2*t;
end
%80
for q=80:22:124
   tubedata1(q,1)=(-3)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-80)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=80:-22:14
   tubedata1(q,1)=(-3)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-80)/22)*((3*R)+(3*t))+2*R+2*t;
end
%81
for q=81:22:125
   tubedata1(q,1)=(-2)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-81)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=81:-22:15
   tubedata1(q,1)=(-2)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-81)/22)*((3*R)+(3*t))+2*R+2*t;
end
%82
for q=82:22:126
   tubedata1(q,1)=(-1)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-82)/22)*((3*R)+(3*t))+2*R+2*t;
```

```
end

for q=82:-22:16
 tubedata1(q,1)=(-1)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-82)/22)*((3*R)+(3*t))+2*R+2*t;
end
%83
for q=83:22:127
   tubedata1(q,1)=0;
   tubedata1(q,2)=((q-83)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=83:-22:17
   tubedata1(q,1)=0;
   tubedata1(q,2)=((q-83)/22)*((3*R)+(3*t))+2*R+2*t;
end
%84
for q=84:22:128
   tubedata1(q,1)=(1)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-84)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=84:-22:18
   tubedata1(q,1)=(1)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-84)/22)*((3*R)+(3*t))+2*R+2*t;
end
%85
for q=85:22:129
   tubedata1(q,1)=(2)*(s+t*sqrt(3));
   tubedata1(q,2)=((q-85)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=85:-22:19
   tubedata1(q,1)=(2)*(s+t*sqrt(3));
```

```
    tubedata1(q,2)=((q-85)/22)*((3*R)+(3*t))+2*R+2*t;
end
%86
for q=86:22:130
    tubedata1(q,1)=(3)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-86)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=86:-22:20
    tubedata1(q,1)=(3)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-86)/22)*((3*R)+(3*t))+2*R+2*t;
end
%87
for q=87:22:131
    tubedata1(q,1)=(4)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-87)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=87:-22:21
    tubedata1(q,1)=(4)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-87)/22)*((3*R)+(3*t))+2*R+2*t;
end
%88
for q=88:22:122
    tubedata1(q,1)=(5)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-88)/22)*((3*R)+(3*t))+2*R+2*t;
end

for q=88:-22:22
    tubedata1(q,1)=(5)*(s+t*sqrt(3));
    tubedata1(q,2)=((q-88)/22)*((3*R)+(3*t))+2*R+2*t;
end
%%%%%%
% define type 2 center grid
```

```
for q=66:1:70
   tubedata2(q,1)=(q-66)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=(R+t)/2;
end

for q=66:-1:61
   tubedata2(q,1)=(q-66)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=(R+t)/2;
end

for q=56:1:60
   tubedata2(q,1)=(q-56)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=-(R+t)/2;
end

for q=56:-1:51
   tubedata2(q,1)=(q-56)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=-(R+t)/2;
end
%61-70
%61
for q=61:20:101
   tubedata2(q,1)=(-5)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-61)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=61:-20:1
   tubedata2(q,1)=(-5)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-61)/20)*((3*R)+(3*t))+(R+t)/2;
end
%62
for q=62:20:102
   tubedata2(q,1)=(-4)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-62)/20)*((3*R)+(3*t))+(R+t)/2;
```

```
end

for q=62:-20:2
   tubedata2(q,1)=(-4)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-62)/20)*((3*R)+(3*t))+(R+t)/2;
end
%63
for q=63:20:103
   tubedata2(q,1)=(-3)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-63)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=63:-20:3
   tubedata2(q,1)=(-3)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-63)/20)*((3*R)+(3*t))+(R+t)/2;
end
%64
for q=64:20:104
   tubedata2(q,1)=(-2)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-64)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=64:-20:4
   tubedata2(q,1)=(-2)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-64)/20)*((3*R)+(3*t))+(R+t)/2;
end
%65
for q=65:20:105
   tubedata2(q,1)=(-1)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-65)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=65:-20:5
   tubedata2(q,1)=(-1)*(s+t*sqrt(3))+L/2;
```

```
    tubedata2(q,2)=((q-65)/20)*((3*R)+(3*t))+(R+t)/2;
end
%66
for q=66:20:106
    tubedata2(q,1)=(0)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-66)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=66:-20:6
    tubedata2(q,1)=(0)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-66)/20)*((3*R)+(3*t))+(R+t)/2;
end
%67
for q=67:20:107
    tubedata2(q,1)=(1)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-67)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=67:-20:7
    tubedata2(q,1)=(1)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-67)/20)*((3*R)+(3*t))+(R+t)/2;
end
%68
for q=68:20:108
    tubedata2(q,1)=(2)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-68)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=68:-20:8
    tubedata2(q,1)=(2)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-68)/20)*((3*R)+(3*t))+(R+t)/2;
end
%69
for q=69:20:109
```

```
    tubedata2(q,1)=(3)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-69)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=69:-20:9
    tubedata2(q,1)=(3)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-69)/20)*((3*R)+(3*t))+(R+t)/2;
end
%70
for q=70:20:110
    tubedata2(q,1)=(4)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-70)/20)*((3*R)+(3*t))+(R+t)/2;
end

for q=70:-20:10
    tubedata2(q,1)=(4)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-70)/20)*((3*R)+(3*t))+(R+t)/2;
end
%51-60
%51
for q=51:20:111
    tubedata2(q,1)=(-5)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-51)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=51:-20:11
    tubedata2(q,1)=(-5)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-51)/20)*((3*R)+(3*t))-(R+t)/2;
end
%52
for q=52:20:112
    tubedata2(q,1)=(-4)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-52)/20)*((3*R)+(3*t))-(R+t)/2;
end
```

```
for q=52:-20:12
   tubedata2(q,1)=(-4)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-52)/20)*((3*R)+(3*t))-(R+t)/2;
end
%53
for q=53:20:113
   tubedata2(q,1)=(-3)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-53)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=53:-20:13
   tubedata2(q,1)=(-3)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-53)/20)*((3*R)+(3*t))-(R+t)/2;
end
%54
for q=54:20:114
   tubedata2(q,1)=(-2)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-54)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=54:-20:14
   tubedata2(q,1)=(-2)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-54)/20)*((3*R)+(3*t))-(R+t)/2;
end
%55
for q=55:20:115
   tubedata2(q,1)=(-1)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-55)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=55:-20:15
   tubedata2(q,1)=(-1)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-55)/20)*((3*R)+(3*t))-(R+t)/2;
```

```
end
%56
for q=56:20:116
   tubedata2(q,1)=(0)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-56)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=56:-20:16
   tubedata2(q,1)=(0)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-56)/20)*((3*R)+(3*t))-(R+t)/2;
end
%57
for q=57:20:117
   tubedata2(q,1)=(1)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-57)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=57:-20:17
   tubedata2(q,1)=(1)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-57)/20)*((3*R)+(3*t))-(R+t)/2;
end
%58
for q=58:20:118
   tubedata2(q,1)=(2)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-58)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=58:-20:18
   tubedata2(q,1)=(2)*(s+t*sqrt(3))+L/2;
   tubedata2(q,2)=((q-58)/20)*((3*R)+(3*t))-(R+t)/2;
end
%59
for q=59:20:119
   tubedata2(q,1)=(3)*(s+t*sqrt(3))+L/2;
```

```
    tubedata2(q,2)=((q-59)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=59:-20:19
    tubedata2(q,1)=(3)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-59)/20)*((3*R)+(3*t))-(R+t)/2;
end
%60
for q=60:20:120
    tubedata2(q,1)=(4)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-60)/20)*((3*R)+(3*t))-(R+t)/2;
end

for q=60:-20:20
    tubedata2(q,1)=(4)*(s+t*sqrt(3))+L/2;
    tubedata2(q,2)=((q-60)/20)*((3*R)+(3*t))-(R+t)/2;
end

%define numbergridx1
sayi=round(max(x)/(s+t*sqrt(3)))+1;
toplamsayi=2*sayi;
ortasayi=1+toplamsayi/2;
numbergridx1(ortasayi)=L/2;
for f=ortasayi+1:1:toplamsayi
numbergridx1(f)=numbergridx1(f-1)+L;
end
for f=ortasayi-1:-1:1
    numbergridx1(f)=numbergridx1(f+1)-L;
end
%define numbergridx2
sayi=round(max(x)/L);
toplamsayi=2*sayi+1;
ortasayi=(1+toplamsayi)/2;
numbergridx2(ortasayi)=0;
```

```
for f=ortasayi+1:1:toplamsayi
numbergridx2(f)=numbergridx2(f-1)+L;
end
for f=ortasayi-1:-1:1
   numbergridx2(f)=numbergridx2(f+1)-L;
end
%define numbergridy1
sayi=round(max(y)/((3/2)*(R+t)));
toplamsayi=2*sayi+1;
ortasayi=(1+toplamsayi)/2;
numbergridy1(ortasayi)=0;
for b=ortasayi+1:toplamsayi
numbergridy1(b)=numbergridy1(b-1)+(3/2)*(R+t);
end
for f=ortasayi:-1:1
   numbergridy1(f)=numbergridy1(f+1)-(3/2)*(R+t);
end
%define numbergridy2
sayi=round(max(y)/((3/2)*(R+t)));
toplamsayi=2*sayi+1;
ortasayi=(1+toplamsayi)/2;
numbergridy2(ortasayi)=0;
for b=ortasayi+1:toplamsayi
numbergridy2(b)=numbergridy2(b-1)+(3/2)*(R+t);
end
for f=ortasayi:-1:1
   numbergridy2(f)=numbergridy2(f+1)-(3/2)*(R+t);
end
% determine which type of grid is convenient for all points
for o=1:1:length(z)
   mindistance1=10000;
   for w=1:1:length(tubedata1)
      distance1=sqrt((x(o,1)-tubedata1(w,1))^2+(y(o,1)-
tubedata1(w,2))^2);
```

```
                if mindistance1>distance1
                    mindistance1=distance1;
                end
            end
            mindistance2=10000;
            for w=1:1:length(tubedata2)
                distance2=sqrt((x(o,1)-tubedata2(w,1))^2+(y(o,1)-
tubedata2(w,2))^2);
                if mindistance2>distance2
                    mindistance2=distance2;
                end
            end
            if mindistance1<mindistance2
              tubex(o) =2*(min(find((x(o,1)<numbergridx1)))-1)-1;
              tubey(o) =2*(min(find((y(o,1)<numbergridy1)))-1)-1;
            end

            if mindistance2<mindistance1
              tubex(o) =2*(min(find((x(o,1)<numbergridx2)))-1);
              tubey(o) =2*(min(find((y(o,1)<numbergridy2)))-1);
            end

        end




T2T_range = [];
T_range = [];
T_avg = [];

for j = 1:1:length(time)
Tnow = T(:,j);
   [tubemean tuberange] = grpstats(Tnow,{tubex' tubey'},{'mean','range'});
```

```
        T2T_range(j) = range(tubemean);
        T_range(j) = mean(tuberange);
        T_avg(j) = mean(Tnow);
        n_tubes = length(tubemean);
        exceldata(j,1)=time(j);
        exceldata(j,2)=T2T_range(j);
        exceldata(j,3)=T_range(j);
        exceldata(j,4)=T_avg(j);
        exceldata(1,6)=n_tubes;

    end
    xlswrite(excel2write,exceldata)
              end
  end
  end
  clear all
  path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle200 
Final DOE Results';
  path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Triangle200 
Final DOE Results Ex';


  DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal 
Results\Triangle200 Final Thermal DOE',1);

  files = dir(path);
  k = 0;
end
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % %
```

A.5 Evaluation code of COMSOL Hexagon Model Thermal Results (20 CPSI)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%

clear all
close all
path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon20 Final
DOE Results';
path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon20 Final
DOE Results Ex';


DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal
Results\Hexagon20 Final Thermal DOE',1);

files = dir(path);
k = 0;

for i = 1:length(files)

  if strfind(files(i).name,'.txt') > 0
    time = [];
    file2read = [path '\' files(i).name];
    excelID=files(i).name(1:strfind(files(i).name,'.txt')-1);
    excel2write= [path2 '\' num2str(excelID) '.xlsx'];
    runID =i-2;
    %data = dlmread(file2read,'',10,0);
    [data,delimiter,header] = importdata(file2read,' ',9);
```

```
times = char(data.textdata(9));
data = (data.data);
    start = strfind(times,'=') + 1;
    stop = strfind(times,'T') - 2;
        stop(1) = [];
        stop = [stop length(times)];
        time = [];
        for j = 1:length(stop)
            time(j) = str2double(times(start(j):stop(j)));
        end
        time = time';
    z = data(:,1) * 1000;
    x = data(:,2) * 1000;
        x = x - mean(x);
    y = data(:,3) * 1000;
        y = y - mean(y);
    temp = size(data);
    T = data(:,4:temp(2));

N = DOE(runID,2);
CPSI = DOE(runID,3)/25.4/25.4;
t = DOE(runID,4);
    switch N
        case 3
            L = sqrt(4/sqrt(3)/CPSI);
            s = L - sqrt(3)/2 * t;
            R = 1/sqrt(3) * s;
            A = sqrt(3)/4 * s^2;
            P = 3*s;
        case 4
            L = sqrt(1/CPSI);
            s = L - t;
            R = sqrt(2)/2 * s;
            A = s^2;
```

```
    P = 4*s;
case 6
    L = sqrt(2/3/sqrt(3)/CPSI);
    s = L - 1/sqrt(3) * t;
    R = s;
    A = 3*sqrt(3)/2 * s^2;
    P = 6*s;

    if DOE(runID,3) == 20

        % define type 1 center grid
        for q=11:1:12
            tubedata1(q,1)=(q-11)* 3*(R+t/sqrt(3));
            tubedata1(q,2)=0;
        end

        for q=11:1:9
            tubedata1(q,1)=(q-11)* 3*(R+t/sqrt(3));
            tubedata1(q,2)=0;
        end

        for q=9:4:17
            tubedata1(q,1)=(-2)* 3*(R+t/sqrt(3));
            tubedata1(q,2)=((q-9)/4)* sqrt(3)*(R+t/sqrt(3));
        end

        for q=9:-4:1
            tubedata1(q,1)=(-2)* 3*(R+t/sqrt(3));
            tubedata1(q,2)=((q-9)/4)* sqrt(3)*(R+t/sqrt(3));
        end

        for q=10:4:18
            tubedata1(q,1)=(-1)* 3*(R+t/sqrt(3));
            tubedata1(q,2)=((q-10)/4)* sqrt(3)*(R+t/sqrt(3));
```

```
end

for q=10:-4:2
   tubedata1(q,1)=(-1)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-10)/4)* sqrt(3)*(R+t/sqrt(3));
end

for q=11:4:19
   tubedata1(q,1)=0;
   tubedata1(q,2)=((q-11)/4)* sqrt(3)*(R+t/sqrt(3));
end

for q=11:-4:3
   tubedata1(q,1)=0;
   tubedata1(q,2)=((q-11)/4)* sqrt(3)*(R+t/sqrt(3));
end

for q=12:4:20
   tubedata1(q,1)=(1)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-12)/4)* sqrt(3)*(R+t/sqrt(3));
end

for q=12:-4:4
   tubedata1(q,1)=(1)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-12)/4)* sqrt(3)*(R+t/sqrt(3));
end
% define type 2 center grid
tubedata2(11,1)=(-3/2)*(R+t/sqrt(3));
tubedata2(11,2)=(sqrt(3)/2)*(R+t/sqrt(3));
tubedata2(12,1)=tubedata2(11,1)+3*(R+t/sqrt(3));
tubedata2(12,2)=(sqrt(3)/2)*(R+t/sqrt(3));
tubedata2(10,1)=tubedata2(11,1)-3*(R+t/sqrt(3));
tubedata2(10,2)=(sqrt(3)/2)*(R+t/sqrt(3));
```

```
for q=11:3:17
   tubedata2(q,1)=(-3/2)*(R+t/sqrt(3));
   tubedata2(q,2)=tubedata2(11,2)+((q-11)/3)* sqrt(3)*(R+t/sqrt(3));
end

for q=11:-3:2
   tubedata2(q,1)=(-3/2)*(R+t/sqrt(3));
   tubedata2(q,2)=tubedata2(11,2)+((q-11)/3)* sqrt(3)*(R+t/sqrt(3));
end

for q=12:3:18
   tubedata2(q,1)=tubedata2(11,1)+3*(R+t/sqrt(3));
   tubedata2(q,2)=tubedata2(12,2)+((q-12)/3)* sqrt(3)*(R+t/sqrt(3));
end

for q=12:-3:3
   tubedata2(q,1)=tubedata2(11,1)+3*(R+t/sqrt(3));
   tubedata2(q,2)=tubedata2(12,2)+((q-12)/3)* sqrt(3)*(R+t/sqrt(3));
end

for q=10:3:16
   tubedata2(q,1)=tubedata2(11,1)-3*(R+t/sqrt(3));
   tubedata2(q,2)=tubedata2(10,2)+((q-10)/3)* sqrt(3)*(R+t/sqrt(3));
end

for q=10:-3:1
   tubedata2(q,1)=tubedata2(11,1)-3*(R+t/sqrt(3));
   tubedata2(q,2)=tubedata2(10,2)+((q-10)/3)* sqrt(3)*(R+t/sqrt(3));
end

%define numbergridx1
sayi=round(max(x)/(3*(R+t/sqrt(3))))+1;
toplamsayi=2*sayi;
ortasayi=1+toplamsayi/2;
```

```
numbergridx1(ortasayi)=R+t/sqrt(3);
numbergridx1(ortasayi+1)=numbergridx1(ortasayi)+3*(R+t/sqrt(3));
for do=ortasayi-1:-1:1
   numbergridx1(do)=numbergridx1(do+1)-3*(R+t/sqrt(3));
end
%define numbergridx2
sayi=round(max(x)/(3*(R+t/sqrt(3))))+1;
toplamsayi=2*sayi-1;
ortasayi=(1+toplamsayi)/2;
numbergridx2(ortasayi)=(-3/2)*(R+t/sqrt(3))+(R+t/sqrt(3));
numbergridx2(ortasayi+1)=numbergridx2(ortasayi)+3*(R+t/sqrt(3));
numbergridx2(ortasayi-1)=numbergridx2(ortasayi)-3*(R+t/sqrt(3));
%define numbergridy1
sayi=round(max(y)/(sqrt(3)*(R+t/sqrt(3))))+1;
toplamsayi=2*sayi;
ortasayi=1+toplamsayi/2;
numbergridy1(ortasayi)=(R+t/sqrt(3))*(sqrt(3)/2);
for b=ortasayi+1:1:toplamsayi
numbergridy1(b)=numbergridy1(b-1)+sqrt(3)*(R+t/sqrt(3));
end
for do=ortasayi:-1:1
   numbergridy1(do)=numbergridy1(do+1)-sqrt(3)*(R+t/sqrt(3));
end
%define numbergridy2
sayi=round(max(y)/(sqrt(3)*(R+t/sqrt(3))))+1;
toplamsayi=2*sayi-1;
ortasayi=(1+toplamsayi)/2;
numbergridy2(ortasayi)=0;
for b=ortasayi+1:1:toplamsayi
numbergridy2(b)=numbergridy2(b-1)+sqrt(3)*(R+t/sqrt(3));
end
for do=ortasayi:-1:1
   numbergridy2(do)=numbergridy2(do+1)-sqrt(3)*(R+t/sqrt(3));
end
```

```
% determine which type of grid is convenient for all points
    for o=1:1:length(z)
      mindistance1=10000;
      for w=1:1:length(tubedata1)
        distance1=sqrt((x(o,1)-tubedata1(w,1))^2+(y(o,1)-
tubedata1(w,2))^2);
        if mindistance1>distance1
         mindistance1=distance1;
        end
      end
      mindistance2=10000;
      for w=1:1:length(tubedata2)
        distance2=sqrt((x(o,1)-tubedata2(w,1))^2+(y(o,1)-
tubedata2(w,2))^2);
        if mindistance2>distance2
          mindistance2=distance2;
        end
      end
      if mindistance1<mindistance2
       tubex(o) =2*(min(find((x(o,1)<numbergridx1)))-1)-1;
       tubey(o) =2*(min(find((y(o,1)<numbergridy1)))-1)-1;
      end

      if mindistance2<mindistance1
       tubex(o) =2*(min(find((x(o,1)<numbergridx2)))-1);
       tubey(o) =2*(min(find((y(o,1)<numbergridy2)))-1);
      end

    end

    T2T_range = [];
    T_range = [];
    T_avg = [];
```

```
                for j = 1:1:length(time)
                    Tnow = T(:,j);
                    [tubemean tuberange] = grpstats(Tnow,{tubex'
tubey'},{'mean','range'});
                    T2T_range(j) = range(tubemean);
                    T_range(j) = mean(tuberange);
                    T_avg(j) = mean(Tnow);
                    n_tubes = length(tubemean);
                    exceldata(j,1)=time(j);
                    exceldata(j,2)=T2T_range(j);
                    exceldata(j,3)=T_range(j);
                    exceldata(j,4)=T_avg(j);
                    exceldata(1,6)=n_tubes;
                end
               xlswrite(excel2write,exceldata)
              end
        end

%        figure(runID)
%        plot(x,y,'.')
    end
    clear all
    path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon20
Final DOE Results';
    path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon20
Final DOE Results Ex';
    DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal
Results\Hexagon20 Final Thermal DOE',1);
    files = dir(path);
    k = 0;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```

A.6 Evaluation code of COMSOL Hexagon Model Thermal Results (200 CPSI)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%

clear all
close all

path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon200 Final DOE Results';
path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon200 Final DOE Results Ex';


DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon200 Final Thermal DOE',1);

files = dir(path);
k = 0;

for i = 1:length(files)

    if strfind(files(i).name,'.txt') > 0
        time = [];
```

```
file2read = [path '\' files(i).name];
excelID=files(i).name(1:strfind(files(i).name,'.txt')-1);
excel2write= [path2 '\' num2str(excelID) '.xlsx'];
runID =i-2;
%data = dlmread(file2read,'',10,0);
[data,delimiter,header] = importdata(file2read,' ',9);
times = char(data.textdata(9));
data = (data.data);
    start = strfind(times,'=') + 1;
    stop = strfind(times,'T') - 2;
        stop(1) = [];
        stop = [stop length(times)];
        time = [];
        for j = 1:length(stop)
            time(j) = str2double(times(start(j):stop(j)));
        end
        time = time';
    z = data(:,1) * 1000;
    x = data(:,2) * 1000;
        x = x - mean(x);
    y = data(:,3) * 1000;
        y = y - mean(y);
    temp = size(data);
    T = data(:,4:temp(2));


N = DOE(runID,2);
CPSI = DOE(runID,3)/25.4/25.4;
t = DOE(runID,4);
    switch N
        case 3
            L = sqrt(4/sqrt(3)/CPSI);
            s = L - sqrt(3)/2 * t;
            R = 1/sqrt(3) * s;
            A = sqrt(3)/4 * s^2;
```

```
    P = 3*s;
case 4
    L = sqrt(1/CPSI);
    s = L - t;
    R = sqrt(2)/2 * s;
    A = s^2;
    P = 4*s;
case 6
    L = sqrt(2/3/sqrt(3)/CPSI);
    s = L - 1/sqrt(3) * t;
    R = s;
    A = 3*sqrt(3)/2 * s^2;
    P = 6*s;

    if DOE(runID,3) == 200

        % define type 1 center grid
        %71-80
        for q=76:1:80
            tubedata1(q,1)=(q-76)* 3*(R+t/sqrt(3));
            tubedata1(q,2)=0;
        end

        for q=76:-1:71
            tubedata1(q,1)=(q-76)* 3*(R+t/sqrt(3));
            tubedata1(q,2)=0;
        end
        % 71
        for q=71:10:141
            tubedata1(q,1)=(-5)* 3*(R+t/sqrt(3));
            tubedata1(q,2)=((q-71)/10)* sqrt(3)*(R+t/sqrt(3));
        end

        for q=71:-10:1
```

```
    tubedata1(q,1)=(-5)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-71)/10)* sqrt(3)*(R+t/sqrt(3));
end
%72
for q=72:10:142
    tubedata1(q,1)=(-4)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-72)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=72:-10:2
    tubedata1(q,1)=(-4)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-72)/10)* sqrt(3)*(R+t/sqrt(3));
end
%73
for q=73:10:143
    tubedata1(q,1)=(-3)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-73)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=73:-10:3
    tubedata1(q,1)=(-3)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-73)/10)* sqrt(3)*(R+t/sqrt(3));
end
%74
for q=74:10:144
    tubedata1(q,1)=(-2)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-74)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=74:-10:4
    tubedata1(q,1)=(-2)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-74)/10)* sqrt(3)*(R+t/sqrt(3));
end
%75
```

```
for q=75:10:145
   tubedata1(q,1)=(-1)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-75)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=75:-10:5
   tubedata1(q,1)=(-1)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-75)/10)* sqrt(3)*(R+t/sqrt(3));
end
%76
for q=76:10:146
   tubedata1(q,1)=(0)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-76)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=76:-10:6
   tubedata1(q,1)=(0)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-76)/10)* sqrt(3)*(R+t/sqrt(3));
end

%77
for q=77:10:147
   tubedata1(q,1)=(1)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-77)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=77:-10:7
   tubedata1(q,1)=(1)* 3*(R+t/sqrt(3));
   tubedata1(q,2)=((q-77)/10)* sqrt(3)*(R+t/sqrt(3));
end

%78
for q=78:10:148
   tubedata1(q,1)=(2)* 3*(R+t/sqrt(3));
```

```
    tubedata1(q,2)=((q-78)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=78:-10:8
    tubedata1(q,1)=(2)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-78)/10)* sqrt(3)*(R+t/sqrt(3));
end

%79
for q=79:10:149
    tubedata1(q,1)=(3)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-79)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=79:-10:9
    tubedata1(q,1)=(3)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-79)/10)* sqrt(3)*(R+t/sqrt(3));
end

%80

for q=80:10:150
    tubedata1(q,1)=(4)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-80)/10)* sqrt(3)*(R+t/sqrt(3));
end

for q=80:-10:10
    tubedata1(q,1)=(4)* 3*(R+t/sqrt(3));
    tubedata1(q,2)=((q-80)/10)* sqrt(3)*(R+t/sqrt(3));
end
% define type 2 center grid
%64-72
tubedata2(68,1)=(-3/2)*(R+t/sqrt(3));
tubedata2(68,2)=(sqrt(3)/2)*(R+t/sqrt(3));
```

```
for q=68:1:72
   tubedata2(q,1)=(q-68)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=(sqrt(3)/2)*(R+t/sqrt(3));;
end

for q=68:-1:64
   tubedata2(q,1)=(q-68)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=(sqrt(3)/2)*(R+t/sqrt(3));;
end
% 64
for q=64:9:118
   tubedata2(q,1)=(-4)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-64)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

for q=64:-9:1
   tubedata2(q,1)=(-4)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-64)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end
%65
for q=65:9:119
   tubedata2(q,1)=(-3)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-65)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

for q=65:-9:2
   tubedata2(q,1)=(-3)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-65)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end
%66
for q=66:9:120
   tubedata2(q,1)=(-2)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-66)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
```

```
end

for q=66:-9:3
   tubedata2(q,1)=(-2)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-66)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end
%67
for q=67:9:121
   tubedata2(q,1)=(-1)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-67)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

for q=67:-9:4
   tubedata2(q,1)=(-1)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-67)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end
%68
for q=68:9:122
   tubedata2(q,1)=(0)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-68)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

for q=68:-9:5
   tubedata2(q,1)=(0)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-68)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end
%69
for q=69:9:123
   tubedata2(q,1)=(1)* 3*(R+t/sqrt(3))+tubedata2(68,1);
   tubedata2(q,2)=((q-69)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

for q=69:-9:6
   tubedata2(q,1)=(1)* 3*(R+t/sqrt(3))+tubedata2(68,1);
```

```
    tubedata2(q,2)=((q-69)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

%70
for q=70:9:124
    tubedata2(q,1)=(2)* 3*(R+t/sqrt(3))+tubedata2(68,1);
    tubedata2(q,2)=((q-70)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

for q=70:-9:7
    tubedata2(q,1)=(2)* 3*(R+t/sqrt(3))+tubedata2(68,1);
    tubedata2(q,2)=((q-70)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

%71
for q=71:9:125
    tubedata2(q,1)=(3)* 3*(R+t/sqrt(3))+tubedata2(68,1);
    tubedata2(q,2)=((q-71)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

for q=71:-9:8
    tubedata2(q,1)=(3)* 3*(R+t/sqrt(3))+tubedata2(68,1);
    tubedata2(q,2)=((q-71)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

%72
for q=72:9:126
    tubedata2(q,1)=(4)* 3*(R+t/sqrt(3))+tubedata2(68,1);
    tubedata2(q,2)=((q-72)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
end

for q=72:-9:9
    tubedata2(q,1)=(4)* 3*(R+t/sqrt(3))+tubedata2(68,1);
    tubedata2(q,2)=((q-72)/9)* sqrt(3)*(R+t/sqrt(3))+tubedata2(68,2);
```

```
end

%define numbergridx1
sayi=round(max(x)/(3*(R+t/sqrt(3))))+1;
toplamsayi=2*sayi;
ortasayi=1+toplamsayi/2;
numbergridx1(ortasayi)=R+t/sqrt(3);
for do=ortasayi+1:1:toplamsayi
numbergridx1(do)=numbergridx1(do-1)+3*(R+t/sqrt(3));
end
for do=ortasayi-1:-1:1
   numbergridx1(do)=numbergridx1(do+1)-3*(R+t/sqrt(3));
end
%define numbergridx2
sayi=round(max(x)/(3*(R+t/sqrt(3))))+1;
toplamsayi=2*sayi-1;
ortasayi=(1+toplamsayi)/2;
numbergridx2(ortasayi)=(-3/2)*(R+t/sqrt(3))+(R+t/sqrt(3));
for do=ortasayi+1:1:toplamsayi
numbergridx2(do)=numbergridx2(do-1)+3*(R+t/sqrt(3));
end
for do=ortasayi-1:-1:1
numbergridx2(do)=numbergridx2(do+1)-3*(R+t/sqrt(3));
end
%define numbergridy1
sayi=round(max(y)/(sqrt(3)*(R+t/sqrt(3))))+1;
toplamsayi=2*sayi;
ortasayi=1+toplamsayi/2;
numbergridy1(ortasayi)=(R+t/sqrt(3))*(sqrt(3)/2);
for b=ortasayi+1:1:toplamsayi
numbergridy1(b)=numbergridy1(b-1)+sqrt(3)*(R+t/sqrt(3));
end
for do=ortasayi-1:-1:1
   numbergridy1(do)=numbergridy1(do+1)-sqrt(3)*(R+t/sqrt(3));
```

```
                end
                %define numbergridy2
                sayi=round(max(y)/(sqrt(3)*(R+t/sqrt(3))))+1;
                toplamsayi=2*sayi-1;
                ortasayi=(1+toplamsayi)/2;
                numbergridy2(ortasayi)=0;
                for b=ortasayi+1:1:toplamsayi
                numbergridy2(b)=numbergridy2(b-1)+sqrt(3)*(R+t/sqrt(3));
                end
                for do=ortasayi-1:-1:1
                   numbergridy2(do)=numbergridy2(do+1)-sqrt(3)*(R+t/sqrt(3));
                end
            % determine which type of grid is convenient for all points
                for o=1:1:length(z)
                   mindistance1=10000;
                   for w=1:1:length(tubedata1)
                      distance1=sqrt((x(o,1)-tubedata1(w,1))^2+(y(o,1)-
tubedata1(w,2))^2);
                      if mindistance1>distance1
                         mindistance1=distance1;
                      end
                   end
                   mindistance2=10000;
                   for w=1:1:length(tubedata2)
                      distance2=sqrt((x(o,1)-tubedata2(w,1))^2+(y(o,1)-
tubedata2(w,2))^2);
                      if mindistance2>distance2
                         mindistance2=distance2;
                      end
                   end
                   if mindistance1<mindistance2
                     tubex(o) =2*(min(find((x(o,1)<numbergridx1)))-1)-1;
                     tubey(o) =2*(min(find((y(o,1)<numbergridy1)))-1)-1;
                   end
```

```
                if mindistance2<mindistance1
                  tubex(o) =2*(min(find((x(o,1)<numbergridx2)))-1);
                  tubey(o) =2*(min(find((y(o,1)<numbergridy2)))-1);
                end

            end

            T2T_range = [];
            T_range = [];
            T_avg = [];

            for j = 1:1:length(time)
            Tnow = T(:,j);
                [tubemean tuberange] = grpstats(Tnow,{tubex'
tubey'},{'mean','range'});
                T2T_range(j) = range(tubemean);
                T_range(j) = mean(tuberange);
                T_avg(j) = mean(Tnow);
                n_tubes = length(tubemean);
                exceldata(j,1)=time(j);
                exceldata(j,2)=T2T_range(j);
                exceldata(j,3)=T_range(j);
                exceldata(j,4)=T_avg(j);
                exceldata(1,6)=n_tubes;
            end
            xlswrite(excel2write,exceldata)
        end
    end

%       figure(runID)
%       plot(x,y,'.')
  end
  clear all
```

```
    path = 'P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon200
Final DOE Results';
    path2= 'P:\STH\09 September Week 2\Final DOE Thermal Results\Hexagon200
Final DOE Results Ex';
    DOE=xlsread('P:\STH\09 September Week 2\Final DOE Thermal
Results\Hexagon200 Final Thermal DOE',1);
    files = dir(path);
    k = 0;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```

A.7 Comparison code of Oxygen Desorption Reaction Conversion Results of Silicon Carbide and Cordierite-1

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%

clc;
clear all;
close all;
path = 'C:\Users\HP\Desktop\Silicon Carbide';
path2 = 'C:\Users\HP\Desktop\compare.xlsx';
```

```
files = dir(path);

bigdata=[];


for id=1:length(files)

if strfind(files(id).name,'.xlsx') >0
    file2read = [path '\' files(id).name];
    runID=files(id).name(4:strfind(files(id).name,'.xlsx')-1);
    data=xlsread(file2read,1);
    for i=1:3
        exceldata(i,1)=str2num(runID);
        exceldata(i,2)=data(i,1);
        exceldata(i,3)=data(i,2);
    end

    for i=4:length(data)
        if data(i,1)==0
            break
        end
        exceldata(i,1)=str2num(runID);
        exceldata(i,2)=data(i,1);
        exceldata(i,3)=data(i,2);

    end

   bigdata=[bigdata;exceldata];
end
exceldata=[];
data=[];
end
xlswrite(path2,bigdata)
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.8 Comparison code of Oxygen Desorption Reaction Conversion Results of Silicon Carbide and Cordierite-2

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc;
clear all;
close all;
path = 'C:\Users\HP\Desktop\Cordierite';
path2 = 'C:\Users\HP\Desktop\compare2.xlsx';


files = dir(path);

bigdata=[];


for id=1:length(files)

if strfind(files(id).name,'.xlsx') >0
   file2read = [path '\' files(id).name];
```

```
    runID=files(id).name(2:strfind(files(id).name,'.xlsx')-1);
    data=xlsread(file2read,1);
    for i=1:3
        exceldata(i,1)=str2num(runID);
        exceldata(i,2)=data(i,1);
        exceldata(i,3)=data(i,2);
    end

    for i=4:length(data)
        if data(i,1)==0
            break
        end
        exceldata(i,1)=str2num(runID);
        exceldata(i,2)=data(i,1);
        exceldata(i,3)=data(i,2);


    end

  bigdata=[bigdata;exceldata];
end
exceldata=[];
data=[];
end
xlswrite(path2,bigdata)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```