



POLITECNICO DI TORINO

CORSO DI LAUREA MAGISTRALE  
IN INGEGNERIA MECCANICA

**Performance improvement of a latent heat  
TES system through shape and topology  
optimization**

*Candidato*

*Mattia Degrandi*

Dipartimento di Ingegneria Meccanica e Aerospaziale

*Relatore*

*Prof. Vittorio Verda*

Dipartimento Energia

October 4, 2020





# **Abstract**

Thermal Energy Storage (TES) systems using Phase Change Material (PCM) are gaining more and more importance to provide a higher heat storage capacity.

This thesis models and studies the design optimization of a Latent Heat TES system to rise the heat transfer developing an improved fin configuration.

This work involves two tasks. First, the Shape Optimization (SO) of a fin structure is examined. Then, the Topology Optimization (TO) of a TES unit is analyzed. Finally, a comparison between shape and topology optimized fins is carried out.

# Contents

<b>1</b>	<b>Theoretical Framework</b>	<b>1</b>
1.1	Overview of optimization . . . . .	1
1.2	Single-Objective Convex Optimization . . . . .	3
1.3	Three computational optimization techniques . . . . .	3
1.4	Stopping criteria . . . . .	5
<b>2</b>	<b>Shape Optimization problem formulation</b>	<b>6</b>
2.1	Single-Objective Shape Optimization . . . . .	6
2.2	Design of Experiments . . . . .	6
2.2.1	Engineering experiments . . . . .	8
2.3	Golden Section Search . . . . .	8
2.3.1	Pseudo-code of the GSS . . . . .	10
2.4	Quadratic Fit Search . . . . .	10
2.4.1	Pseudo-code of the QFS . . . . .	12
<b>3</b>	<b>Thermal diffusive SO problem</b>	<b>13</b>
3.1	PCM for LHTESS . . . . .	13
3.2	Simulation workflow . . . . .	14
3.3	Stationary heat conduction problem . . . . .	15
3.3.1	Geometry and boundary conditions . . . . .	15
3.3.2	Governing equation . . . . .	17
3.3.3	Computational implementation . . . . .	19
3.4	Response Surface Method . . . . .	19
3.4.1	Implementation of the MATLAB program . . . . .	21
3.5	Steady-state single-objective SO with one DOF . . . . .	24
3.6	Steady-state single-objective SO with two DOF . . . . .	26
3.7	Time-dependent heat conduction problem . . . . .	30
3.7.1	Geometry and boundary conditions . . . . .	30
3.7.2	Governing equation . . . . .	30
3.7.3	Computational implementation . . . . .	34
3.8	Transient-state single-objective SO with two DOF . . . . .	34

<b>4</b>	<b>Topology Optimization problem formulation</b>	<b>40</b>
4.1	Topology optimization . . . . .	40
4.2	Density-based approach . . . . .	41
4.2.1	SIMP model for TO . . . . .	42
4.3	Regularization . . . . .	43
4.4	Discrete TO problem formulation . . . . .	44
4.4.1	TO of heat conduction problem . . . . .	46
4.5	How to solve the TO problem . . . . .	48
<b>5</b>	<b>Thermal diffusive TO model</b>	<b>51</b>
5.1	Simulation workflow . . . . .	51
5.2	Stationary heat conduction TO problem . . . . .	52
5.2.1	Geometry and boundary conditions . . . . .	52
5.2.2	Modified governing equations . . . . .	53
5.2.3	Mesh independence study . . . . .	55
5.2.4	TO model . . . . .	56
5.2.5	Triple regularization strategy . . . . .	58
5.2.6	Resulting TO model . . . . .	61
5.2.7	Computational implementation . . . . .	61
5.3	Simulation results . . . . .	62
5.4	Presentation of the case studies . . . . .	66
5.4.1	$\gamma_{max}$ sensitivity study . . . . .	67
5.4.2	$\Gamma$ sensitivity study . . . . .	71
5.5	Time-dependent heat conduction TO problem . . . . .	75
5.5.1	Modified governing equations . . . . .	75
5.5.2	TO model . . . . .	76
5.5.3	Double regularization strategy . . . . .	77
5.5.4	Computational implementation . . . . .	78
5.6	Simulation results . . . . .	78
5.7	Conclusion . . . . .	79
<b>6</b>	<b>Topology optimized design validation</b>	<b>81</b>
6.1	Stationary validation protocol . . . . .	81
6.1.1	Finned geometry generation . . . . .	81
6.1.2	Thermal field modelling . . . . .	82
6.2	Simulation results . . . . .	83
6.3	Sequential coupling of Topology and Shape Optimization . . . . .	84
6.3.1	Stationary heat conduction SO problem . . . . .	85
6.4	Time-dependent validation protocol . . . . .	88
6.4.1	Finned geometry generation . . . . .	88
6.4.2	Thermal field modelling . . . . .	89

6.5	Simulation results . . . . .	90
6.5.1	Time-dependent heat conduction TSO problem . . . .	91
6.6	Robustness of the TSO technique . . . . .	94
6.7	Conclusion . . . . .	95

# List of Figures

1.1	Block diagram of a design optimization scheme. . . . .	4
2.1	General model of a process/system. . . . .	7
2.2	Representation of the golden ratio using a line segment. . . . .	8
3.1	Block diagram of an engineering design. . . . .	14
3.2	Thermal energy storage unit. . . . .	15
3.3	Solution domain. . . . .	15
3.4	(a) Dirichlet BC (b) Neumann BC (c) Volumetric heat generation. . . . .	17
3.5	Geometric model with mesh details. . . . .	19
3.6	Block diagram of the DoE methodology. . . . .	20
3.7	Fin geometry parameters. . . . .	21
3.8	(a) Response surface (b) Isolevel curves. . . . .	24
3.9	(a) GSS to minimize $f(L, \theta)$ (b) Convergence history of the solution. . . . .	26
3.10	(a) Temperature profile (b) Temperature contours. . . . .	26
3.11	(a) QFS to minimize $f(L, \theta)$ (b) Convergence history of the solution. . . . .	30
3.12	Initial temperature. . . . .	31
3.13	PCM specific heat capacity model. . . . .	32
3.14	Liquid mass fraction. . . . .	33
3.15	(a) GSS to maximize $f(\theta)$ (b) Convergence history of the solution . . . . .	37
3.16	Temperature fields. (a) $t = 600[s]$ (b) $t = 1800[s]$ (c) $t = 3600[s]$ 37	
3.17	Temperature contours. (a) $t = 600[s]$ (b) $t = 1800[s]$ (c) $t = 3600[s]$ . . . . .	38
3.18	Liquid fraction fields. (a) $t = 600[s]$ (b) $t = 1800[s]$ (c) $t = 3600[s]$ . . . . .	38
3.19	(a) $T_{avg}$ evolution (b) $E_{tot}$ released . . . . .	38
4.1	Pseudo-density variable. . . . .	41

4.2	SIMP penalty functions . . . . .	42
4.3	Heaviside step function. . . . .	45
4.4	Block diagram of the GCMMA scheme. . . . .	50
5.1	Flowchart of the TO. . . . .	53
5.2	Quarter circle design domain. . . . .	54
5.3	(a) Dirichlet BC (b) Neumann BC (c) Volumetric heat generation. . . . .	54
5.4	Mesh scheme. (a) $h_{max}$ (b) $h_{int}$ (c) $h_{min}$ . . . . .	56
5.5	SIMP interpolation model. (a) $k_{SIMP}(s)$ (b) $Q_{SIMP}(s)$ . . . . .	59
5.6	Mesh independence analysis. (a) $f(s)_{final}$ (b) $T_{avg}$ . . . . .	64
5.7	First simulation. (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	64
5.8	First simulation. (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function . . . . .	64
5.9	Second simulation. (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	65
5.10	Second simulation. (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function . . . . .	65
5.11	Third simulation. (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	65
5.12	Third simulation. (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function . . . . .	65
5.13	Projected design field. (a) First simulation (b) Second simulation (c) Third simulation . . . . .	66
5.14	First simulation at constant $\gamma_{max,2}$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	67
5.15	First simulation at constant $\gamma_{max,2}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function . . . . .	67
5.16	Second simulation at constant $\gamma_{max,2}$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	68
5.17	Second simulation at constant $\gamma_{max,2}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function . . . . .	68
5.18	Third simulation at constant $\gamma_{max,2}$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	68
5.19	Third simulation at constant $\gamma_{max,2}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function . . . . .	68
5.20	First simulation at constant $\gamma_{max,3}$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	69

5.21	First simulation at constant $\gamma_{max,3}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function	69
5.22	Second simulation at constant $\gamma_{max,3}$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	69
5.23	Second simulation at constant $\gamma_{max,3}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function . . . . .	69
5.24	Third simulation at constant $\gamma_{max,3}$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	70
5.25	Third simulation at constant $\gamma_{max,3}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function . . . . .	70
5.26	(a) $f(s)_{final}$ (b) $T_{avg}$ . . . . .	70
5.27	First simulation at constant $\Gamma_2$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	71
5.28	First simulation at constant $\Gamma_2$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function	72
5.29	Second simulation at constant $\Gamma_2$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	72
5.30	Second simulation at constant $\Gamma_2$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function	72
5.31	Third simulation at constant $\Gamma_2$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	72
5.32	Third simulation at constant $\Gamma_2$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function	73
5.33	First simulation at constant $\Gamma_3$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	73
5.34	First simulation at constant $\Gamma_3$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function	73
5.35	Second simulation at constant $\Gamma_3$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	73
5.36	Second simulation at constant $\Gamma_3$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function	74
5.37	Third simulation at constant $\Gamma_3$ . (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	74
5.38	Third simulation at constant $\Gamma_3$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function	74
5.39	(a) $f(s)_{final}$ (b) $T_{avg}$ . . . . .	75
5.40	SIMP interpolation model. (a) $\rho_{SIMP}(s)$ (b) $C_{p,SIMP}(s)$ . . . .	77
5.41	First simulation. (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	78

5.42	First simulation. (a) Projected variable (b) Whole layout (c) Target function . . . . .	79
5.43	Second simulation. (a) Design variable (b) Temperature field (c) Temperature contour . . . . .	79
5.44	Second simulation. (a) Projected variable (b) Whole layout (c) Target function . . . . .	79
6.1	Filter applied to (a) $s$ (b) $s_p$ . . . . .	82
6.2	(a) Mesh part (b) Geometric configuration . . . . .	82
6.3	(a) Dirichlet BC (b) Neumann BC (c) Volumetric heat source . . . . .	83
6.4	Mesh scheme . . . . .	83
6.5	(a) Temperature distribution (b) Isotherms . . . . .	84
6.6	(a) Free Shape Domain (b) Free Shape Boundary . . . . .	86
6.7	(a) Optimized shape (b) Target function . . . . .	86
6.8	Filter applied to (a) $s$ (b) $s_p$ . . . . .	88
6.9	(a) Mesh part (b) Collapse Edges (c) Geometric configuration . . . . .	88
6.10	(a) Dirichlet BC (b) Neumann BC (c) Initial temperature . . . . .	89
6.11	Mesh scheme . . . . .	89
6.12	Temperature fields. (a) $t = 600[s]$ (b) $t = 1800[s]$ (c) $t = 3600[s]$ . . . . .	90
6.13	Temperature contours. (a) $t = 600[s]$ (b) $t = 1800[s]$ (c) $t = 3600[s]$ . . . . .	90
6.14	(a) Free Shape Domain (b) Free Shape Boundary . . . . .	91
6.15	(a) Optimized shape (b) Target function . . . . .	92
6.16	Temperature fields. (a) $t = 600[s]$ (b) $t = 1800[s]$ (c) $t = 3600[s]$ . . . . .	93
6.17	Temperature contours. (a) $t = 600[s]$ (b) $t = 1800[s]$ (c) $t = 3600[s]$ . . . . .	93
6.18	Liquid fraction fields. (a) $t = 600[s]$ (b) $t = 1800[s]$ (c) $t = 3600[s]$ . . . . .	93
6.19	(a) $T_{avg}$ evolution (b) $E_{tot}$ released . . . . .	94
6.20	(a) Optimized shape for $\gamma_{max,2}$ (b) Optimized shape for $\gamma_{max,3}$ . . . . .	95



# List of Tables

3.1	Geometry parameters . . . . .	16
3.2	Boundary conditions . . . . .	17
3.3	Materials properties . . . . .	18
3.4	The best design solution . . . . .	26
3.5	The best design solution . . . . .	29
3.6	Boundary conditions . . . . .	30
3.7	Materials properties . . . . .	31
3.8	The best design solution . . . . .	37
5.1	Geometric parameters . . . . .	52
5.2	Boundary conditions . . . . .	54
5.3	Thermophysical properties . . . . .	54
5.4	Mesh parameters . . . . .	56
5.5	Model parameters . . . . .	61
5.6	Simulation (a) First (b) Second (c) Third . . . . .	61
5.7	Simulation (a) First (b) Second (c) Third . . . . .	62
5.8	Simulation (a) First (b) Second (c) Third . . . . .	63
5.9	Fin cross-sectional area requirement . . . . .	67
5.10	$T_{avg}$ varying values of $\gamma_{max}$ . . . . .	70
5.11	Scale factors . . . . .	71
5.12	$T_{avg}$ varying values of $\Gamma$ . . . . .	74
5.13	Model parameters . . . . .	77
5.14	Simulation (a) First (b) Second . . . . .	78
6.1	Numerical solutions . . . . .	84
6.2	Fins surface area . . . . .	85
6.3	Optimization factors . . . . .	86
6.4	Fins surface area . . . . .	87
6.5	TSO versus SO . . . . .	87
6.6	Numerical solutions . . . . .	90
6.7	Optimization factors . . . . .	91
6.8	Fins surface area . . . . .	92

6.9	TSO versus SO . . . . .	92
6.10	Fin surface area . . . . .	95

# Chapter 1

## Theoretical Framework

### 1.1 Overview of optimization

The focus of this chapter is to highlight the main themes of engineering design. The topics covered by this section are provided by Tobe (2015) [1], Pluke (2020) [2], Verbart et al. (2015) [21], Rojas Labanda (2015) [22] and lecture notes of Sarkar (2018) [3].

When the engineering community talks about optimization, it is typically referring to design optimization. The power of this engineering tool is its ability to discover through evaluation of a set of design alternatives the most effective or functional possible design, which best meets prescribed design requirements. Consequently, design optimization is a very important field, because it helps design engineers to make better decisions.

From a mathematical standpoint, design optimization is the iterative process of varying the parameters  $s_k$  to find the maximum or minimum of some attributes  $f_i(s_k)$  subjected to certain specified limitations  $c_j(s_k)$ . These basic elements are described below.

- **Design variables**  $s_k$  are the parameters, that may be changed in an optimization procedure for searching the optimal design. These variables are generally unknown.
- **Objective functions**  $f_i(s_k)$  are the performance targets to meet. To ensure good designs and achieve all the desired goals, correct modelling and simulation of components, systems and devices is essential.
- **Constraint functions**  $c_j(s_k)$  are the design requirements to satisfy.

The design constraints can be:

- **Geometric** such as restrictions on the width or height, perimeter or surface.
- **Performance** like to minimize the average or maximum temperature.

Mathematically, a multi-objective optimization model can be easily stated as

$$\begin{cases} \max \text{ or } \min & f_i(s_k), i, k = 1, \dots, n \\ \text{subject to} & c_j(s_k), j = 1, \dots, n \\ & s_k \in S \end{cases} \quad (1.1)$$

Solving this problem the following solutions are found:

$$\begin{cases} s_k^* & \text{such that } f_i(s_k^*) \geq \text{ or } \leq f_i(s_k) \\ & s_k \in S \end{cases} \quad (1.2)$$

where

- $s_k$  is the vector of design variables.
- $f_i(s_k)$  are the set of design objectives to optimize.
- $c_j(s_k)$  are the design constraints of the model.
- $s_k^*$  is the optimal set of values of independent variables.
- $f_i(s_k^*)$  are the optimum targets reached, which are known as optimal Pareto front.

It should check the existence of a set of optimal solutions, since  $s_k^*$  cannot even exist.

The biggest difference between single- and multi-objective optimization comes from the fact that in the first case only one function will be maximized or minimized, instead of in the second one design engineers wish to discover the set of values of independent variables, which maximize or minimize all the functions. An implication of this last type of problem is that these functions may conflict with each other. Therefore, design engineers must balance design objectives to find the best trade-off, because when a function is optimized, others can worsen.

## 1.2 Single-Objective Convex Optimization

In this section, it will be described the mathematical formulation of a single-objective convex optimization problem as discussed in Verbart et al. (2015) [21] and Rojas Labanda (2015) [22].

**Single-Objective Convex Optimization (SOCO)** involves the minimization of a convex function, which may or may not be subject to specific requirements. Generally, CO plays one of the major roles in most engineering applications, since aiming to minimize a convex function, over a convex set of values, only one global minimum can exist.

If some restrictions are imposed, it is asked to solve a constrained SOCO. This kind of problem can be written in general form as

$$\begin{cases} \min & f(s) \\ \text{subject to} & c_j(s), j = 1, \dots, n \\ & s \in S \end{cases} \quad (1.3)$$

Here,  $c_j(s)$  is the vector of specified design requirements Eq. 1.4, that might be  $g(s) >, < \text{ or } = 0$ , i.e. inequality or equality constraints.

$$g(s) = \{g_1(s), \dots, g_n(s)\}^T \quad (1.4)$$

Moreover, the set of all feasible values  $\Sigma$  of the problem may be expressed in the following form

$$\Sigma = \{s \in S | c_j(s) \leq 0, j = 1, \dots, n\} \quad (1.5)$$

thus  $\Sigma \neq \emptyset$ .

When no limitations are established, it is asked to solve an unconstrained SOCO. Formulation of this type of problem is simplified

$$\left\{ \min \quad f(s) \right. \quad (1.6)$$

The basic difference between these two types of minimization problems are the restrictions  $c_j(s)$ : in a constrained problem  $\Sigma \subset \mathbb{R}^n$  and conversely in an unconstrained one  $\Sigma \equiv \mathbb{R}^n$ .

## 1.3 Three computational optimization techniques

This section provides a general overview of three numerical methods used to perform a design optimization. These contributions can be found in Tobe (2015) [1] and Pluke (2020) [2].

Engineers and scientists classify methods for design optimization into three categories:

- **Size Optimization** consists in seeking the best tuning of size parameters varying them.
- **Shape Optimization (SO)** means to optimize the shape adjusting the surface shape.
- **Topology Optimization (TO)** seeks to optimize material layout working on addition and subtraction of a fictitious material.

The purpose of these numerical methodologies is to generate a model with better performance. To reach the best possible output an iterative process addresses the solution towards a resulting layout meeting the defined performance targets and satisfying all constraint requirements. Fig. 1.1 provides a block diagram of a generic design optimization procedure.

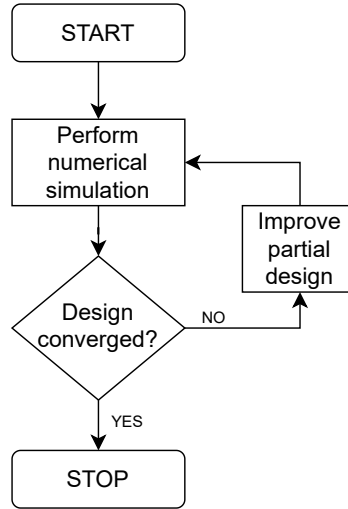


Figure 1.1: Block diagram of a design optimization scheme.

Since this thesis focuses on single-objective S- and TO, the greatest differences between these two types of mathematical approach are presented below.

- In **SO** one or more design variables describe the user-created geometric features. In contrast, **TO** uses only one fictitious density variable.

- In **SO** a given initial shape is progressively modified not exceeding the upper and lower bound of deformation allowed, i.e. the restrictions imposed on the geometric structure. Conversely, in **TO** there are no topological limitations on the resulting layout.
- As a consequence of the previous point, the final shape reached with **SO** is not completely free and thus SO cannot explore all possible designs. Instead of **TO** offers remarkable potential to identify new conceptual designs, not being dependent on the initial geometric configuration.

## 1.4 Stopping criteria

In this section, it is provided an explanation of the necessity of two stopping criteria to terminate an iterative process converging towards the desired goal. The two termination conditions that will be implemented in the algorithms as a stopping test are:

1. **Control of the increment:** the iterative process stops when the inequality 1.7 is satisfied:

$$|x^{n+1} - x^n| < \epsilon \quad (1.7)$$

2. **Control of the number of loop iterations:** the iterative scheme terminates as soon as the inequality 1.8 is fulfilled:

$$n > N_{max} \quad (1.8)$$

Here,  $x^n$  is a variable updated in each iteration,  $\epsilon$  is a fixed tolerance,  $N_{max}$  is the maximum allowed number of iterations the algorithm may perform.

The implementation of the stopping criteria ensures that

- When the calculated error becomes less than a tolerance defined a-priori a satisfactory output is obtained.
- The loops cannot continue to indefinitely iterate and hence reach the convergence within a finite number of steps.

In addition to checking of those conditions could be useful to check the speed of convergence, generally called **convergence rate**. Evaluation of this indicator is essential to quantify how much quickly the algorithm approaches the global optimum and thus to estimate efficiency of the code. If the convergence rate is not sufficiently fast some modifications of the script should be considered.

# Chapter 2

## Shape Optimization problem formulation

### 2.1 Single-Objective Shape Optimization

SO is a mathematical approach that optimizes surface shape within a given design space, for a prescribed set of boundary conditions such that the resulting layout achieves a desired performance target. The design variables can be chosen from shape parameters such as diameter of hole, rotation angle about axis, thickness along component height or any other measure.

As mentioned in Sec. 1.1, SO is considered a success if it gradually converges to the optimal shape, while fulfilling all the constraints prescribed by design requirements.

### 2.2 Design of Experiments

In this section, it will be presented the design and analysis of experiments referring to the lecture notes of Maiti (2017) [4] and Sarkar (2018) [3]. In the following Subsec. 2.2.1 it will further extend this discussion.

Design engineers typically perform **Design of Experiments (DoE)** to explore model performance with minimum simulation effort, to better understand the design space. Using this kind of approach many relevant design information can be captured.

A generic DoE procedure consists in evaluating a process/system response, i.e. output parameter, which is affected by many independent variables, i.e. input parameters.

An experiment is carried out to find out the functional relationship between inputs and output, to know how much input variables  $x, z$  will in-



fluence the value of output variable  $y$ .

$$y = f(x, z) \quad (2.1)$$

This mathematical model may be a linear or non-linear equation.

Fig. 2.1 represents a generic representation of a process/system model to simplify physical reality, which can be extremely complex.

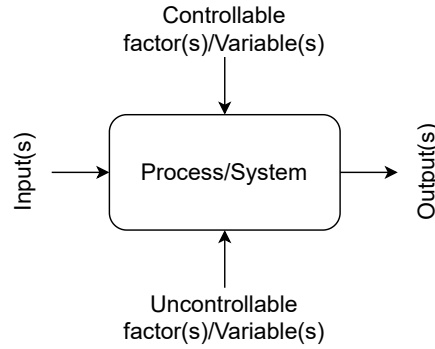


Figure 2.1: General model of a process/system.

Output is the key indicator that is computed to judge a process/system performance. Controllable factors are those whose levels can be varied in each experiment to seek the best possible settings that allow to get an output close to the desired objective. Conversely, uncontrollable factors are those whose values cannot be manipulated and hence their effect should be minimized.

To investigate the output response and well depict the process/system behaviour, variations of the model inputs are accomplished by means of a series of tests. The purpose of this way to proceed is to determine the most influential input variables on the output response. These experiments are the key factor that plays a dominant role in quantifying the effect of the inputs changes on the desired outcome and finding out the region in which the response is least sensitive to those variations. Mathematical achievement of this region enables to decrease the variability of output.

Discovering a clear ranking of the inputs according to their influence on the output response, it is possible to drop the number of independent variables, set the most influential inputs and address simulations towards shorter development time and guarantee the validity of numerical results.

### 2.2.1 Engineering experiments

Experimentation is extremely important in several different engineering activities for the following reasons:

- New products and/or processes design/development.
- Existing product and/or processes performances improvement.
- Alternative designs evaluation.

To satisfy product requirements and/or process specifications the following procedure is used:

1. First of all, levels and ranges of the input(s) are chosen.
2. Next, it is necessary to select the output(s).
3. Then, experimental designs are implemented.
4. Subsequently, experimental outcomes are analyzed.
5. Finally, conclusions are drawn and then dominant factors are discovered.

## 2.3 Golden Section Search

This section is concerned with the description of the **Golden Section Search (GSS)** method provided in the book of Giordano et al. (2013) [5], lecture slides of Andersson et al. (2014) [6] and Heath (2002) [7].

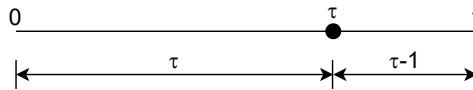


Figure 2.2: Representation of the golden ratio using a line segment.

Before starting to explain how to implement this algorithm, it should be clarified what is meant when the expression **golden ratio** is used. If a line segment, whose length is taken equal to 1, is divided into two unequal subparts of length  $\tau$  and  $1 - \tau$ , respectively, thus it is possible to write the following equation:

$$\frac{1}{\tau} = \frac{\tau}{1 - \tau} \Rightarrow \tau^2 + \tau - 1 = 0 \quad (2.2)$$

Eq. 2.2 yields the two roots

$$\tau_1 = \frac{\sqrt{5} - 1}{2} \approx 0.618 \quad \text{and} \quad \tau_2 = \frac{-\sqrt{5} - 1}{2} \approx 0.382 \quad (2.3)$$

Only the positive root  $\tau_1 \in (0, 1)$ . This constant is called golden ratio.

The GSS numerical method is only applied when

- It is known that a global maximum or minimum exists within a specified interval.
- The real-valued objective function  $f(x)$  must be unimodal, i.e. it has only one peak (maximum) or valley (minimum), on a closed interval  $[a, b]$  of  $\mathbb{R}$ . For a minimization problem exists a unique  $x^* \in [a, b]$  such that  $f(x^*)$  is minimum of  $f$  on  $[a, b]$  and  $f$  is monotonically decreasing for  $x \leq x^*$  and strictly increasing for  $x^* \leq x$ . It can be noted that convex functions are always unimodal. One drawback of this kind of function is that the convergence rate is slow, because it is linear.
- The algorithm provides an approximation of the global optimum rather than the exact solution.

The search scheme is iterative:

1. **Initialization:** define the interval  $[a, b]$  and set a desired tolerance level  $\epsilon$ .
2. **Calculation of the initial points:**  $d = a + (1 - \tau)(b - a)$  and  $c = a + \tau(b - a)$ .
3. **Evaluation of the initial objective function values:**  $f(d)$  and  $f(c)$ .
4. **Comparison between  $f(d)$  and  $f(c)$ :**
  - (a) if  $f(d) < f(c)$  then discard  $(c, b)$ , because the new interval of search is  $(a, c)$ . Update the new points:  $b$  becomes the previous  $c$ ,  $a$  does not change,  $c$  becomes the previous  $d$  and  $d$  is equal to  $a + (1 - \tau)(b - a)$ .
  - (b) if  $f(d) > f(c)$  then discard  $(a, d)$ , since the new interval of search is  $(d, b)$ .  $a$  becomes the previous  $d$ ,  $d$  becomes the previous  $c$ ,  $b$  remains unchanged and the new  $c$  is equal to  $a - \tau(b - a)$ .
5. **Convergence test:** if the remaining interval length  $\Delta = |c - d|$  is less than the required tolerance  $\epsilon$  exit from the loop, otherwise go back to step 4.

It is shown that

- Unimodality of the objective function enables to discard fractions of a given interval.
- The scalar quantity  $\epsilon$  represents the maximum length of the remaining interval and hence defines the accuracy of the approximate solution.
- To repeat the execution of the main loop it is necessary to compute only one new function evaluation.

### 2.3.1 Pseudo-code of the GSS

The implementation of the GSS is yielded in Alg. 1. In the following pseudo-code it is assumed that

- $kIter$  is the iteration number.
- $MaxIter$  is the maximum allowed number of iterations.
- $StopTol$  is the termination tolerance.
- $\psi_{BEST}$  is the approximated optimal value of the shape design variable.
- $\varphi_{BEST}$  is the approximated minimum value of the objective function.

## 2.4 Quadratic Fit Search

This section is concerned with the presentation of the **Quadratic Fit Search (QFS)** method described in the book of Giordano et al. (2013) [5], lecture slides of Andersson et al. (2014) [6] and Heath (2002) [7].

The QFS is a technique for seeking the minimum of a unimodal function by fitting a parabola through three points. The point at which the fitted parabola is minimized represents the approximation of the optimal solution. The search scheme is iterative:

1. **Initialization:** define the interval  $[a, b]$  and set a desired tolerance level  $\epsilon$ .
2. **Definition of the three starting points:**  $x_1 = a$ ,  $x_2 = (b - a)/2$  and  $x_3 = b$ .
3. **Evaluation of the initial objective function values:**  $f(x_1)$ ,  $f(x_2)$  and  $f(x_3)$ .

---

**Algorithm 1:** Pseudo-code for the GSS implementation

---

**Input:** a, b  
**Output:**  $\psi$ ,  $\varphi$   
**Data:** kIter; MaxIter; StopTol

```
1 Compute:  $\tau=(\sqrt{5}-1)/2$ ;  $d=a+(1-\tau)(b-a)$ ;  $c=a+\tau(b-a)$ ;  
2  $f_1 = f(d)$ ;  $f_2 = f(c)$ ;  
3 while  $kIter < MaxIter$  do  
4   if  $Fd < Fc$  then  
5      $b=c$ ;  $c=d$ ;  $f_2 = f_1$ ;  $d=a+(1-\tau)(b-a)$ ;  $f_1 = f(d)$ ;  
6   else  
7      $a=d$ ;  $d=c$ ;  $f_1 = f_2$ ;  $c=a-\tau(b-a)$ ;  $f_2 = f(c)$ ;  
8   end  
9    $errGolden = \text{abs}(c-d)$ ;  
10  if  $errGolden < StopTol$  then  
11    return  $\psi_{BEST}$  &  $\varphi_{BEST}$   
12  end  
13   $kIter = kIter + 1$ ;  
14 end
```

---

4. **Estimation of the quadratic polynomial coefficients:**  $c_1$ ,  $c_2$  and  $c_3$  for the quadratic function  $p(x)$  that goes through  $(x_1, y_1)$ ,  $(x_2, y_2)$  and  $(x_3, y_3)$ .

$$p(x) = c_1x^2 + c_2x + c_3 \quad (2.4)$$

5. **Calculation of the minimum:** once values of coefficients are known it is possible to solve for finding the unique minimum where the derivative is zero.

$$x = -\frac{c_2}{2c_1} \quad (2.5)$$

6. **Choice of a new set of points:**  $x_1$ ,  $x_2$  and  $x_3$  are updated.
7. **Convergence test:** if the increment  $\Delta = |y - y_2|$  is less than the required tolerance  $\epsilon$  exit from the loop, otherwise go back to step 4.

It is shown that

- $f(x_2) \leq f(x_1)$  and  $f(x_2) \leq f(x_3)$ .
- Once the main loop is completed, the approximated optimal value of the shape design variable is estimated by  $x_2$ .

- QFS is typically faster than GSS, since its convergence rate is super-linear.

### 2.4.1 Pseudo-code of the QFS

The implementation of the QFS is provided in Alg. 2. Inputs, outputs and data are the same as Alg. 1.

---

**Algorithm 2:** Pseudo-code for the QFS implementation

---

```

Input: a, b
Output:  $\psi$ ,  $\varphi$ 
Data: kIter; MaxIter; StopTol
15 Compute:  $x(1)=a$ ;  $x(2)=(b-a)/2$ ;  $x(3)=b$ ;
16 while  $kIter < MaxIter$  do
17   Compute:  $y(1) = g(x(1))$ ,  $y(2) = g(x(2))$ ;  $y(3) = g(x(3))$ ;
18    $coeff = \text{polyfit}(x, y, 2)$ ;  $f = \text{polyval}(coeff, x)$ ;
19    $xOpt = -coeff(2)/(2*coeff(1))$ ;  $yOpt = f(xOpt)$ ;
20    $errParab = \text{abs}(yOpt - \min(y))$ ;
21   if  $f$  is U-shaped:  $f(xOpt) < \min(f(a), f(b))$  then
22     if  $xOpt < xVal(2)$  then
23        $x(3) = x(2)$ ;  $x(2) = xOpt$ ;
24     else
25        $x(1) = x(2)$ ;  $x(2) = xOpt$ ;
26     end
27   else
28     if  $y(1) < yOpt$  then
29        $xOpt = x(1)$ ;
30     else if  $y(3) < yOpt$  then
31        $xOpt = x(3)$ ;
32     else
33        $xOpt = x(3)$ ;
34     end
35   end
36   if  $errParab < StopTol$  then
37     return  $\psi_{BEST}$  &  $\varphi_{BEST}$ 
38   end
39    $kIter = kIter + 1$ ;
40 end

```

---

# Chapter 3

## Thermal diffusive SO problem

### 3.1 PCM for LHTESS

This section reports why PCM are used for **Thermal Energy Storage (TES)**. This presentation is based on the works of Darzi et al. (2016) [13], Lohrasbi et al. (2017) [14] and Sheikholeslami et al. (2016) [15].

A TES technology is generally defined as a temporary thermal energy storage device, which quickly absorbs heat and releases it on demand.

PCM absorbs and releases large quantities of thermal energy as latent heat during melting and solidification, i.e. phase transition. Therefore, the main advantage of LHTESS is its ability to provide a high energy storage density.

The primary drawback of PCM is associated with its low thermal conductivity, which slows down the thermal transient response and thus determines a low thermal energy density absorbed or released during melting and solidification process. One method to overcome this limitation consists in implementing a heat transfer enhancement technique. This strategy involves coupling of PCM with a **Thermal Conductivity Enhancer (TCE)**, such as aluminium fins embedded in a TES tank. Because of the addition of the metal fins, either charging/discharging rate or maximum energy storage capacity must be analyzed to find the best trade-off.

The total amount of energy released during discharging mode is expressed as follows:

$$E_{released} = \int_{T_1}^{T_{liq}} C_{p,1} dT + L + \int_{T_{sol}}^{T_2} C_{p,2} dT \quad (3.1)$$

where

- $C_{p,1}$  and  $C_{p,2}$  are the specific heat capacity at constant pressure of liquid and solid phase, respectively.  $C_{p,1}$  can be equal to  $C_{p,2}$ .

- $L$  is the specific latent heat.
- $T_{liq}$  and  $T_{sol}$  denote the liquidus and solidus states, respectively.

## 3.2 Simulation workflow

A **Finite Element Method (FEM)** based simulation software, for solving various applied physics and engineering applications, consists of the following five steps:

1. **Geometric modelling:** the geometric model is created or imported.
2. **Materials and physics selection:** materials are assigned to the sub-domains of the geometry. Boundary and initial conditions, sources and fluxes are imposed on the equations according to the physical fields.
3. **Meshing:** solution domain is discretized by using the finite elements.
4. **Solving:** numerical algorithm are used to solve the algebraic system of equations.
5. **Post-processing:** numerical results are displayed to find the information desired.

Fig. 3.2 provides a block diagram of a generic engineering design.

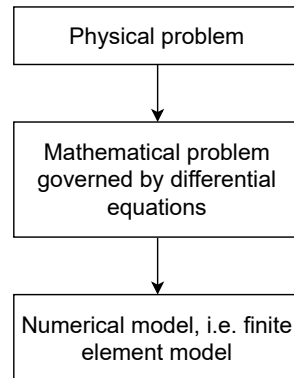


Figure 3.1: Block diagram of an engineering design.



### 3.3 Stationary heat conduction problem

#### 3.3.1 Geometry and boundary conditions

The physical system studied in this thesis is a LHTESS, which is illustrated in Fig. 3.2.

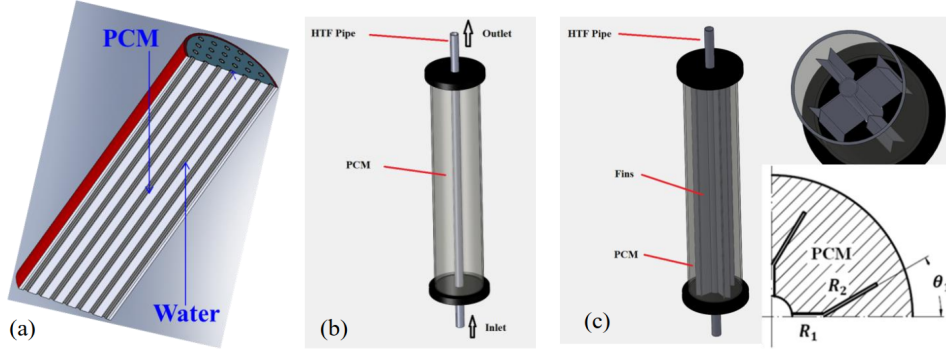


Figure 3.2: Thermal energy storage unit.

The design optimization is carried out using a simplified 2D model that represents the behaviour of a real shell-and-tube heat exchanger.

When dealing with complex geometries, if the problem is axisymmetric it is possible to exploit this geometric property to simplify the model geometry and reduce the computational expense of the solver, i.e. the time needed for running a numerical simulation. The 2D axisymmetric model created is sketched in Fig. 3.3.

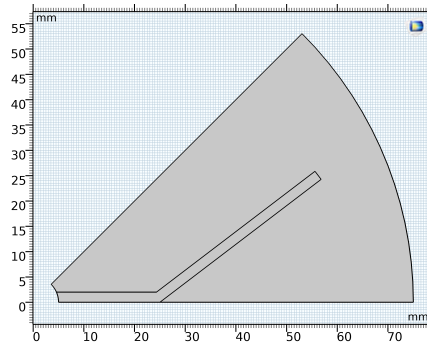


Figure 3.3: Solution domain.

The geometric model used to perform the simulation, based on discharging mode of PCM, consists of one-eighth 2D storage element, which has an inner radius of  $5[mm]$  and an outer one of  $75[mm]$ . The metal fin, i.e. thermal conductivity enhancer, is connected to the inner channel containing the

**Heat Transfer Fluid (HTF)** in order to rise thermal penetration depth into the PCM. The space between the fin and storage tank is filled with PCM. In this simplified 2D horizontal cross section model, there is no HTF flowing through the inner pipe. The geometry parameters and some of their ranges that will be used in SO procedure are tabulated in Tab. 3.1.

Table 3.1: Geometry parameters

Description	Symbol	Value in SI units
Inner radius	$R_{in}$	0.005[m]
Outer radius	$R_{out}$	0.075[m]
Primary branch length	$L_1$	[0.02, 0.04][m]
Secondary branch length	$L_2$	[0.02, 0.04][m]
Inclination angle	$\theta$	[0, $\pi/4$ ]

The boundary  $\partial\Omega = \Gamma$  of the solution domain  $\Omega$  consists of two disjoint boundaries

$$\partial\Omega = \Gamma_D \cup \Gamma_N \quad (3.2)$$

Here,  $\Gamma_D$  and  $\Gamma_N$  are the Dirichlet and Neumann boundary, respectively. The physical problem includes the following **Boundary Conditions (BC)**:

$$\begin{cases} T = T_{Dir} & \text{on } \Gamma_D \\ k\nabla T \cdot n = 0 & \text{on } \Gamma_N \\ -k\nabla^2 T = Q_{GEN} & \text{in } \Omega \end{cases} \quad (3.3)$$

where

- $T_{Dir}$  denotes the Dirichlet BC imposed on the inner wall  $\Gamma_D$  exposed to the HTF. This indicates that the temperature remains constant.
- $k\nabla T \cdot n = 0$  prescribes the Neumann BC set on the other walls. These are adiabatic or thermally insulated walls that result in null heat fluxes.
- $Q_{GEN}$  denotes the volumetric heat source term applied to the PCM phase that can be written as:

$$Q_{GEN} = \frac{L_{PCM} \cdot \rho_{PCM}}{\Delta t} \quad (3.4)$$

Here,  $L_{PCM}$  is the latent heat capacity and  $\rho_{PCM}$  is the PCM mass density.  $\Delta t$  the expedient to reduce the transient-state problem to an equivalent steady-state one.

The BC are shown in Fig. 3.4 and listed in Tab. 3.2.

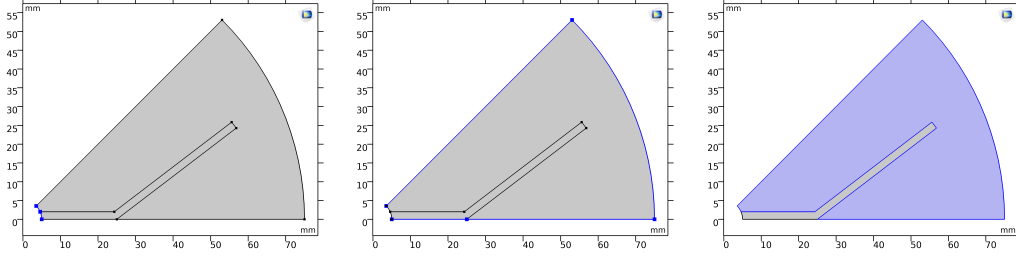


Figure 3.4: (a) Dirichlet BC (b) Neumann BC (c) Volumetric heat generation.

Table 3.2: Boundary conditions

Description	Symbol	Value in SI units
Dirichlet boundary condition	$T_{Dir}$	$37.5+273.15=310.65$ [K]
Time duration of discharging mode	$\Delta t$	3600 [s]

### 3.3.2 Governing equation

The researches conducted by Lohrasbi et al. (2016) [11] and Yang et al. (2019) [12] highlight that in solidification process, conduction heat transfer is the dominant heat transfer mechanism, while natural convection is dominant in melting process. This last heat transfer mechanism in solidification mode has one order of magnitude lower than that in melting one. In fact, as reported in the paper of Yang et al. (2019) [12], using natural convection model is almost the same as using pure conduction model to analyze PCM solidification processes. Therefore, in the simulation of discharging mode, natural convection can be neglected.

In the present study, it is assumed that conduction is the only heat transfer mechanism occurring inside the LHTESS. Consequently, the governing equation, which models the steady-state thermal response of the system, is formulated for the entire physical domain:

$$-k\nabla^2 T = Q_{GEN} \quad (3.5)$$

Here, the first term accounts for the diffusive heat transfer driven by the thermal conductivity.

The thermophysical properties of aluminium fin and PCM are summarized in Tab. 3.3.

The **Heat Transfer Interface** of COMSOL®Multiphysics sets up and solves the heat transfer in solids through heat diffusion based on Fourier's law of heat conduction. This law establishes a linear relationship between the thermal heat flux  $\mathbf{q}$  and gradient of the temperature field  $\nabla T$ . The energy

Table 3.3: Materials properties

Description	Symbol	Value in SI units
Thermal conductivity	$k_{Al}$	$180[W/(m \cdot K)]$
Thermal conductivity	$k_{PCM}$	$0.15[W/(m \cdot K)]$
Latent heat of melting	$L_{PCM}$	$200[kJ/kg]$
Density	$\rho_{PCM}$	$800[kg/m^3]$

balance equation for stationary problems in solids is written as follows:

$$\rho C_p \mathbf{u}_{trans} \cdot \nabla T + \nabla \cdot (\mathbf{q} + \mathbf{q}_r) = Q + q_0 + Q_{ted} \quad (3.6)$$

with

$$\mathbf{q} = -k \nabla T \quad (3.7)$$

where:

- $\rho$  is the density measured in  $[kg/m^3]$
- $C_p$  is the specific heat capacity measured in  $[J/(kg \cdot K)]$
- $T$  is the absolute temperature measured in  $[K]$
- $\mathbf{u}_{trans}$  is the velocity vector of translational motion measured in  $[m/s]$
- $\mathbf{q}$  is the conductive heat flux measured in  $[W/m^2]$
- $\mathbf{q}_r$  is the radiative heat flux measured in  $[W/m^2]$
- $Q$  contains additional heat source and is measured in  $[W/m^3]$
- $q_0$  is the inward heat flux measured in  $[W/m^2]$
- $Q_{ted}$  is the thermo-elastic damping measured in  $[W/m^3]$
- $k$  is the thermal conductivity measured in  $[W/(m \cdot K)]$

The energy conservation equation describes the thermal energy transport for predicting the temperature field distribution. Considering the Eq. 5.31, the conductive heat flux depends only on the thermal conductivity and the minus sign is due to the fact that positive heat transfer is directed towards negative temperature gradient. This is true for isotropic material, i.e. material with equal thermal conductivity in any direction. The knowledge of the velocity field  $\mathbf{u}_{trans}$  is not required, since natural convection can be ignored. The terms  $q_0$  and  $Q_{ted}$  disappear from Eq. 5.30 because the system is adiabatic and no coupling between the strain and temperature profile occurs.

Furthermore, no radiation is transmitted as well as no additional heat source are applied. Consequently,  $\mathbf{q}_r$  and  $Q$  are deleted from Eq. 5.30.

The solution domain is discretized into 5837 elements using an unstructured mesh consisting of triangular elements with a grid size of 1 [mm].

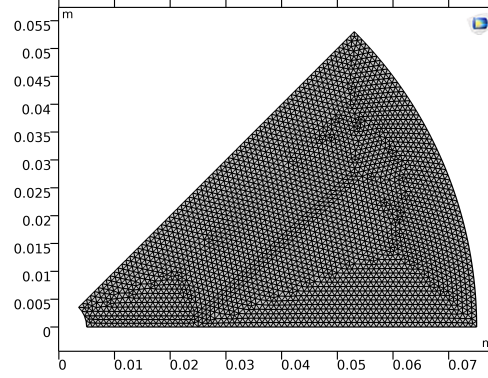


Figure 3.5: Geometric model with mesh details.

As shown in Fig. 3.5 the grid consists of triangular elements. If one needs to conduct a more accurate analysis, could select three different mesh sizes. The finest mesh is required in regions where greater gradients are expected, while the coarsest mesh density is applied to areas where lower changes in the numerical solution occur. Furthermore, to get an accurate solution from fin base to fin tip, it is necessary to locate a high mesh density zone around the fin tip.

### 3.3.3 Computational implementation

The study step and solver configuration are set-up in the **Study** branch of COMSOL.

The **MULTifrontal Massively Parallel Sparse direct Solver (MUMPS)** algorithm is selected to obtain the temperature field distribution.

A constant (Newton) non-linear solver is implemented. Furthermore, for favouring the convergence trend of a non-linear problem the Jacobian matrix is updated at each iteration. The relative tolerance is set to  $10^{-7}$ .

## 3.4 Response Surface Method

Before conducting a SO, it is advisable to develop a **Response Surface Method (RSM)**, which considers the size and shape of the aluminium fin. The implementation of this iterative process is necessary to acquire a better

understanding about what to expect from a thermal viewpoint. Thanks to several experiments the right insights will be extracted to link inputs to output response.

The aim of the response surface tool is to combine many different inputs, observe impact on the outcomes and search the design parameters having a large effect on the objective function, accelerating the whole SO process.

To generate the response surface, target function and all constraint requirements need to be selected. Moreover, based on the designer's judgment, it could be necessary to evaluate a wide number of design points, to get an accurate response surface nearer the real solutions of the system. This way to proceed provides a performance map to visualize the behaviour of a process/system, analyzing specific sets of data. Scattering uniformly the experiments in the design space, the overall performance trend can be captured. Fig. 3.6 illustrates the workflow scheme for the implementation of a DoE procedure.

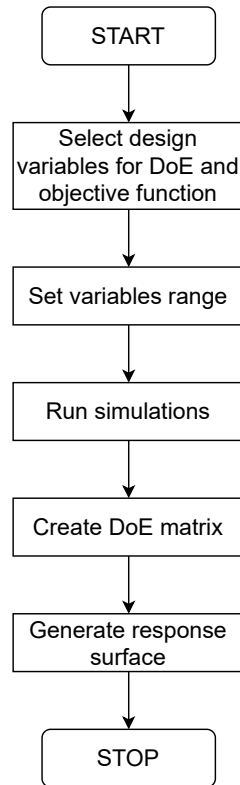


Figure 3.6: Block diagram of the DoE methodology.

The DoE strategy couples a model developed in COMSOL with a M-file using the Livelink<sup>TM</sup> for MATLAB®. This is a very useful tool when it is necessary to integrate a COMSOL **Finite Element Analysis (FEA)** with the numerical computation capability of MATLAB.

The M-file, i.e. MATLAB scripts, is executed for exploring the design space and thus to evaluate a wide range of possible configurations of the fins. With this type of workflow, a design engineer gets more detailed information to address the subsequent SO. In fact, knowing what geometric configurations yield a significant performance improvement, the design variables domain can be reduced. Narrowing the domain of the allowable variations of independent variables, the design optimization procedure can be accelerated. This is particularly evident when developing a complex products and/or processes with many different input parameters.

### 3.4.1 Implementation of the MATLAB program

The purpose of the MATLAB code is to investigate how the response of interest changes varying values of the shape design variables. The M-file helps speed-up the COMSOL simulations and collect all input and output data of interest.

As a first step of the design optimization, 25 shape designs are evaluated by modifying, in each iteration of the algorithm, the value assumed by two input parameters:

- Fin branch inclination angle:  $\theta$ .
- Fin branch length:  $L_2$ .

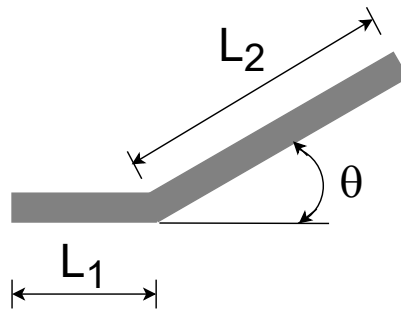


Figure 3.7: Fin geometry parameters.

The first section of the script used for creating the performance map, i.e. a surface plot that describes overall average temperature depending on the inclination angle  $\theta$  and longitudinal length  $L_2$ , is shown below.

```
%% Generate output Tavg
% R2 and theta are the independent variables or degrees of freedom

%-----
% Input data
%-----
i=1;j=1;k=1; %Iteration counters
R2Val=0.02:0.005:0.04; thetaVal=15:7.5:45; %Values of the input
variables

%-----
% Iteration loop over the first degree of freedom, i.e. R2
%-----
for i=1:length(R2Val)
    %-----
    % Iteration loop over the second degree of freedom, i.e. theta
    %-----
    for j=1:length(thetaVal)
        L2=R2Val(i);
        L1=0.06-L2;
        angle=thetaVal(j);
        model.param.set('R1',L1)
        model.param.set('R2',L2)
        model.param.set('theta',angle)
        model.study('std1').run
        model.result.numerical('av1').setResult
        Tav=mphtable(model,'tbl1')
        %-----
        % Output
        %-----
        TavgVal(k)=Tav.data; %Vector to store Tavg
        k=k+1;
    end
end
```

Throughout the iterative process, it has to take into account the prescribed fin surface area to meet the design specification set. If the cross section surface area occupied by the fins changes from one numerical simulation to another, the value of energy storage capacity will be different, as suggested in Sheikholeslami (2018) [8]. The area constraint condition can be satisfied evaluating it at every iteration, verifying the following equality:

$$L_1 = 0.06 - L_2 \quad (3.8)$$

In Eq. 4.13,  $L_2$  is the independent variable and  $L_1$  is the dependent one.

Gathering values of independent variables employed and numerical results obtained together, the response surface may be generated. Once the performance map is displayed, it is possible to understand what shape design greatly minimizes the target function.

To accomplish this, one additional phase is required. In fact input and output data of interest must be put in matrix form. The first column of the



matrix will correspond to the vector of lengths  $R_{2,Val}$ , the second column to the vector of angles  $theta_{Val}$  and the last one to the vector of average temperature over the whole computational domain  $T_{avg,Val}$ . Consequently, matrix  $M$  becomes

$$M_{25,3} = \begin{pmatrix} R2Val_{1,1} & thetaVal_{1,2} & TavgVal_{1,3} \\ R2Val_{2,1} & thetaVal_{2,2} & TavgVal_{2,3} \\ \vdots & \vdots & \vdots \\ R2Val_{25,1} & thetaVal_{25,2} & TavgVal_{25,3} \end{pmatrix}$$

When data are well organized in matrix form, post-processing can be carried out using the following code:

```
%% Post-process

X = M(:,1);
Y = M(:,2);
Z = M(:,3);
%-----
% Obtain grid data
%-----
[xGrid,yGrid] = meshgrid(linspace(min(X),max(X)),linspace(min(Y),max(Y)));
%-----
% Evaluate interpolated surface with 'cubic' option
%-----
zGrid = griddata(X(:),Y(:),Z(:),xGrid(:),yGrid(:),'cubic');
zGrid = reshape(zGrid,size(xGrid));
%-----
% Plot contours
%-----
figure
contour(xGrid,yGrid,zGrid,'ShowText','on','LineWidth',1)
hold on; scatter(X,Y,'om','Display','100G OM4') %magenta
colormap('jet'); colorbar;
title('(b) Contours of T_{avg}','fontsize',12,'fontweight','bold');
xlabel('R_2 [mm]','fontsize',10,'fontweight','bold');
ylabel('\theta [deg]','fontsize',10,'fontweight','bold');
zlabel('T_{avg} [K]','fontsize',10,'fontweight','bold');
grid on; box off;
%-----
% Plot map
%-----
figure
surf(xGrid,yGrid,zGrid)
colormap('jet'); colorbar;
title('(a) Map of T_{avg}','fontsize',12,'fontweight','bold');
xlabel('R_2 [mm]','fontsize',10,'fontweight','bold');
ylabel('\theta [deg]','fontsize',10,'fontweight','bold');
zlabel('T_{avg} [K]','fontsize',10,'fontweight','bold');
grid on; box off;
```

Response surface and isolevel curves are displayed in Fig. (a) and (b) 3.8.

The graphical findings of the response surface analysis help get a good visual of the performance map, which guides the design optimization strategy. Indeed, it can be concluded that the fin branch length  $L_2$  has a significant

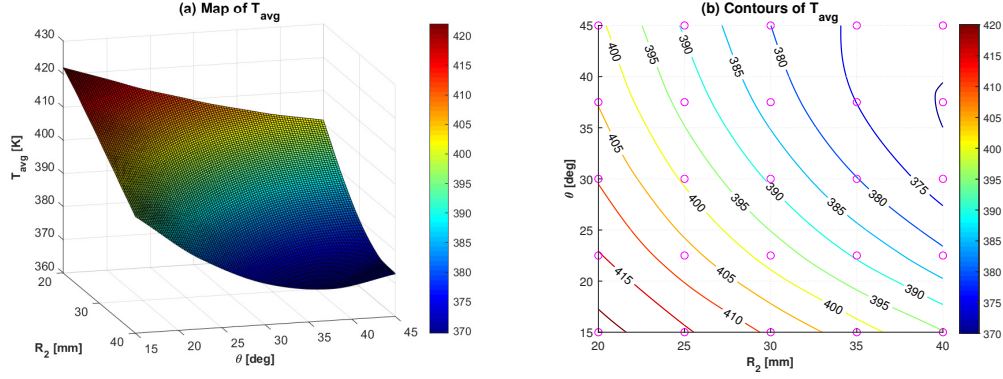


Figure 3.8: (a) Response surface (b) Isolevel curves.

impact on the enhancement of thermal penetration depth into the PCM, i.e. the conductive heat is transferred more rapidly.

The experimental design approach described above, has huge benefits, especially in terms of simulation time saved, which can be the bottle neck both in purely academic study and industrial applications. This tool can shorten simulation time of the design optimization procedure reducing the number of degrees of freedom (DoF) below the initial set of inputs by discarding factors having low impact on output response. Therefore, design engineers may select the best set of input variables for achieving the desired goal.

### 3.5 Steady-state single-objective SO with one DOF

The algorithm begins by initializing the upper and lower bounds of the design variable and computing relations between golden ratio and initial points  $d$  and  $c$ . The overall average temperature  $T_{avg}$  is selected as objective function for thermal performance evaluation of the fin configuration.

```
t=2/(1+sqrt(5));
c=a*(1-t)+b*t;
d=a*t+b*(1-t);
DOF=c;
model.param.set('theta',DOF)
model.study('std1').run
model.result.numerical('av1').setResult
Tav=mphtable(model,'tbl1')
Fc=Tav.data;
DOF=d;
model.param.set('theta',DOF)
model.study('std1').run
model.result.numerical('av1').setResult
Tav=mphtable(model,'tbl1')
Fd=Tav.data;
```

Subsequently, the main loop of the algorithm is executed.

```

if Fd<Fc
    b=c;
    c=d;
    Fc=Fd;
    d=a+(1-t)*(b-a);
    DOF=d;
    model.param.set('theta',DOF)
    model.study('std1').run
    model.result.numerical('av1').setResult
    Tav=mphtable(model,'tbl1')
    Fd=Tav.data;
else
    a=d;
    d=c;
    Fd=Fc;
    c=a+t*(b-a);
    DOF=c;
    model.param.set('theta',DOF)
    model.study('std1').run
    model.result.numerical('av1').setResult
    Tav=mphtable(model,'tbl1')
    Fc=Tav.data;
end

```

The above loop is iteratively evaluated in order to discover the optimal design variable that minimizes the objective function, until the stopping criteria are fulfilled.

```

%-----
% Plot convergence history
%-----
figure(2)
x=0:k;
y=tolGolden;
loglog(x,y*ones(size(x)),'b-','LineWidth',2); hold on;
loglog(k,errGolden,'rx','LineWidth',2); grid on; hold on;
xlim([0 k]);
ylim([10^-2 10^2]);
title('Convergence history of the GSS');
xlabel('Iteration k');
ylabel('Error');
%-----
% Check stopping criteria
%-----
if errGolden<tolGolden
    status='optimal';
    break;
end
if k>maxIter
    status='stopped';
    break;
end
%-----
% Increment iteration counter
%-----
k=k+1;

```

The numerical results obtained from this MATLAB code are displayed in Fig. 3.9 and listed in Tab. 3.4.

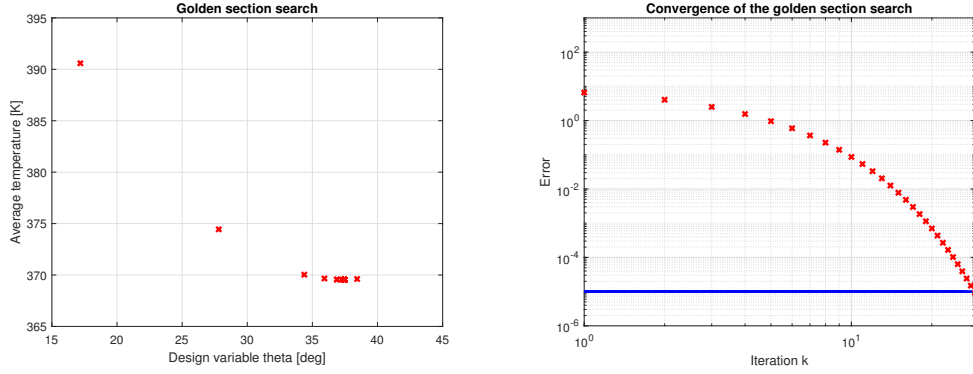


Figure 3.9: (a) GSS to minimize  $f(L, \theta)$  (b) Convergence history of the solution.

Table 3.4: The best design solution

Description	Symbol	Value in SI units
Optimum shape design variable	$\theta_{BEST}$	37.339267[deg]
Maximum design objective	$f(\theta)_{BEST}$	369.642466[K]
Number of iterations	$n_{iter}$	29

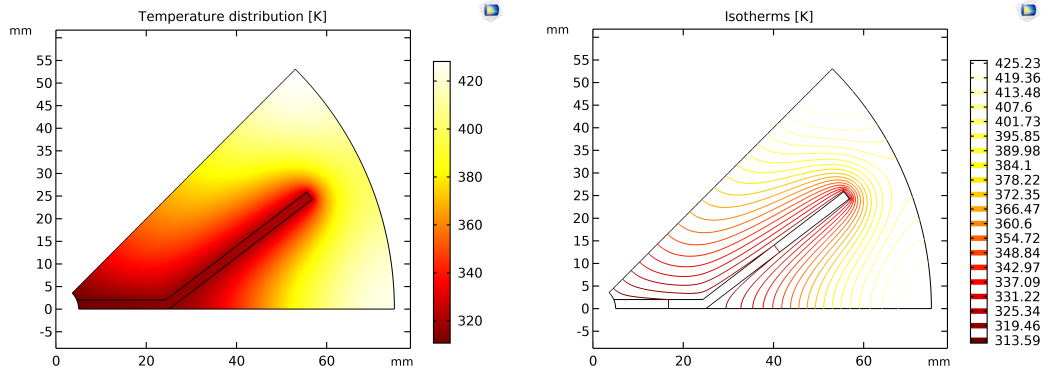


Figure 3.10: (a) Temperature profile (b) Temperature contours.

## 3.6 Steady-state single-objective SO with two DOF

The algorithm starts by pre-setting initial values and desired tolerance. Then, the algorithm follows an initialization phase, which consists of two initial vectors to store a set of three input points  $P$ ,  $Q$  and  $R$  to determine a parabola that passes through them.

```
%
% Generate x-values for plotting —> starting values of R2 and theta
%
```

```

xVal(1)=a;
xVal(2)=(b-a)/2;
xVal(3)=b;
XVal(1)=A;
XVal(2)=(B-A)/2;
XVal(3)=B;

```

Subsequently, the main loop of the algorithm, made up of two iterative steps, is executed. The first step focuses on selecting an optimum value of  $L_2$ , while satisfying volume fraction constraint.

```

%-----
% Loop over the first degree of freedom, i.e. R2
%-----
for i=1:length(XVal)
    angle=ThetaOpt;
    L2=XVal(i);
    L1=0.06-L2;
    model.param.set('theta',angle)
    model.param.set('R2',L2)
    model.param.set('R1',L1)
    model.study('std1').run
    model.result.numerical('av1').setResult
    Tav=mphtable(model,'tbl1')
    YVal(i)=Tav.data;
end
%-----
% Polyfit -> compute the coefficients
%-----
m=2;
Coeff=polyfit(XVal,YVal,m);
%-----
% Polyval -> evaluate the interpolating polynomial
%-----
F=polyval(Coeff,XVal);
%-----
%Plot
%-----
FF=@(Z) polyval(Coeff,Z);
%-----
% Find minimum of the parabola
%-----
L2Opt=-Coeff(2)/(2*Coeff(1));
%l=L2Opt;
%if L2Opt<0
%    L2Opt=-l;
%end
Tavg1=FF(L2Opt);
%-----
% Evaluate the residue
%-----
errParab1=abs(Tavg1-min(YVal))
%-----
% f(s) achieves neither a minimum nor a maximum on (A,B)
%-----
if YVal(1)<Tavg1
    L2Opt=XVal(1);
elseif YVal(3)<Tavg1
    L2Opt=XVal(3);
else
    L2Opt=XVal(3);

```

```

end
R2Opt=L2Opt;
R1Opt=0.06-R2Opt;

```

Next, the second step focuses on selecting an optimum value of  $\theta$ .

```

%-----
% Loop over the second degree of freedom, i.e. theta
%-----
for i=1:length(xVal)
    L2=R2Opt;
    L1=R1Opt;
    angle=xVal(i);
    model.param.set('theta',angle)
    model.param.set('R2',L2)
    model.param.set('R1',L1)
    model.study('std1').run
    model.result.numerical('av1').setResult
    Tav=mphtable(model,'tbl1')
    yVal(i)=Tav.data;
end
%-----
% Polyfit -> compute the coefficients
%-----
m=2;
coeff=polyfit(xVal,yVal,m);
%-----
% Polyval -> evaluate the interpolating polynomial
%-----
f=polyval(coeff,xVal);
%-----
% Plot
%-----
ff=@(z) polyval(coeff,z);
%-----
% Check concavity of the parabola
%-----
if coeff(1)<0
    status='maximum';
    sprintf('Parabolic graph is concave down');
elseif coeff(1)>0
    status='minimum';
    sprintf('Parabolic graph is concave up');
    continue
end
figure(1); hold on;
z=a:1e-5:b;
plot(z,ff(z),'b-','LineWidth',1); grid on;
title('Interpolating polynomial');
xlabel('Design variable theta [deg]');
ylabel('Average temperature [K]');
%-----
% Find minimum of the parabola
%-----
AngleOpt=-coeff(2)/(2*coeff(1));
Tavg2=ff(AngleOpt);
errParab2=abs(Tavg2-min(yVal))
if AngleOpt<xVal(2)
    xVal(3)=xVal(2);
    xVal(2)=AngleOpt;
else
    xVal(1)=xVal(2);

```

```

        xVal(2)=AngleOpt;
    end
    ThetaOpt=AngleOpt;

```

The above two loops are iteratively evaluated until the stopping criteria are fulfilled.

```

%-----
% Evaluate error
%-----
errParab=min(errParab1 , errParab2);
%-----
% Plot convergence history
%-----
figure(2)
x=0:k;
y=StopTol;
loglog(x,y*ones(size(x)), 'b-', 'LineWidth',1); hold on;
loglog(k,errParab, 'rx', 'LineWidth',3); grid on; hold on;
xlim([0 k]);
ylim([10^-4 10^2]);
title('Convergence history of the QFS');
xlabel('Iteration k');
ylabel('Error');
%-----
% Check stopping criteria
%-----
if errParab<StopTol
    status='optimal';
    disp('Solution converges');
    break;
end
if k>MaxIter
    status='stopped';
    disp('Maximum number of iterations exceeded');
    error('No convergence');
    break;
end
%-----
% Iterate the procedure
%-----
k=k+1;

```

The numerical results obtained from this MATLAB code are depicted in Fig. 3.11 and listed in Tab. 3.5.

Table 3.5: The best design solution

Description	Symbol	Value in SI units
Optimum shape design variable	$\theta_{BEST}$	37.346642[deg]
Optimum shape design variable	$L_{BEST}$	40[mm]
Maximum design objective	$f(\theta)_{BEST}$	369.642758[K]
Number of iterations	$n_{iter}$	4

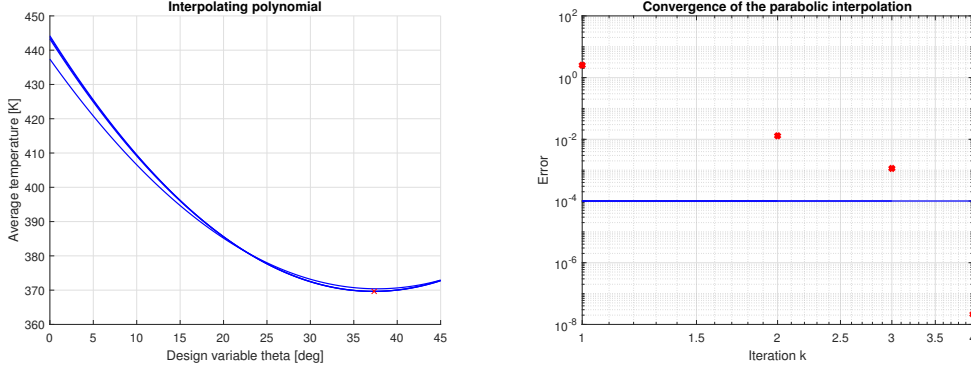


Figure 3.11: (a) QFS to minimize  $f(L, \theta)$  (b) Convergence history of the solution.

## 3.7 Time-dependent heat conduction problem

### 3.7.1 Geometry and boundary conditions

In this transient-state heat transfer analysis, the 2D axisymmetric geometry remains the same as that sketched in Fig. 3.3. What changes is the addition of one **Initial Condition (IC)** to specify:

$$\begin{cases} T = T_{Dir} & \text{on } \Gamma_D \\ k \nabla T \cdot n = 0 & \text{on } \Gamma_N \\ T = T_{init} & \text{in } \Omega \end{cases} \quad (3.9)$$

where

- $T_{init}$  prescribes the domain temperature at the initial time. This temperature is high for the discharging mode, due to stored energy. Therefore, the PCM is at superheated liquid state.

The IC is illustrated in Fig. 3.12 and reported in Tab. 3.6.

Table 3.6: Boundary conditions

Description	Symbol	Value in SI units
Initial temperature	$T_{init}$	$75+273.15=348.15$ [K]

### 3.7.2 Governing equation

The additional materials data for aluminium fin and PCM are summarized in Tab. 3.7.



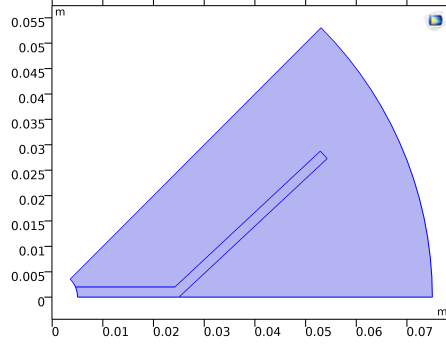


Figure 3.12: Initial temperature.

Table 3.7: Materials properties

Description	Symbol	Value in SI units
Specific heat capacity	$Cp_{Al}$	$880[J/(kg \cdot K)]$
Specific heat capacity	$Cp_{PCM}$	$840[J/(kg \cdot K)]$
Density	$\rho_{Al}$	$2750[kg/m^3]$
	$\Delta t_{melt,PCM}$	$8[K]$
	$T_{avgmelt,PCM}$	$60 + 273.15 = 333.15[K]$

The temperature field is controlled by the energy balance equation. To simulate the behaviour under transient-state conditions the next equation must be solved:

$$\rho C_p \left( \frac{\partial T}{\partial t} + \mathbf{u}_{trans} \cdot \nabla T \right) + \nabla \cdot (\mathbf{q} + \mathbf{q}_r) = Q + q_0 + Q_{ted} \quad (3.10)$$

with

$$\mathbf{q} = -k \nabla T \quad (3.11)$$

Here, the first term accounts for the thermal accumulation or release in the system.

How it happens to Eq. 5.30, certain terms of the previous equation disappear. To survive are only the first and third terms.

The PCM is modelled as solid material considering the latent heat associated with a melting/solidification process. The effective specific heat capacity for the phase change modelling is a function of temperature and can be written as

$$C_{eff,PCM} = C_{p,PCM} + \frac{L_{PCM}}{\Delta T_{SL} \sqrt{\pi}} \exp \left( -\frac{(T - T_{avg \text{ melt}})^2}{\Delta T_{SL}^2} \right) \quad (3.12)$$

where

- $\Delta T_{SL}$  is half of the phase change range.
- $T_{\text{avg melt}}$  is the average phase change temperature.

The Gaussian curve of the specific heat capacity model for the considered PCM is displayed in Fig. 3.13.

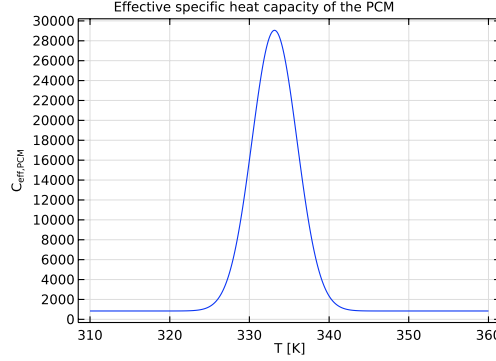


Figure 3.13: PCM specific heat capacity model.

As it is evident from the above figure, during the solidification process the PCM releases latent heat of fusion. Therefore, the effective specific heat capacity in the mushy region accounts for both sensible and latent heat.

The latent heat content of the PCM can change between 0 and  $L_{PCM}$ , when the PCM is in the solid or liquid state, respectively. For the sake of simplicity, it is assumed that the PCM liquid fraction  $\sigma$ , i.e. mass fraction of the liquid phase, in the mushy zone is a linear function defined as:

$$\begin{cases} \text{Liquid : } T > T_{liq} & 1 \\ \text{Mushy : } T_{sol} \leq T \leq T_{liq} & \frac{T - T_{sol}}{T_{liq} - T_{sol}} \\ \text{Solid : } T < T_{sol} & 0 \end{cases} \quad (3.13)$$

Here,  $T_{liq}$  and  $T_{sol}$  are the liquidus and solidus temperatures of PCM, respectively.  $T_{liq}$  is the temperature at which the PCM starts to change from liquid to solid phase and  $T_{sol}$  is the temperature at which the PCM completely changes into solid phase. The linear variation of liquid fraction in the mushy region is visualized in Fig. 3.14.

An example of an overall linear trajectory traced by the liquid fraction variation during solidification process is presented in the work of Lohrasbi et al. [11]. As illustrated in that paper, the whole trend slightly deviates from linearity only at the beginning and end of the discharging mode.

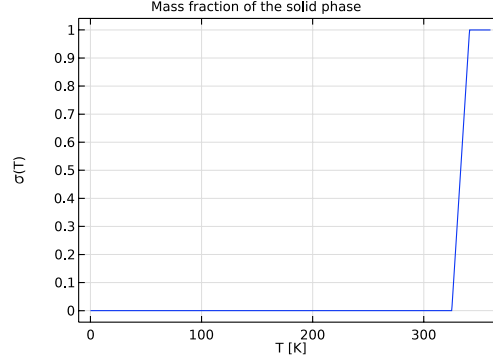


Figure 3.14: Liquid mass fraction.

The PCM specific internal energy, which is the sum of latent and sensible heat, can be calculated as

$$U_{tot,PCM} = C_{p,PCM}T + \sigma L_{PCM} \quad (3.14)$$

Here,  $T_{init}$  and  $T_{fin}$  are the initial and final temperature of the discharging process, respectively.

The volumetric energy density released by the PCM can be determined from the following equation:

$$E_{tot} = \rho_{PCM}(C_{p,PCM}T + \sigma L_{PCM}) \quad (3.15)$$

Here,  $\sigma$  is the liquid fraction of the PCM ongoing phase change.

The design objective is to maximize the total heat transfer during the discharging process. This goal is achieved by maximizing the **Total Net Heat Rate (TNHR)**. Selecting this dependent variable, the transient-state thermal investigation can be carried out to discover the best fin shape design. In fact, maximization of thermal diffusion corresponds to minimization of average temperature over the whole solution domain and thus acceleration of the solidification process. Moreover, using TNHR it is guaranteed that the optimization problem is convex and hence there is one global optimum solution. Reaching the lowest average temperature, the most efficient fin geometry in rising thermal penetration depth into the PCM is found. This approach is well explained in the work of Sheikholeslami (2018) [8].

Referring to the COMSOL User's Guide [9], TNHR is the integral of **Total Heat Flux**, i.e. convective plus conductive and radiative heat flux, over all external boundaries:

$$TNHR = \int_{\partial\Omega_{ext}} (\rho \mathbf{u} E_i - k \nabla T + \mathbf{q}_r) \cdot \mathbf{n} d\omega \quad (3.16)$$

where

- $E_i$  is the internal energy defined as:
  - (a)  $E_i = H$  for solid domains
  - (b)  $E_i = H - p/\rho$  for fluid domains

Here,  $H$  is the enthalpy.

Eq. 3.16 represents the sum of incoming and outgoing total heat flux through the system.

The grid mesh used to carry out the 2D simulation is the same as that displayed in Fig. 3.5.

### 3.7.3 Computational implementation

Both of the numerical solvers chosen in Subsec. 3.3.3 are again used.

The absolute and relative tolerances are set to 0.001 and 0.01, respectively. The tolerance factor, which multiplies the relative tolerance, is tuned to 0.0001.

The time-dependent SO problem is solved using a time integration of the PDEs choosing the **Backward Differentiation Formula (BDF)** method. This computational scheme is chosen with a minimum and maximum order of 1 and 5, respectively. Moreover, the **Nonlinear controller** is implemented for an efficient time-step control.

## 3.8 Transient-state single-objective SO with two DOF

To perform solid heat transfer simulations to discover the best possible shape design, a transient-state heat conduction in solids analysis is performed.

To conduct optimization for searching the optimal shape design it is required to compute one derived value and then store one table cell value. As mentioned in the previous section, the TNHR reached after 3600 seconds from the beginning of the simulation is selected as the objective function  $f(\theta)$  to be maximized. This choice is due to the primary design goal, which consists in improving the conductive heat transfer at the most possible. Therefore, considering that the low thermal conductivity of the PCM makes the discharging mode slow during phase transition, it is required to seek the conducting path that more quickly accelerates the solidification rate. This search corresponds to find the maximum value of the target function.

After being evaluated, the heat transfer variable is stored in one table consisting of two columns. Since the data type is **struct** it is necessary to

work with structures and arrays. Using the **dot operator** it is possible to refer to individual elements within the structure, which are known as **fields**.

```
s=struct('field1',value1,'field2',value2)
```

where

- **field1**, **field2** are the field names that correspond to  $Time[s]$  and  $TNHR[W]$ , respectively.
- **value1**, **value2** are the data assigned to those fields.

To access the elements of a struct is employed an array, whose elements are numbers. Instructions to extract only the value of interest are

```
Delete Column(0) from the struct, i.e. the table
s(1).f=table.data
v=[];
v(1)=s(1).f
```

Here, the first instruction removes left column from table of results, then value stored in that table is associated to a struct created. Next, an array is declared, and the last instruction assigns the structure value to that array.

The algorithm starts by pre-setting initial values and desired tolerance.

```
%-----
% Initialization
%-----
DOF=c;
model.param.set('theta',DOF)
model.study('std1').run
model.result.numerical('av1').set('table','tbl1');
model.result.numerical('av1').setResult
model.result.table('tbl1').removeColumn(0);
TNHR=mphtable(model,'tbl1')
F(1).c=TNHR.data
Fc=[];
Fc(1)=F(1).c
DOF=d;
model.param.set('theta',DOF)
model.study('std1').run
model.result.numerical('av1').set('table','tbl1');
model.result.numerical('av1').setResult
model.result.table('tbl1').removeColumn(0);
TNHR=mphtable(model,'tbl1')
F(1).d=TNHR.data
Fd=[];
Fd(1)=F(1).d
```

Then, the main loop of the algorithm continues until the stopping conditions are fulfilled. At each code iteration, one design objective is recomputed.

```

%-----
% Main loop
%-----
while true
    if Fd(1)>Fc(1)
        b=c;
        c=d;
        Fc(1)=Fd(1);
        d=a+(1-t)*(b-a);
        DOF=d;
        model.param.set('theta',DOF)
        model.study('std1').run
        model.result.numerical('av1').set('table','tbl1');
        model.result.numerical('av1').setResult
        model.result.table('tbl1').removeColumn(0);
        TNHR=mphtable(model,'tbl1')
        F(1).d=TNHR.data
        Fd=[];
        Fd(1)=F(1).d
    else
        a=d;
        d=c;
        Fd(1)=Fc(1);
        c=a+t*(b-a);
        DOF=c;
        model.param.set('theta',DOF)
        model.study('std1').run
        model.result.numerical('av1').set('table','tbl1');
        model.result.numerical('av1').setResult
        model.result.table('tbl1').removeColumn(0);
        TNHR=mphtable(model,'tbl1')
        F(1).c=TNHR.data
        Fc=[];
        Fc(1)=F(1).c
    end
    figure(1)
    plot(c,Fc(1),d,Fd(1),'rx','LineWidth',2); grid on; hold on;
%-----
% Plot convergence history
%-----
errGolden=abs(c-d);
figure(2)
x=0:k;
y=tolGolden;
loglog(x,y*ones(size(x)),'b-','LineWidth',2); hold on;
loglog(k,errGolden,'rx','LineWidth',2); grid on; hold on;
xlim([0 k]);
ylim([10^-3 10^2]);
title('Convergence history of the GSS');
xlabel('Iteration k');
ylabel('Error');
%-----
% Check stopping criteria
%-----
if errGolden<tolGolden
    status='optimal';
    break;
end
if k>maxIter
    status='stopped';
    break;
end
end

```

```

%————
% Increment iteration counter
%————
k=k+1;
end

```

The numerical results obtained from this MATLAB script are depicted in Fig. 3.15 and summarized in Tab. 3.8.

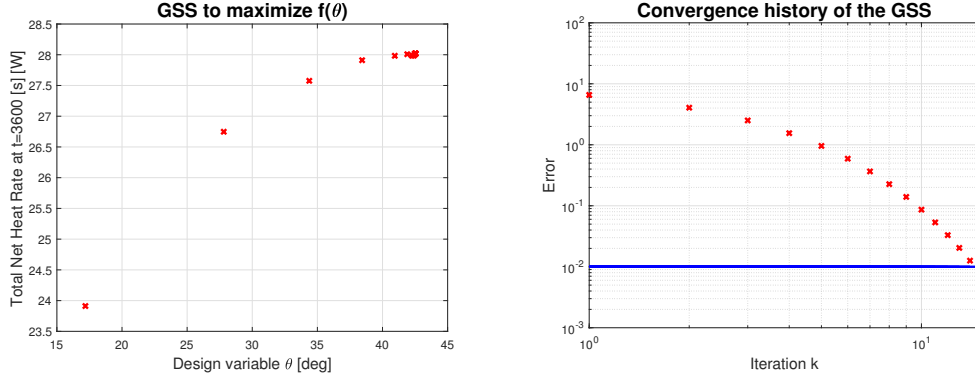


Figure 3.15: (a) GSS to maximize  $f(\theta)$  (b) Convergence history of the solution

Table 3.8: The best design solution

Description	Symbol	Value in SI units
Optimum shape design variable	$\theta_{trans,BEST}$	42.5456[deg]
Maximum design objective	$f(\theta)_{trans,BEST}$	28.02[W]
Number of iterations	$n_{iter}$	15

The temperature profiles over the computational domain and temperature contours of the optimal shape design are calculated in COMSOL and illustrated in the figures below.

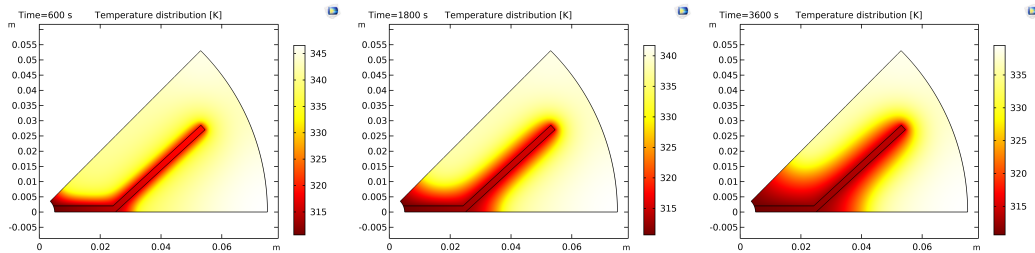


Figure 3.16: Temperature fields. (a)  $t = 600$ [s] (b)  $t = 1800$ [s] (c)  $t = 3600$ [s]

The best shape design enables to reach an overall average temperature of 328.98 [K] after 3600 seconds.

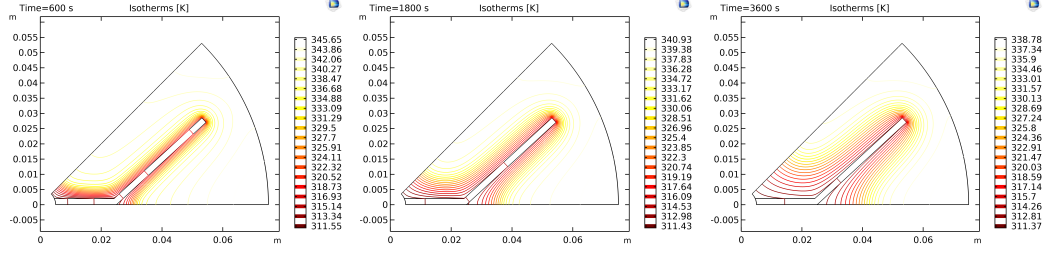


Figure 3.17: Temperature contours. (a)  $t = 600[s]$  (b)  $t = 1800[s]$  (c)  $t = 3600[s]$

The liquid fraction evolution during the solidification process is displayed in the following figures.

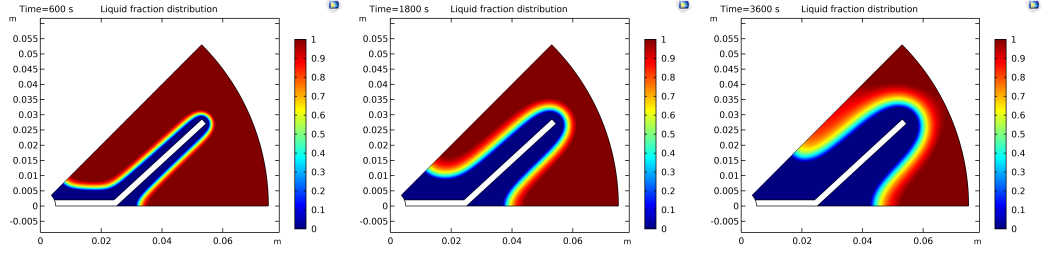


Figure 3.18: Liquid fraction fields. (a)  $t = 600[s]$  (b)  $t = 1800[s]$  (c)  $t = 3600[s]$

The figures presented above clearly show that the solidification initially starts around the metal fin and then solid phase gradually grows and spreads throughout the storage tank.

The heat transfer variables are analyzed to map the thermal performance. Average temperature history and thermal energy storage density are the most significant performance indicators that must be controlled, since they provide direct feedback on design fitness.

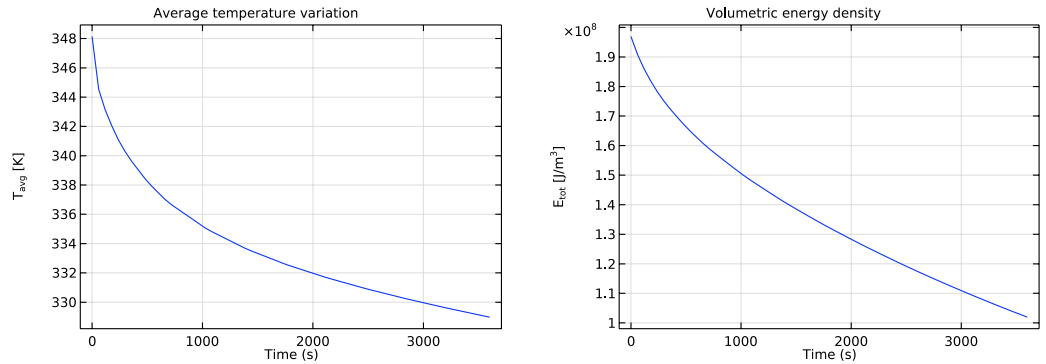


Figure 3.19: (a)  $T_{avg}$  evolution (b)  $E_{tot}$  released



It can be observed from Fig. (a) 6.19 that the average temperature profile experiences a steep decline in the beginning of the discharging process and then it continues following a steady downward trend. The discharging process of the PCM can be divided into three steps. The first phase is characterized by sensible heat release while temperature sharply falls. At the second step, PCM releases latent heat and the temperature continues to decrease, but more slowly. In the third phase, PCM releases sensible heat. As it is shown in Fig. (b) 6.19  $E_{tot}$  follows a similar trajectory as that traced by  $T_{avg}$ . After an initial sudden fall, the total energy released continuously drops along the discharging mode.

In addition to the performance parameters described above, the overall internal energy change and global energy release by the PCM, during its discharging process, can be estimated as follows:

$$\Delta U_{tot,PCM} = \int_{T_{ref}}^T C_{p,PCM} dT + \sigma L_{PCM} \quad (3.17)$$

$$\Delta E_{tot,PCM} = \rho_{PCM} \left( \int_{T_{ref}}^T C_{p,PCM} dT + \sigma L_{PCM} \right) \quad (3.18)$$

The overall internal energy change is found to be  $1.1546 \cdot 10^5 [J/kg]$ , while the global energy release is  $0.9237 \cdot 10^8 [J/m^3]$ .

# Chapter 4

## Topology Optimization problem formulation

### 4.1 Topology optimization

The presentation in this section is based upon the works of Tobe (2015) [1] and Deaton et al (2014) [19].

The analysis of several papers reveals that most of the efforts have been directed to the deep investigation of stationary heat conduction inside a rectangular design domain, while time-dependent TO has been less studied.

In the field of thermal engineering, TO is an iterative design approach that seeks to find the best possible or optimal layout for a high conducting path by properly changing material distribution inside a given physical design space, so that the resulting design meets all the prescribed design requirements, such as those imposed by performance targets and manufacturability conditions. Therefore, TO is usually performed for discovering innovative and non-intuitive designs during the conceptual or preliminary design phase.

Density-based TO is the mathematical tool to find out where to place solid or void at every point within a design domain to generate a black and white tree-like structure.

To reach the main goal of the TO process, an artificial density field, which takes values between 0 (empty space) and 1 (full material) is introduced for distinguishing one material from another, as shown in Fig. 4.1. Consequently, this pseudo-density field represents the design variable in the TO method.

A density-like variable can be mathematically defined as follows

$$\begin{cases} s(\Omega) \in [0, 1] \\ s(\Omega) = 1 \iff \in \Omega_{mat} \\ s(\Omega) = 0 \iff \in \Omega \setminus \Omega_{mat} = \Omega_{empty} \end{cases} \quad (4.1)$$

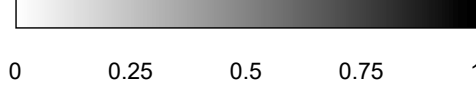


Figure 4.1: Pseudo-density variable.

where

- $\Omega$  is the design domain.
- $\Omega_{mat}$  is the region containing material.
- $\Omega_{empty}$  is the empty space.
- 0 and 1 are the lower and upper bounds of the artificial density variable.

## 4.2 Density-based approach

In this section, it will be described how to steer the design solution towards black and white layout, or solid and void. This methodology focuses on the works of Deaton et al. (2014) [20], Verbart et al. (2015) [21] and Rojas Labanda (2015) [22], Koga et al. (2013) [23]. In the next Subsec. 4.2.1 it will further extend this presentation.

The most common method used for solving TO problem is the **Solid Isotropic Material with Penalization (SIMP)**, which is classified as a **density-based approach**. The primary goal of the SIMP scheme is to converge to a binary design, where the pseudo-density variable may be equal to 0, representing the PCM with its low thermal conductivity, or to 1, representing the aluminium with its high thermal conductivity. It should be noted that this artificial density variable does not correspond to the mass density field measured in  $[kg/m^3]$ . This density-like variable represents the design variable for the optimization procedure.

When a topology design presents several elements with an intermediate artificial density is not manufacturable. Therefore, to avoid gray intermediate regions in the resulting design it is necessary to implement a non-linear relation, which pushes the gray transition zones to a discrete values 0 and 1.

The SIMP model is based on a power law equation:

$$\zeta(s) = s^p, \quad p \geq 1 \quad (4.2)$$

where

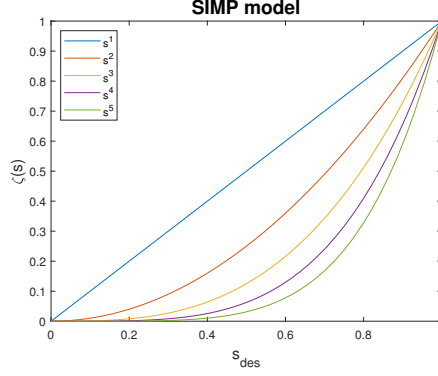


Figure 4.2: SIMP penalty functions

- $s$  is the density-like variable
- $p$  is the penalization coefficient

As can be seen in Fig. 4.2, the SIMP method introduces a monotonous function in order to ensure the convexity of the design optimization process and hence to limit the presence of local minima. In this manner a global optimum solution can be reached.

#### 4.2.1 SIMP model for TO

As explained in the previous section, one of the key themes of a TO approach is the material interpolation model that describes how the physical properties are influenced when a variation from one material to another occurs.

The general form of the SIMP penalization scheme for scaling the material properties is given by

$$\alpha_{SIMP}(s) = \alpha_{void} + (\alpha_{solid} - \alpha_{void})s^p, \quad p > 1 \quad (4.3)$$

where

- $\alpha_{SIMP}(s)$  represents a base property of a given isotropic material.
- $\alpha_{void}$  and  $\alpha_{solid}$  are the material properties of the solid and empty phases, respectively.
- $s$  is the fictitious density variable.
- $p$  is the penalty exponent

From the standpoint of thermophysical material properties modelling,  $\alpha$  may be a thermal conductivity, a specific heat capacity or a mass density.

As it is highlighted in Eq. 4.3, one will choose to use  $p > 1$  to exponentially penalize isotropic material in terms of pseudo-density variable. For problems where the volume constraint is prescribed, if a value of  $p$  much greater than 1 is set, the gray transition zones are highly penalized. Consequently, it can be noted that the penalization coefficient impacts on the quality of the solution.

Design engineers wish to obtain a pure or close to 0 and 1 material layout. Rojas (2016) [26] underlines that the SIMP method does not regularize the TO problem, because the resulting layout is not completely clear of gray-scale. Therefore, to achieve a fully black and white layout, i.e. a 0 and 1 topology, the material model must be well-conditioned specifying an additional design constraint on the perimeter for 2D structure or surfaces for 3D structure or sequentially applying filter and projection technique. These themes will be analyzed in the next section.

### 4.3 Regularization

Even though the SIMP model heavily penalizes the intermediate values of the density-like variable  $s$ , it does not suppress the numerical instabilities, such as the formation of checkerboard material distribution patterns and small-scale changes of the tree-like configuration. As reported by Soprani et al. (2016) [29] when implementing the SIMP methodology, it cannot be ensured the convergence to a global optimum solution.

As explained by Zhang et al. (2020) [27] and Soprani et al. (2016) [29], in order to regularize the pseudo-density field and obtain sharper interface boundaries between material and void, it is introduced the following **Partial Differential Equation (PDE)** with a homogeneous Neumann BC:

$$\begin{cases} -r_{fil}^2 \nabla^2 s_f + s_f = s & \text{in } \Omega \\ \nabla s_f \cdot n = 0 & \text{on } \partial\Omega \end{cases} \quad (4.4)$$

where

- $r_{fil}$  denotes the filter radius, which determines the minimum length scale of the design features that may appear in the final layout. This parameter is defined as 0.5 times the actual mesh size. Therefore, a value of  $0.5 \cdot h_{grid}$  means that the artificial density of an element is projected over all the neighbouring elements whose physical distance

between the centroids of  $i - th$  and  $j - th$  elements is lower than or equal to half of the element size.

- $s_f$  is the filtered design variable.

The first equation written above is known as **Helmholtz Filter Method** to filter small-scale topology design features and thus to smooth the interfaces between material and void, i.e. empty space.

Since density-like filtering produces large fuzzy regions, i.e. a band of intermediate density-like values, which are difficult to interpret and from an optimization standpoint are undesirable, the Helmholtz PDE needs to cooperate with the hyperbolic tangent projection method proposed by Wang et al. (2011) [28]:

$$s_p = \frac{\tanh(\beta\eta) + \tanh(\beta(s_f - \eta))}{\tanh(\beta\eta) + \tanh(\beta(1 - \eta))} \quad (4.5)$$

where

- $s_p$  is the projected design variable.
- $\eta$  denotes the projection threshold.
- $\beta$  is the projection slope, which controls the projection steepness.

This Heaviside step function transforms a filtered variable into a projected one, alleviating the problem associated with the gray transition zones. Indeed, this scheme projects every pseudo-density variable above and below a certain threshold value towards 0 and 1, respectively.

It can be noticed from Fig. (a) 4.3 that values of  $s_f$  greater or lower, respectively, than  $\eta$  are projected to 1 or 0, with a speed depending on  $\beta$ :

$$\begin{cases} 0 & \text{if } s_f < \eta \\ \eta & \text{if } s_f = \eta \\ 1 & \text{if } s_f > \eta \end{cases} \quad (4.6)$$

## 4.4 Discrete TO problem formulation

This section argues the general mathematical formulation of a density-based TO problems related to the researches of Deaton et al. (2014) [20] and Koga et al. (2013) [23] and studies of Verbart et al. (2015) [21] and Rojas Labanda (2015) [22].

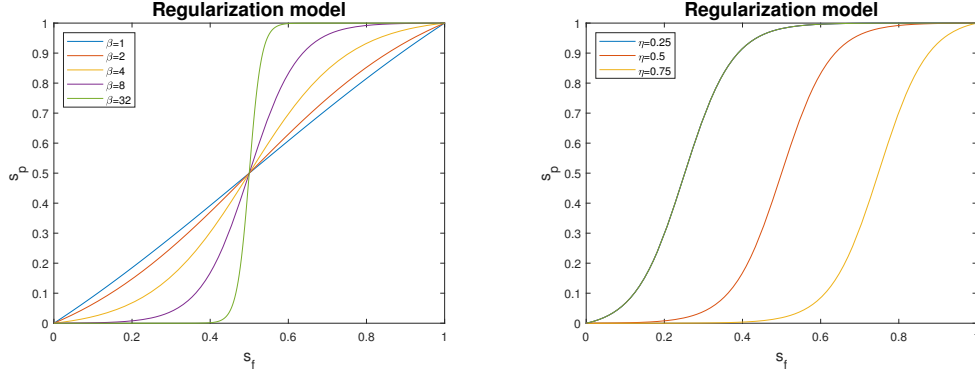


Figure 4.3: Heaviside step function.

A generalized TO problem addresses the engineering goal of placing a certain material within a given design domain, maximizing or minimizing an objective function  $f(s)$  and considering a set of constraint functions  $c(s)$ . This mathematical problem can be formulated as follows:

$$\left\{ \begin{array}{ll} \text{find} & s(\Omega) \\ \text{min} & f(s) \\ \text{subject to} & c(s) \leq 0 \\ \text{where} & V_\Omega = \int_\Omega s d\Omega = V_f V_{mat} \\ & 0 \leq s \leq 1 \end{array} \right. \quad (4.7)$$

or equivalently the discrete optimization problem can be written in matrix form

$$\left\{ \begin{array}{ll} \text{find} & s(\Omega) \\ \text{min} & f(s) = \mathbf{U}^T [\mathbf{K}(s)] \mathbf{U} = \mathbf{U}^T \mathbf{F} \\ \text{subject to} & [\mathbf{K}(s)] \mathbf{U} = \mathbf{F} \\ \text{where} & V_\Omega = \int_\Omega s d\Omega = V_f V_{mat} \\ & 0 \leq s \leq 1 \end{array} \right. \quad (4.8)$$

where

- $\mathbf{f}(s)$  is the design objective, which has to be minimized.
- $\mathbf{U}$  is the nodal unknown vector, i.e. the design solution.
- $\mathbf{K}(s)$  is the global stiffness matrix.
- $\mathbf{F}$  is the nodal load vector.
- $s$  is the density-like field vector.

- $V_\Omega$  is the design domain volume.
- $V_{mat}$  is the volume occupied by the material.
- $V_f$  is the prescribed maximum volume fraction requirement.

$[\mathbf{K}(\mathbf{s})]\mathbf{U}(\mathbf{s}) = \mathbf{F}$  is the discrete PDE version of the TO problem.

#### 4.4.1 TO of heat conduction problem

In this subsection, it will be discussed how to express a general formulation of the thermal diffusive TO problem. This topic is based on the papers of Ikonen et al. (2018) [18] and Manuel et al. (2017) [19].

Many studies have investigated TO for the design optimization of thermal system using heat conduction. This is because the heat transfer enhancement is very important in both research and industrial practice.

A summary of benefits includes:

- An increase in heat transfer performance for a given quantity of metal material.
- A cost savings thanks to a decrease in size.

Selecting the optimal topology of a certain thermal system, it will be determined the heat transfer enhancement, because this performance characteristic depends on the geometry and conducting path configuration.

Applying a discretized optimization model, TO of steady-state heat conduction problem can be mathematically stated in finite element formulation as follows:

$$\begin{cases} \text{find} & \mathbf{s}(\Omega) \\ \text{min} & \mathbf{f}(\mathbf{s}) = \mathbf{T}^T [\mathbf{K}(\mathbf{s})] \mathbf{T} = \mathbf{T}^T \mathbf{Q} \\ \text{subject to} & [\mathbf{K}(\mathbf{s})] \mathbf{T} = \mathbf{Q} \\ \text{where} & V(\mathbf{s}) = \int_\Omega s d\Omega = V_f V_\Omega \\ & 0 \leq \mathbf{s} \leq 1 \end{cases} \quad (4.9)$$

where

- $\mathbf{f}(\mathbf{s})$  is the objective function to be minimized.
- $\mathbf{T}$  is the nodal temperature vector.
- $\mathbf{K}(\mathbf{s})$  is the global thermal conductivity matrix of heat transfer equation.



- $\mathbf{Q}$  is the nodal applied thermal load vector.

It is important to point out that, since temperature is a scalar quantity, there is only one DOF per node.

$\mathbf{U}(\mathbf{s}) = [\mathbf{K}(\mathbf{s})]^{-1} \mathbf{Q}$  and  $[\mathbf{K}(\mathbf{s})]$  formed by the assembly of all element thermal conductivity matrices expressed as

$$[\mathbf{K}(\mathbf{s})] = \sum_{e=1}^E \mathbf{k}_e = k_{SIMP}(s_e) \mathbf{k}_0 \quad (4.10)$$

where

- $\mathbf{k}_e$  denotes the stiffness matrix in element  $e$ , numbered as  $e = 1, \dots, E$ .
- $s_e$  refers to element artificial density variable representing the volume fraction occupied by the metallic material.

The governing equation of heat transfer by conduction, for any solid domain, is:

$$k \nabla^2 T + Q_{GEN} = 0 \quad (4.11)$$

The TO model becomes

$$\left\{ \begin{array}{ll} \text{find} & \mathbf{s}(\Omega) \\ \text{min} & f(\mathbf{s}) \\ \text{subject to} & \mathbf{k}(\mathbf{s}) \nabla^2 T + Q_{GEN} = 0 \text{ in } \Omega \\ & T = T_{Dir} \quad \text{on } \Gamma_D \\ & \mathbf{k}(\mathbf{s}) \nabla T \cdot \mathbf{n} = 0 \quad \text{on } \Gamma_N \\ \text{where} & V(\mathbf{s}) = \int_{\Omega} s d\Omega = V_f V_{\Omega} \\ & 0 \leq \mathbf{s} \leq 1 \end{array} \right. \quad (4.12)$$

The weak formulation of the steady-state heat conduction problem can be written as follows:

$$\int_{\Omega} q(\mathbf{s}) \nabla T - Q_{GEN} d\Omega = 0 \quad (4.13)$$

Here,  $q(\mathbf{s}) = -\mathbf{k}(\mathbf{s}) \nabla T$  is the conductive heat flux.

Therefore, the general optimization objective is stated as

$$\text{Minimize } f(\mathbf{s}) = \int_{\Omega} \mathbf{k}(\mathbf{s}) \nabla T \cdot \nabla T d\Omega = \int_{\Omega} k(\mathbf{s}) \left[ \left( \frac{\partial T}{\partial x} \right)^2 + \left( \frac{\partial T}{\partial y} \right)^2 \right] d\Omega \quad (4.14)$$

or equivalently one can consider an alternative form for the design objective. This second option does not explicitly comprise the effect of the loading condition and is defined as follows

$$\text{Minimize } g(s) = \frac{1}{A_\Omega} \int_\Omega T d\Omega \quad (4.15)$$

Here, it is interesting to observe that the first target function may be represented by the dot product (or scalar product) between the transpose of the nodal temperature vector and nodal applied thermal load vector, i.e.  $f(s) = \mathbf{T}^T \mathbf{Q}$ .

## 4.5 How to solve the TO problem

This section focuses on how to solve the TO problem. This presentation is based upon the book of Christensen et al. (2009) [24], study of Rojas Labanda (2015) [22] and paper of Burger et al. (2013) [25].

TO problem is solved using the **Globally Convergent version of the Method of Moving Asymptotes (GCMMA)** that is implemented in COMSOL with the solver name **MMA**. This mathematical programming scheme has been developed by Svanberg in 1995 to solve structural optimization problems.

The properties of GCMMA are:

- Convexity, which is helpful in solving design optimization problems involving non-monotonous functions.
- Sequentially, which is expressed by successive convex approximate sub-problems, even though the starting problem is non-convex. The solution of each sub-problem is used for a new approximation.

This algorithm creates convex approximations of both the design objective and constraint functions, even though the starting problem is non-convex. In this way the problem functions present a  $U$  shape with a minimum between two vertical asymptotes.

For iteration index  $k = (0; 1)$ , i.e. when no information about the previous iterations may be exploited, the moving asymptotes for design variable  $s_i$  are updated according to the following rule:

$$L_i^k = s_i^k - (UB_i - LB_i) \quad U_i^k = s_i^k + (UB_i - LB_i) \quad \text{for} \quad (4.16)$$

Here,  $U_i^k$  and  $L_i^k$  are the current upper and lower bounds of design variable, respectively, which are changed during iterations.  
for  $k \geq 2$

- If  $(s_i^k - s_i^{k-1})$  and  $(s_i^{k-1} - s_i^{k-2})$  have opposite signs, implying oscillation, the asymptotes are restricted, forcing them nearer to  $s_i$ :

$$L_i^k = s_i^k - \delta(s_i^{k-1} - L_i^{k-1}) \quad U_i^k = s_i^k + \delta(U_i^{k-1} - s_i^{k-1}) \quad (4.17)$$

- If  $(s_i^k - s_i^{k-1})$  and  $(s_i^{k-1} - s_i^{k-2})$  have equal signs, implying convergence slowdown, the asymptotes are relaxed, pushing them away from  $s_i$  to speed up the convergence:

$$L_i^k = s_i^k - (s_i^{k-1} - L_i^{k-1})/\delta \quad U_i^k = s_i^k + (U_i^{k-1} - s_i^{k-1})/\delta \quad (4.18)$$

Here,  $\delta$  is a fixed real number less than 1.

In each iteration, the design variable must satisfy the following condition:

$$L_i^k < s_i^k < U_i^k \quad (4.19)$$

The solution procedure of the GCMMA algorithm for obtaining the optimal material distribution is illustrated in Fig. 4.4.

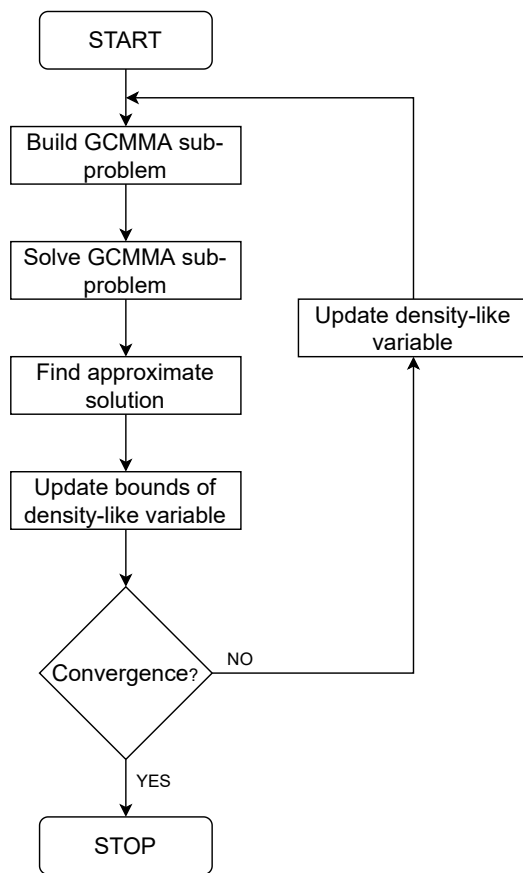


Figure 4.4: Block diagram of the GCMMA scheme.

# Chapter 5

## Thermal diffusive TO model

### 5.1 Simulation workflow

To implement the model for thermal TO of steady-state heat conduction in a LHTESS is used COMSOL. For solving a heat diffusion TO problem in response to a volumetric heat source the following COMSOL modules are required:

- **Heat Transfer (ht)** to solve the thermal diffusion problem and hence compute the temperature field distribution.
- **Optimization (opt)** to solve the constrained TO problem and thus estimate the density-like field.
- **Coefficient Form PDE (c)** to solve the Helmholtz PDE and therefore calculate the filtered density field.

TO process is looped over a series of design updates, each of which based on the design obtained in the previous iteration step. The computational workflow of a density-based TO approach is depicted in Fig. 5.1 and consists of the next phases:

1. First, the design objective, constraints and bounds are selected and the pseudo-density variable  $s$  is initialized. The initial solution consists of a uniform material distribution of the design variable over all elements in the design domain.
2. Second, the governing equations are solved to obtain the artificial density field.

3. Third, the density-like field just calculated is regularized by means of density filter and hyperbolic tangent projection to get the filtered  $s_f$  and projected  $s_p$  density field.
4. Fourth, the target function  $f(s)$  and design constraints  $c(s)$  are evaluated.
5. Fifth, if the topology design converges, the procedure is stopped, otherwise the iterative process continues updating the pseudo density variable, using the GCMMA. The optimization procedure is terminated when variation of target function, between two consecutive numerical outcomes, falls below a certain threshold defined a-priori.

Before proceeding with the pre-processing phase of the numerical simulation, it is pointed out that the TO strategy, which will be presented in the following sections of this chapter, is a tailored approach developed for the LHTESS under investigation. The major goal of this methodology is to propose a guideline for the settings of the optimization parameters to obtain successful designs.

## 5.2 Stationary heat conduction TO problem

### 5.2.1 Geometry and boundary conditions

Due to the symmetry nature of the system, the problem is solved as 2D axisymmetric to simplify geometry and boundary conditions and decrease the burden of computational complexity.

Fig. 5.2 sketches the geometry model consisting of a quarter circle.

The geometric parameters that are used to build the geometry model are reported below.

Table 5.1: Geometric parameters

Description	Symbol	Value in SI units
Inner radius	$R_{in}$	0.005[m]
Outer radius	$R_{out}$	0.075[m]

The boundary conditions imposed are:

$$\begin{cases} -k\nabla^2 T = Q_{GEN} & \text{in } \Omega \\ k\nabla T \cdot n = 0 & \text{on } \Gamma_N \\ T = T_{Dir} & \text{on } \Gamma_D \end{cases} \quad (5.1)$$

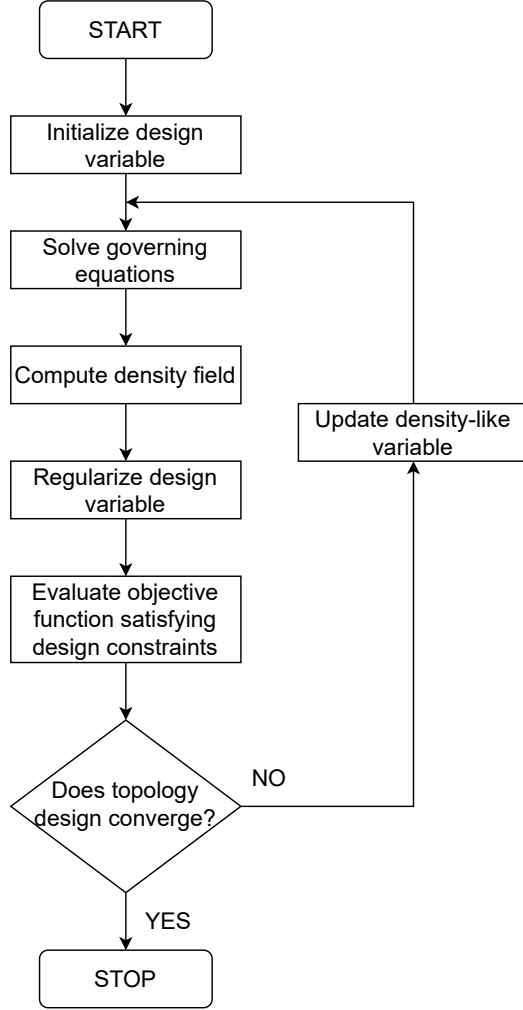


Figure 5.1: Flowchart of the TO.

where the volumetric heat source is defined as follows

$$Q_{GEN} = \frac{L_{PCM} \cdot \rho_{PCM}}{\Delta t} \quad (5.2)$$

These BCs are reported in Tab. 5.2 and shown in the next figures.

### 5.2.2 Modified governing equations

The primary goal of this work is to implement a stationary thermal diffusive TO model to search the best high conducting path, which minimizes the

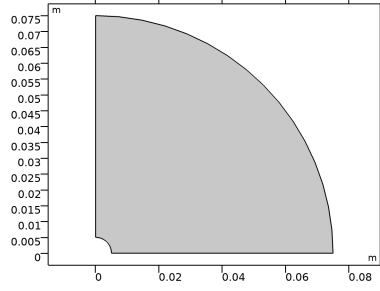


Figure 5.2: Quarter circle design domain.

Table 5.2: Boundary conditions

Description	Symbol	Value in SI units
Dirichlet boundary condition	$T_{Dir}$	$37.5+273.15=310.65$ [K]
Time duration of discharging process	$\Delta t$	3600 [s]

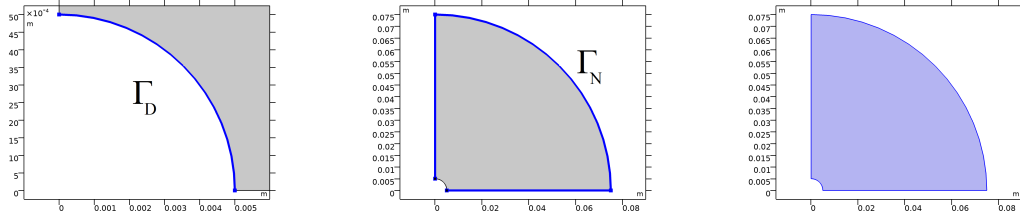


Figure 5.3: (a) Dirichlet BC (b) Neumann BC (c) Volumetric heat generation.

mean temperature over the whole computational domain.

The material properties that are necessary to describe the steady-state behaviour of the thermal system under investigation are tabulated in the following table.

Table 5.3: Thermophysical properties

Description	Symbol	Value in SI units
Thermal conductivity	$k_{Al}$	$180[W/(m \cdot K)]$
Thermal conductivity	$k_{PCM}$	$0.15[W/(m \cdot K)]$
Latent heat of melting	$L_{PCM}$	$200[kJ/kg]$
Density	$\rho_{PCM}$	$800[kg/m^3]$

The Heat Transfer module of COMSOL sets up and solves the heat transfer in solids through pure heat conduction. The energy balance equation for stationary problems in solids is written as follows:

$$\rho C_p \mathbf{u}_{trans} \cdot \nabla T + \nabla \cdot (\mathbf{q}(s) + \mathbf{q}_r) = Q + q_0 + Q_{ted} \quad (5.3)$$



with

$$\mathbf{q}(s) = -k(s)\nabla T \quad (5.4)$$

where:

- $\rho$  is the density measured in  $[kg/m^3]$
- $C_p$  is the specific heat capacity measured in  $[J/(kg \cdot K)]$
- $T$  is the absolute temperature measured in  $[K]$
- $\mathbf{u}_{trans}$  is the velocity vector of translational motion measured in  $[m/s]$
- $\mathbf{q}(s)$  is the conductive heat flux measured in  $[W/m^2]$
- $\mathbf{q}_r$  is the radiative heat flux measured in  $[W/m^2]$
- $Q$  contains additional heat source and is measured in  $[W/m^3]$
- $q_0$  is the inward heat flux measured in  $[W/m^2]$
- $Q_{ted}$  is the thermo-elastic damping measured in  $[W/m^3]$
- $k$  is the thermal conductivity measured in  $[W/(m \cdot K)]$

### 5.2.3 Mesh independence study

A grid independence test is carried out to quantify how much the quality of the mesh impacts on the simulation results. In fact, it is well known that the mesh density and element shape affect both the solution accuracy and speed of the solver. This is because a finer mesh can better capture local effects than a coarser one.

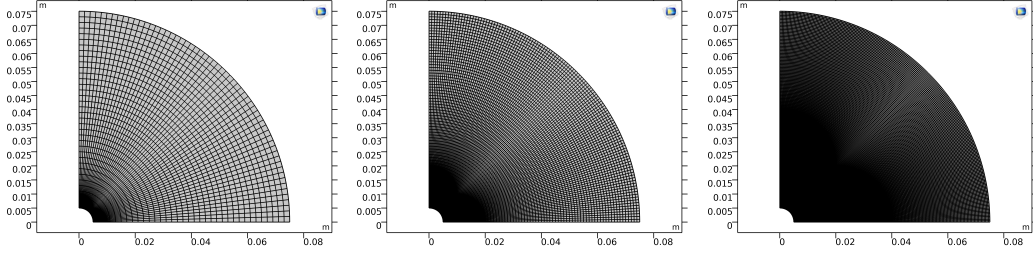
This procedure of investigating accuracy is conducted to guarantee that the used grid resolution is fine enough to capture all the physics involved in the optimization problem. To reach a good balance between precision of the numerical solution and available computing resources, a percentage change in the objective function less than 1% between two successive simulations results is established as termination criterion. If this condition is satisfied, the mesh refinement is stopped, because the numerical solution is not more influenced by the number of grid elements.

To better understand the influence of grid resolution over the resulting layout, the mesh dependency analysis is conducted by performing the same TO process with three different mesh sizes that are listed in the next table.

The entire computational domain is discretized using a mapped quadrilateral mesh for better control of the elements shape and size as displayed in the figures below.

Table 5.4: Mesh parameters

Description	Symbol	Value in SI units
Maximum mesh size	$h_{max}$	0.002 [m]
Intermediate mesh size	$h_{int}$	0.001 [m]
Minimum mesh size	$h_{min}$	0.0005 [m]

Figure 5.4: Mesh scheme. (a)  $h_{max}$  (b)  $h_{int}$  (c)  $h_{min}$ 

### 5.2.4 TO model

In this section, it will be presented how to set-up the constrained design optimization problem.

A vast amount of papers on TO formulation for heat transfer enhancement have been published. Consultation of several studies, such as You et al. (2019) [16] and Haertel et al. (2018) [17], reveals that many efforts have been directed towards the investigation of LHTESS. This is due to its high volumetric energy density that drops the energy storage cost.

A proper tuning of the opt module is necessary to develop an accurate numerical solution and hence to reach a well-designed fin pattern. To achieve the best possible design, control variable field, integral objective and inequality constraints need to be set-up. The next steps are particularly critical, because the optimization process depends on the physics of the problem under investigation.

The aim of this TO analysis is to efficiently distribute a fixed percentage of highly conducting solid material throughout the design domain to determine the optimum coral-like structure for improving the heat transfer performance.

The optimization strategy is implemented in the Optimization Module of COMSOL using a quadratic discretization. Firstly, the optimization goal has to be identified. In the opt interface the design objective is expressed as **Integral Objective** function:

$$f(T, s) = \frac{1}{A_{\Omega}} \int_{\Omega} T(s) d\Omega \quad (5.5)$$

where:

- $A_\Omega$  is the surface area of the solution domain
- $T(s)$  is the temperature depending on the pseudo-density variable

The Eq. 5.32 represents the average temperature criterion. In fact, the TO task is to minimize the mean temperature over the whole solution domain. Additionally, it should be observed that for selecting the variable  $T$  as design target an integration operator must be defined.

Secondly, the **Control Variable Field** defines the artificial density variable  $s = s(x, y)$  as a scalar field. The **Control Variable Bounds** sets lower and upper bounds corresponding to inequality constraints of the form

$$0 \leq s(x, y) \leq 1 \quad (5.6)$$

Moreover, it is necessary to set initial value of the control variable equal to  $\gamma_{max}$  that is the ratio between the fin cross-sectional area and design domain area

$$\gamma_{max} = \frac{A_{fins}}{A_\Omega} \quad (5.7)$$

The conductive material fraction is prescribed a-priori.

$$0 \leq \int_\Omega s(x, y) d\Omega \leq \gamma_{max} A_\Omega \quad (5.8)$$

This constrained quantity of high conductivity material is initially scattered in a homogeneous manner within the problem domain and then through a design optimization process the material distribution is changed to achieve the optimal thermal conductive pattern.

Thirdly, the optimization problem is subject to two design requirements, i.e. surface area and perimeter inequality constraints:

$$\frac{s}{A_\Omega \cdot \gamma_{max}} = \frac{1}{A_\Omega \cdot \gamma_{max}} \int_\Omega s(x, y) d\Omega \quad (5.9)$$

$$\Gamma \cdot (d(s, x)^2 + d(s, y)^2) = \Gamma \cdot \int_\Omega |\nabla s(x, y)|^2 d\Omega \quad (5.10)$$

Here,  $\Gamma$  denotes a scaling factor, which is computed as follows:

$$\Gamma = \frac{h' h''}{A_{fins}} \quad (5.11)$$

where

- $h'$  is the current mesh size.

- $h''$  is the parameter governing the size of details in the resulting layout.

The design restrictions introduce two regularization techniques that are normalized with respect to the domain area and mesh scale, respectively. In the opt module these limitations on the design variable are formulated as **Integral Inequality Constraints** functions.

As previously mentioned in Sec. 5.2.2, the heat conduction under steady-state conditions can be analyzed using the Fourier's law

$$-k(s)\nabla^2 T = Q_{GEN}(s) \quad (5.12)$$

Here,  $k(s)$  is the design-dependent thermal conductivity changing with the density-like variable that can be calculated by means of the SIMP non-linear interpolation method:

$$k_{SIMP}(s) = k_{PCM} + (k_{Al} - k_{PCM})s^p \quad (5.13)$$

where

- $k_{SIMP}(s)$  is the effective thermal conductivity.
- $p$  is the penalization coefficient, chosen to be 3.

The power-law interpolation scheme of the volumetric heat generation is the same as that formulated in Eq. 5.13:

$$Q_{SIMP}(s) = (1 - s^p)Q_{GEN} \quad (5.14)$$

As can be seen from the previous two equations, when  $s$  equals 1 a certain element  $i$  has the highest thermal conductivity  $k_{Al}$  and no volumetric heat generation. Conversely, when  $s$  equals 0 a certain element  $i$  has the lowest thermal conductivity  $k_{PCM}$  and contributes to volumetric heat source.

These two mathematical models are shown in the next figures.

### 5.2.5 Triple regularization strategy

In this section it will be discussed how the triple regularization strategy works to achieve a more regularized and convex optimization problem.

The triple filtering approach consists in applying the Helmholtz PDE and hyperbolic tangent projection together three times. This technique is derived from the double filter approach described by Christiansen et al. (2015) [30]. As explained in this paper, density filtering and projection are sequentially implemented a second time on the already filtered and projected design variable. The primary goal pursued is to avoid the formation of thin geometric

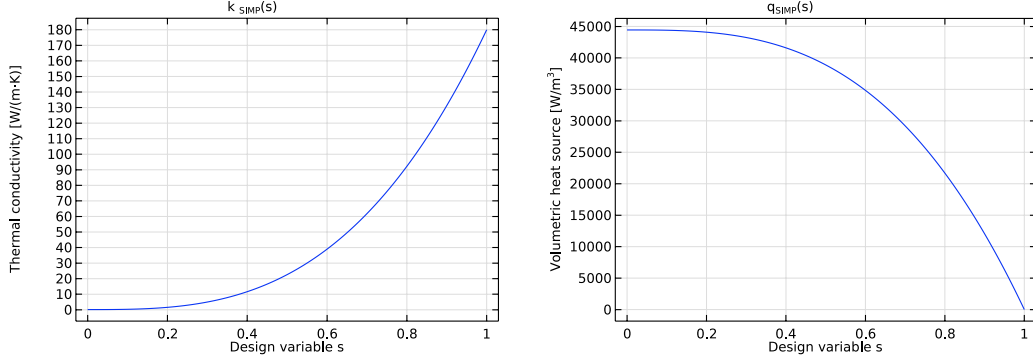


Figure 5.5: SIMP interpolation model. (a)  $k_{SIMP}(s)$  (b)  $Q_{SIMP}(s)$

irregularities and hence appearance of uncontrolled variations in the resulting optimized design.

The study conducted by Christiansen et al. (2015) [30] found out that issues related to unpredictable changes in the fuzzy zones are overcome. In fact, along each edge of the final optimized design no intermediate design variable values appear. The difference between optimal topologies and those that are obtained without applying the double regularization approach consists in the fact that using this approach the existence of local minima is deleted.

In this work will be used the triple filter approach. Therefore, the technique developed by Christiansen et al. (2015) [30] is applied one more time. To implement this procedure six additional parameters are required

- $r_{fil,2}$  is a second filter radius
- $\beta_2$  is a second steepness parameter
- $\eta_2$  is a second threshold factor
- $r_{fil,3}$  is a third filter radius
- $\beta_3$  is a third steepness parameter
- $\eta_3$  is a third threshold factor

The selection of these parameters is a very crucial step in the design optimization process. It is argued by Christiansen et al. (2015) [30] the additional parameters have to be defined with respect to the first filtering phase. The second set of values must be assigned in the following way:

- $r_{fil,2} = \frac{1}{2}r_{fil,1}$

- $\beta_2 = \frac{1}{2}\beta_1$
- $\eta_2 = \eta_1$
- $r_{fil,3} = \frac{1}{2}r_{fil,2}$
- $\beta_3 = \frac{1}{2}\beta_2$
- $\eta_3 = \eta_2$

Christiansen et al. (2015) [30] explain that if  $r_{fil,2}$  is chosen to be too high in comparison with  $r_{fil,1}$  unpredictable changes in the final layout may occur. Consequently, the double filter approach has a null effect. Selecting  $r_{fil,2}$  lower than  $r_{fil,1}$  it is ensured to achieve a refined version of the first projected design variable  $s_p$ . Also the assignment of  $\beta_2$  plays a critical role, because the optimization procedure might converge to local minima instead of the global one. Choosing  $\beta_2$  greater than  $\beta_1$  topology is instantly driven towards black and white layout and hence can result in a sub-optimal design. The projection threshold is chosen to remain constant throughout the study, because it is demonstrated that this choice has well worked for all the investigated cases.

The regularization strategy described in Sec. 4.3 is implemented in the Coefficient Form PDE Module of COMSOL using a quadratic discretization. The governing equation to be solved, i.e. the Helmholtz PDE, assumes the next form

$$-c\nabla^2 s_f + as_f = f \quad (5.15)$$

where

- $c = (0.5 \times h)^2$  is the diffusion coefficient and  $h$  is the current mesh size.
- $s_f$  is the independent variable.
- $a = 1$  is the absorption coefficient.
- $f = s$  is the source term and  $s$  is the density-like variable.

Once the values of  $s_f$  are computed, the dependent variable  $s_p$  can be evaluated by means of the **Variables** feature

$$s_p = \frac{\tanh(\beta\eta) + \tanh(\beta(s_f - \eta))}{\tanh(\beta\eta) + \tanh(\beta(1 - \eta))} \quad (5.16)$$

The tuning of TO model parameters is shown in the following tables. The steepness parameter  $\beta$  for the projection step has a starting value of 32 and then it is halved after each simulation until a final value of 8 is reached. The filter radius  $r_{fil}$  for the projection level takes an initial value of 1[mm]. Next, it is halved in the same manner as  $\beta$  to achieve a final value of 0.25[mm].

Table 5.5: Model parameters

Description	Symbol	Value in SI units
SIMP exponent	$p$	3
Volume fraction	$\gamma_{max}$	0.054567
Scale factor	$\Gamma$	0.0083
Threshold parameter	$\eta$	0.25

Table 5.6: Simulation (a) First (b) Second (c) Third

Symbol	Value	Symbol	Value	Symbol	Value
$r_{fil}$	$0.5 \times h_{max}$	$r_{fil}$	$0.5 \times h_{int}$	$r_{fil}$	$0.5 \times h_{min}$
$\beta$	32	$\beta$	16	$\beta$	8

### 5.2.6 Resulting TO model

Considering all implementations described in the previous sections, the resulting TO model becomes

$$\left\{ \begin{array}{ll} \text{find} & s(\Omega) \\ \text{min} & f(T, s) \\ \text{subject to} & \begin{array}{ll} k(s)\nabla^2 T + Q_{GEN} = 0 & \text{in } \Omega \\ T = T_{Dir} & \text{on } \Gamma_D \\ k(s)\nabla T \cdot n = 0 & \text{on } \Gamma_N \end{array} \\ & \begin{array}{ll} -r_{fil}^2 \nabla^2 s_f + s_f = s & \text{in } \Omega \\ \nabla s_f \cdot n = 0 & \text{on } \partial\Omega \end{array} \\ \text{where} & \begin{array}{l} A(s) = \int_{\Omega} s d\Omega = \gamma_{max} A_{\Omega} \\ 0 \leq s \leq 1 \end{array} \end{array} \right. \quad (5.17)$$

### 5.2.7 Computational implementation

In this section, it is will be described how to run a simulation.

The optimization problem is computed using the **Globally Convergent** version of the **Method of Moving Asymptotes (GCMMA)**. An **Optimality tolerance**, i.e. change between two consecutive density-like variable, of 0.02 is specified for the first simulation. Subsequently, this parameter is halved after each simulation until a final value of 0.005 is obtained. At the end of each iteration, the numerical solution reached must be checked to see if the error due to the discretization satisfies the desired tolerance. If the error estimated is less than the threshold specified the optimization iteration stops, because the termination condition is satisfied. Furthermore, it is

added a second stopping criterion. This condition consists in assigning the maximum number of model evaluations.

The MUMPS algorithm is chosen to calculate the numerical solution of the sparse linear system of discretized finite element equations of the form

$$\mathbf{A}\mathbf{u} = \mathbf{b} \quad (5.18)$$

where

- $A$  is the  $M \times M$  sparse coefficient matrix.
- $u$  is the  $M \times 1$  vector of unknowns.
- $b$  is the  $M \times 1$  vector of known quantities.

A constant (Newton) non-linear solver is implemented. Moreover, to encourage the convergence trajectory of a non-linear problem the Jacobian matrix is updated on every iteration.

The information extracted from the **Utility** feature are listed in the tables below.

Table 5.7: Simulation (a) First (b) Second (c) Third

Symbol	Value	Symbol	Value	Symbol	Value
N. of MEs	2065	N. of MEs	8260	N. of MEs	33040
N. of T DOFs	2160	N. of T DOFs	8449	N. of T DOFs	33417
N. of $s$ DOFs	8449	N. of $s$ DOFs	33417	N. of $s$ DOFs	132913
$\epsilon_{opt,1}$	0.02	$\epsilon_{opt,2}$	0.01	$\epsilon_{opt,3}$	0.005
$MaxIter_1$	1000	$MaxIter_2$	500	$MaxIter_3$	250

where

- **N. of MEs** is the number of mesh elements.
- **N. of DOFs** is the number of degrees of freedom.
- $\epsilon_{opt}$  is the optimality tolerance.

### 5.3 Simulation results

The simulation results using COMSOL have helped understand the solution process of the design optimization problem.

The numerical results obtained from the simulation are tabulated in the next tables.

where



Table 5.8: Simulation (a) First (b) Second (c) Third

Parameters	Value	Parameters	Value	Parameters	Value
$NIter$	201	$NIter$	88	$NIter$	41
$f(s)_{fin}$	1.637595	$f(s)_{fin}$	1.595627	$f(s)_{fin}$	1.592987
$T_{avg}$	372.33 [K]	$T_{avg}$	362.79 [K]	$T_{avg}$	362.19 [K]
$s_{avg} = s_{f,avg}$	0.054568	$s_{avg} = s_{f,avg}$	0.054569	$s_{avg} = s_{f,avg}$	0.054566
$S_{avg}$	0.087312	$S_{avg}$	0.087715	$S_{avg}$	0.087252
$s_{p,avg}$	0.085808	$s_{p,avg}$	0.089468	$s_{p,avg}$	0.088963

- $NIter$  is the number of GCMMA iterations.
- $\cdot_{avg}$  is the subscript denoting the **Average operator (aveop)**.
- $\cdot_{int}$  is the subscript referring to the **Integration operator (intop)**.

As listed in the tables above, both overall mean temperature  $T_{avg}$  and objective function  $f(s)$  drop with decreasing mesh size. Conversely, the projected design variable  $s_p$  increases with reducing grid size.

To fully achieve a desired level of accuracy of the numerical solution and demonstrate that the topology optimized designs are mesh-independent, a grid convergence study is performed. This analysis is carried out by refining the mesh size after each simulation.

As stated in Sec. 5.2.3, the grid convergence study is considered to be completed when a difference in values of the final objective function  $f(s)_{final}$  and overall average temperature  $T_{avg}$  between two consecutive computational solutions less than 1% is reached. The accuracy  $\chi$  can be evaluated as

$$\%_{change, f(s)} = \left| \frac{f(s)_{final,3} - f(s)_{final,2}}{f(s)_{final,2}} \right| 100\% = 0.002\% \Rightarrow \chi_{f(s)} = 99.998\% \quad (5.19)$$

$$\%_{change, T_{avg}} = \left| \frac{T_{avg,final,3} - T_{avg,final,2}}{T_{avg,final,2}} \right| 100\% = 0.002\% \Rightarrow \chi_{T_{avg}} = 99.998\% \quad (5.20)$$

As a result of the grid sensitivity analysis implemented, it is found that the most suitable size of the mesh is  $1[mm]$  to provide a grid-independent numerical solution. Consequently, if the grid size was further declined the mesh quality would rise, but computational expense required to solve the optimization problem would not produce more accurate computational results. Therefore, it can be observed that the accuracy of the TO approach is verified.

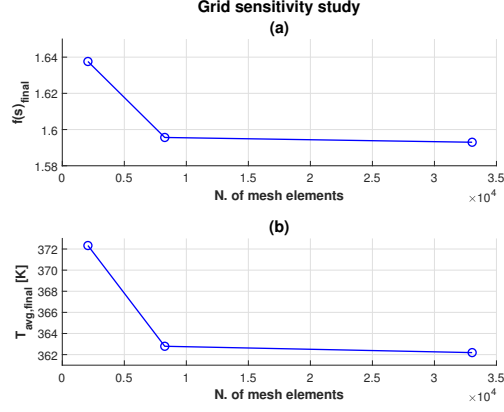


Figure 5.6: Mesh independence analysis. (a)  $f(s)_{final}$  (b)  $T_{avg}$

The convergence trend of the mesh sensitivity study for the three grid levels employed is displayed in Fig. (a) and (b) 5.6. The figures above show that the design objective becomes insensitive to further mesh refinement after three simulations, i.e. when an asymptotic convergence to a plateau is achieved.

After solving the optimization problem, the optimal designs can be visualized.

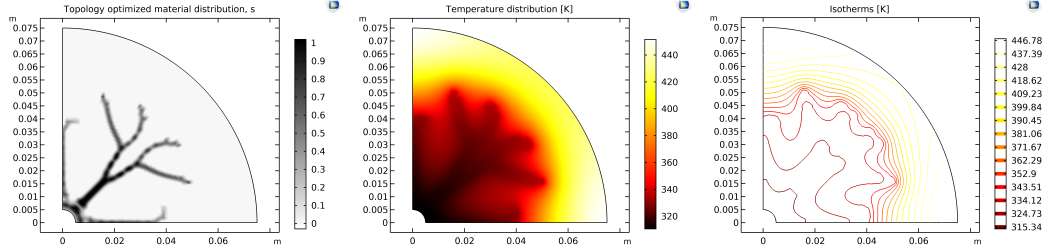


Figure 5.7: First simulation. (a) Design variable (b) Temperature field (c) Temperature contour

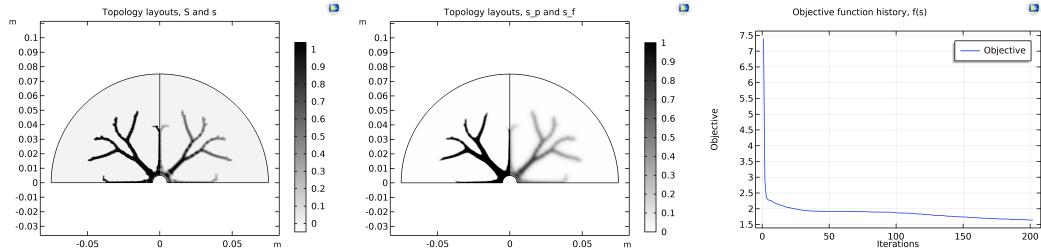


Figure 5.8: First simulation. (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

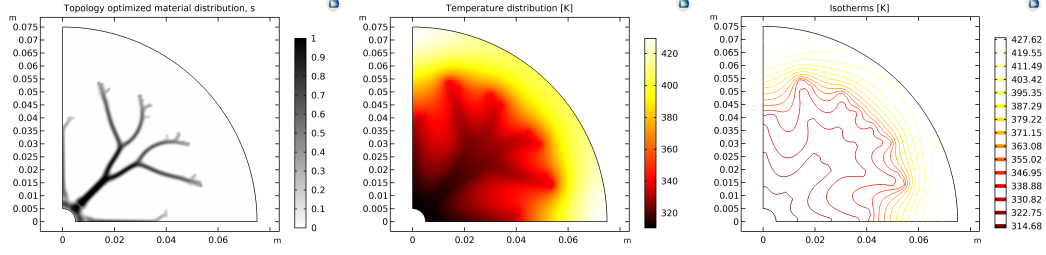


Figure 5.9: Second simulation. (a) Design variable (b) Temperature field (c) Temperature contour

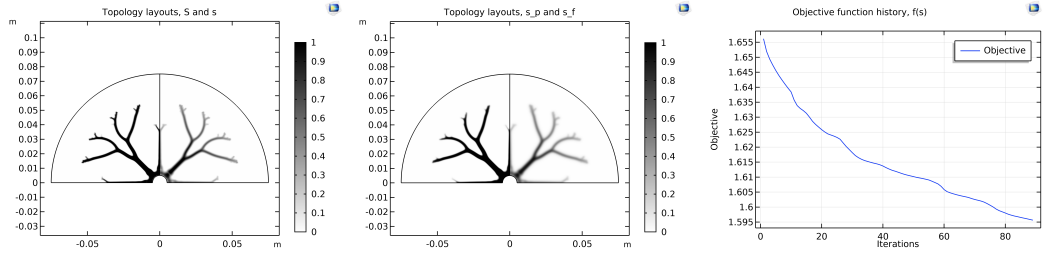


Figure 5.10: Second simulation. (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

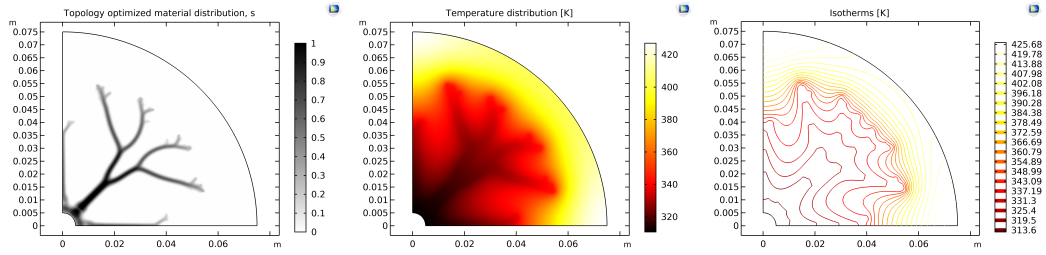


Figure 5.11: Third simulation. (a) Design variable (b) Temperature field (c) Temperature contour

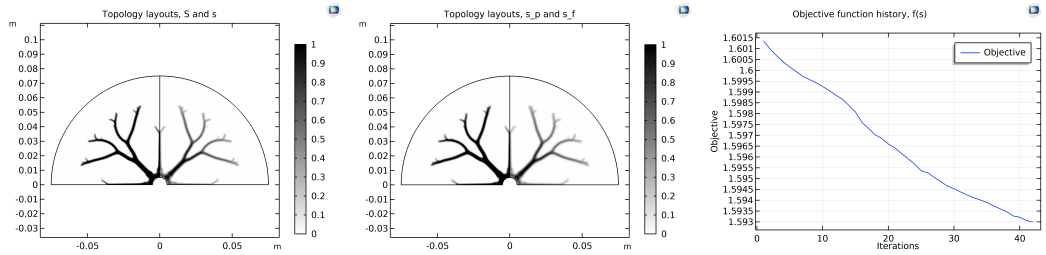


Figure 5.12: Third simulation. (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

As can be seen in the previous figures the pseudo-density field, i.e. the control variable field  $s$ , is illustrated through a black and white design, where black zones correspond to high conductivity material distribution. These figures clearly show that the quality of the topologies improves by applying

the techniques argued in the previous sections. Indeed, both fuzzy zones and geometric irregularities, which makes the manufacturing process impossible or at least very difficult, gradually disappear.

The tree-like topologies are constituted of one main trunk with two secondary branches, which form a *Y* configuration. Each of these branches have two further branches with various thin bulges. Additionally, it is important to point out that between the second and third optimized design no significant changes may be detected. In fact, only very small details enable to distinguish one layout from another. These small features are due to reduction of the filter radius.

As expected, the temperature profile experiences lower values in the regions close to the edge where the Dirichlet boundary condition is imposed.

According to the line graphs, which traces variations of the objective functions against GCMMA iterations, it is possible to state that the target functions monotonically decrease and thus the downward trajectories approach a horizontal asymptote. Therefore, the design optimization process converges to a stable value.

Since the topology optimized designs are not easy to understand, due to the fact that only a quarter of the design domain is analyzed, it is helpful to use the **Sector Data Set** feature. Using this functionality the entire solution domain can be visualized.

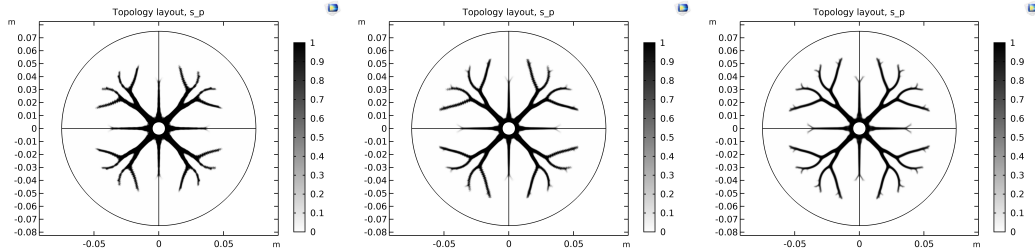


Figure 5.13: Projected design field. (a) First simulation (b) Second simulation (c) Third simulation

## 5.4 Presentation of the case studies

In this section, it will be discussed the numerical tests conducted by varying combination of input parameters.

The first phase of this analysis will be concerned with investigating how changes in scale factor  $\Gamma$  would affect system performance. The next step will consider the understanding of how variations in volume fraction  $\gamma_{max}$  would impact overall performance. All these studies are carried out by maintaining constant projection threshold  $\eta$ .

### 5.4.1 $\gamma_{max}$ sensitivity study

To discover the influence of  $\gamma_{max}$  over design target, two different maximum fin surface area fractions are numerically studied as listed below.

Table 5.9: Fin cross-sectional area requirement

$\gamma_{max,2} = \gamma_{max,1} \times 2$	0.10913
$\gamma_{max,3} = \gamma_{max,1} \times 3$	0.1637

The optimized designs and temperature fields obtained after the optimization process are depicted in the following figures.

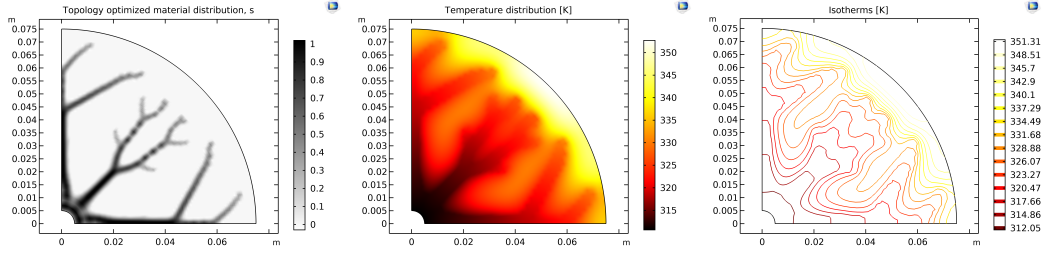


Figure 5.14: First simulation at constant  $\gamma_{max,2}$ . (a) Design variable (b) Temperature field (c) Temperature contour

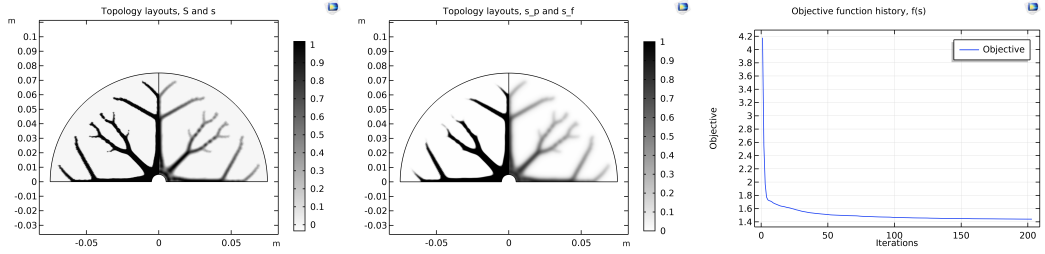


Figure 5.15: First simulation at constant  $\gamma_{max,2}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

Comparison among pseudo-density fields obtained conducting  $\gamma_{max}$  sensitivity analysis and those of the reference case study provided in Sec. 5.3 reveals that the coral-like topologies show a growing number of branches with increasing value of  $\gamma_{max}$ . This is because raising amount of highly conducting solid material the optimal designs become more complex.

It is useful for estimating the sensitivity of TO model to changes in the maximum fin volume fraction to display the computational outcomes.

It is evident from the figures above that both the mean temperature over the entire solution domain and final objective function  $f(s)_{final}$  decline with a

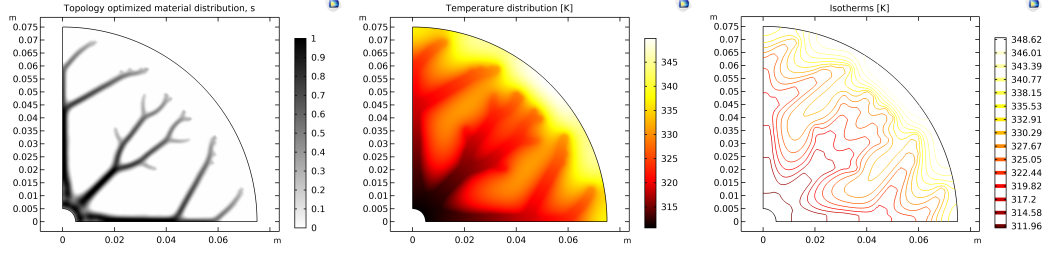


Figure 5.16: Second simulation at constant  $\gamma_{max,2}$ . (a) Design variable (b) Temperature field (c) Temperature contour

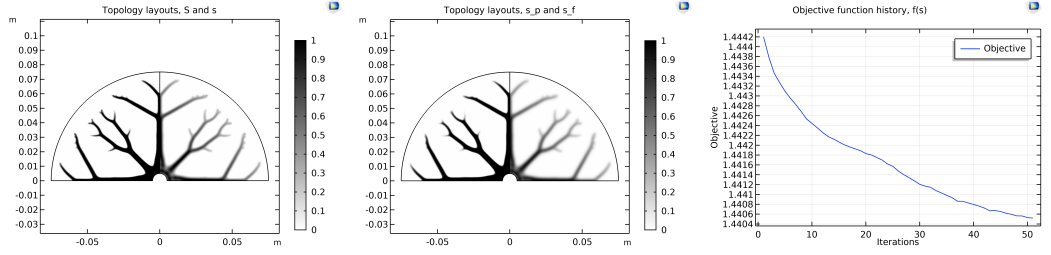


Figure 5.17: Second simulation at constant  $\gamma_{max,2}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

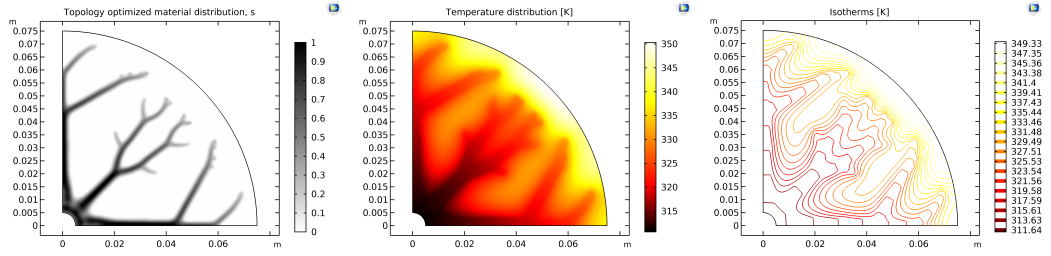


Figure 5.18: Third simulation at constant  $\gamma_{max,2}$ . (a) Design variable (b) Temperature field (c) Temperature contour

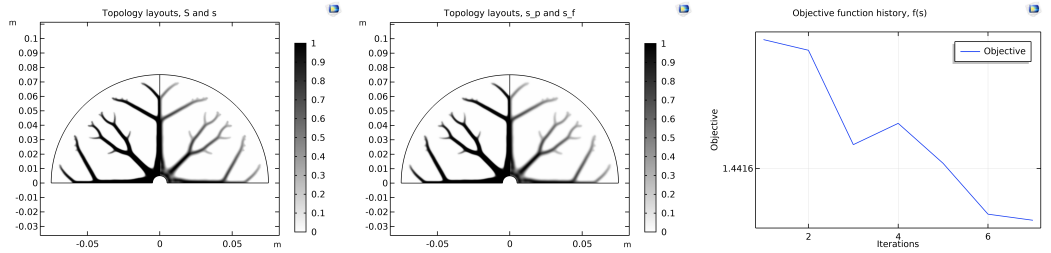


Figure 5.19: Third simulation at constant  $\gamma_{max,2}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

steady rise of the volume fraction constraint. Consequently, an increase of the amount of high conductive material can be operated if a strong enhancement

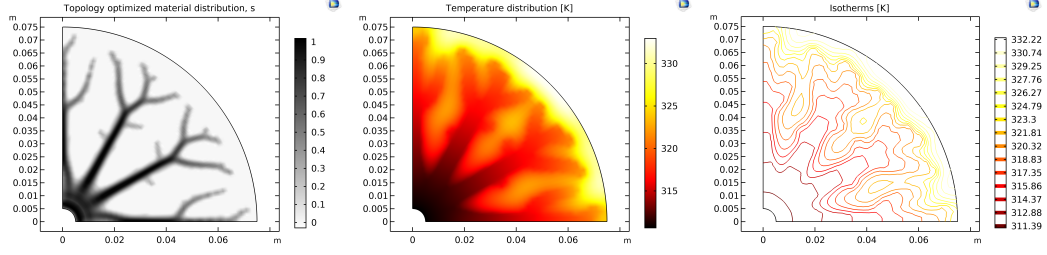


Figure 5.20: First simulation at constant  $\gamma_{max,3}$ . (a) Design variable (b) Temperature field (c) Temperature contour

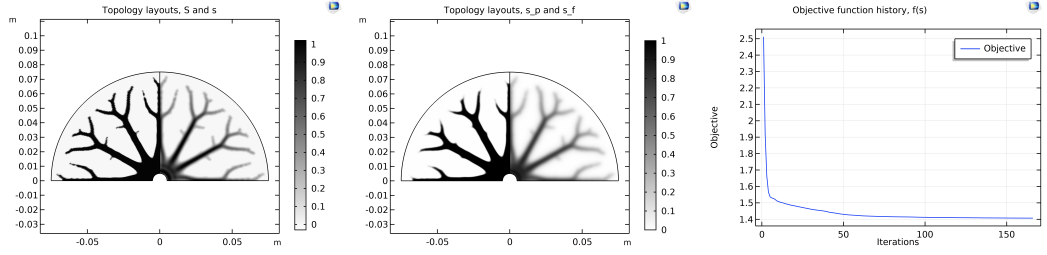


Figure 5.21: First simulation at constant  $\gamma_{max,3}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

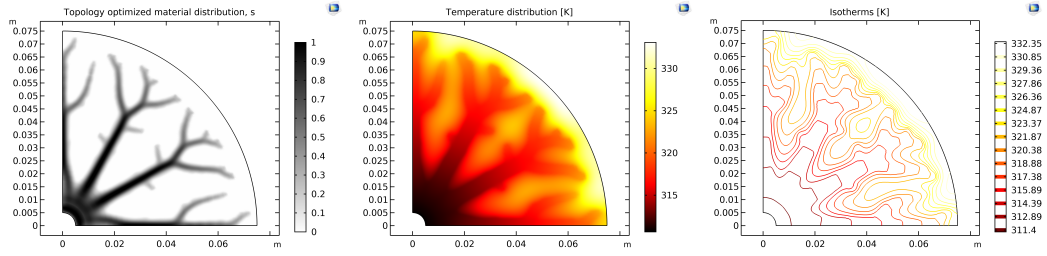


Figure 5.22: Second simulation at constant  $\gamma_{max,3}$ . (a) Design variable (b) Temperature field (c) Temperature contour

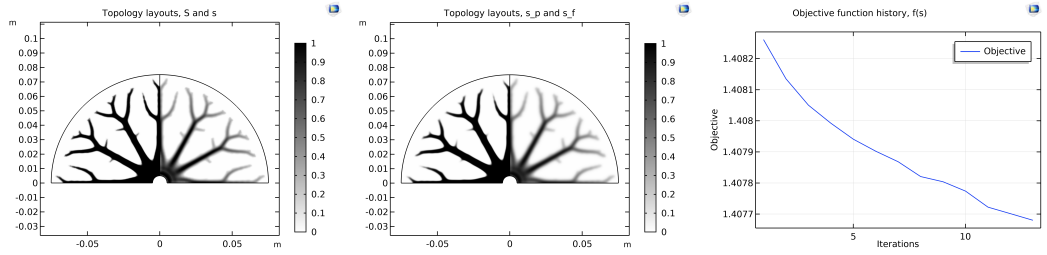


Figure 5.23: Second simulation at constant  $\gamma_{max,3}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

of the thermal penetration depth into the PCM is required. The percentage of performance improvement due to the solid material usage is quantified in

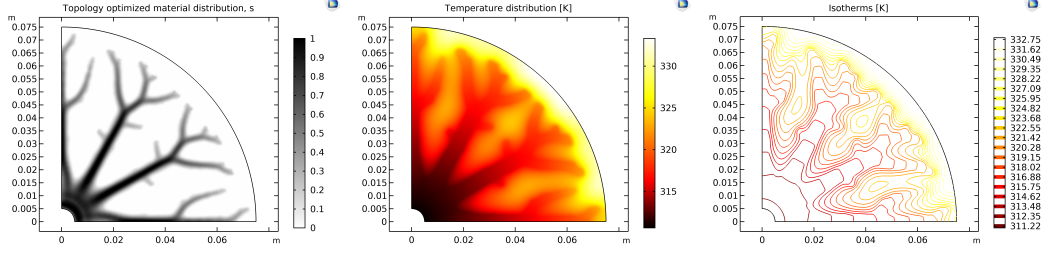


Figure 5.24: Third simulation at constant  $\gamma_{max,3}$ . (a) Design variable (b) Temperature field (c) Temperature contour

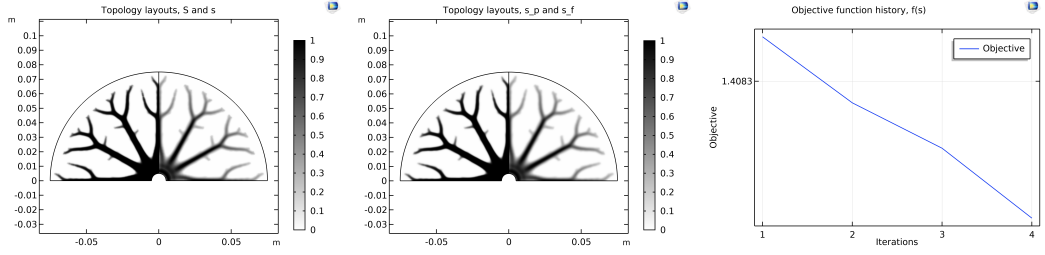


Figure 5.25: Third simulation at constant  $\gamma_{max,3}$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

Table 5.10:  $T_{avg}$  varying values of  $\gamma_{max}$

$T_{avg,1}$ for $\gamma_{max} = 0.054567$	$362.19[K]$
$T_{avg,2}$ for $\gamma_{max} = 0.10913$	$327.76[K]$
$T_{avg,3}$ for $\gamma_{max} = 0.1637$	$320.18[K]$

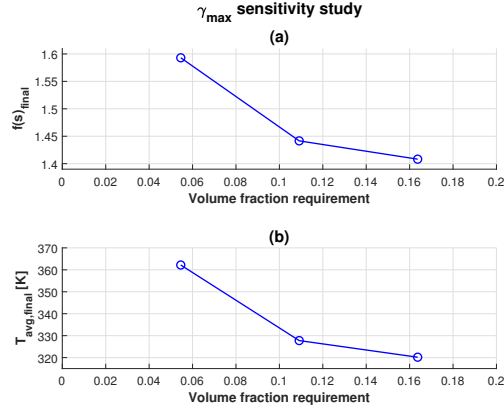


Figure 5.26: (a)  $f(s)_{final}$  (b)  $T_{avg}$

the following way:

$$\Delta_{T_{avg,1 \rightarrow 2}} = T_{avg,1} - T_{avg,2} = 34.43[K] \quad (5.21)$$



$$\%change_{T_{avg,1 \rightarrow 2}} = \frac{\Delta_{T_{avg,1 \rightarrow 2}}}{T_{avg,1}} \cdot 100\% = 9.506\% \quad (5.22)$$

$$\Delta_{T_{avg,2 \rightarrow 3}} = T_{avg,2} - T_{avg,3} = 7.58[K] \quad (5.23)$$

$$\%change_{T_{avg,2 \rightarrow 3}} = \frac{\Delta_{T_{avg,2 \rightarrow 3}}}{T_{avg,2}} \cdot 100\% = 2.313\% \quad (5.24)$$

### 5.4.2 $\Gamma$ sensitivity study

To find out the influence of  $\Gamma$  on design objective, two different scaling parameters are examined in this sensitivity study. These two factors are tabulated in the next table.

Table 5.11: Scale factors

$\Gamma_2 = h_{Int} \cdot h_{Int} / FinSurf$	0.0042
$\Gamma_3 = h_{Max} \cdot h_{Max} / FinSurf$	0.0167

The optimum topologies and temperature profiles obtained after the optimization procedure are illustrated in the figures below.

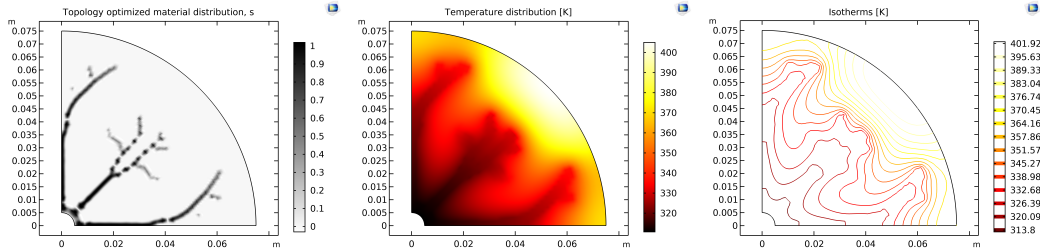


Figure 5.27: First simulation at constant  $\Gamma_2$ . (a) Design variable (b) Temperature field (c) Temperature contour

It can be noticed that the resulting topologies maintain two branches in a Y configuration as well as the reference case study. It is clear that a reduction of  $\Gamma$  leads to appearance of two narrow and long additional branches. Conversely, an increase of the scaling factor conducts to disappearance of these two branches. Moreover, the coral-like structure resulting from the assignment of  $\Gamma_2$  is much more well-defined with respect to the other one. Indeed, the tree-like topology related to  $\Gamma_3$  exhibits various fuzzy transition regions and thus it is worse-defined. The last thing to underline is that even though the optimization problem is carried out at constant  $\gamma_{max}$ , the final layouts achieved by changing values of  $\Gamma$  are quite divergent.

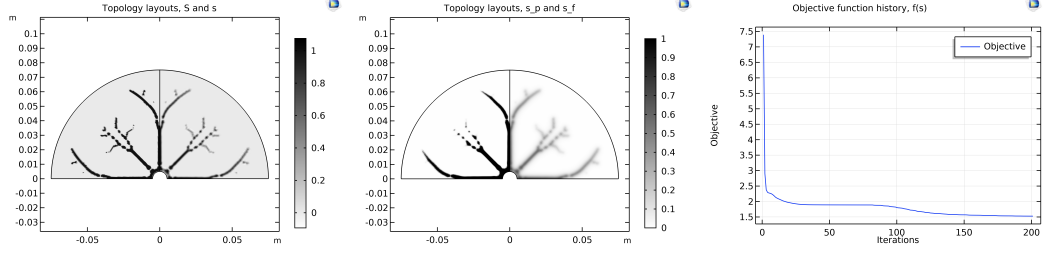


Figure 5.28: First simulation at constant  $\Gamma_2$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

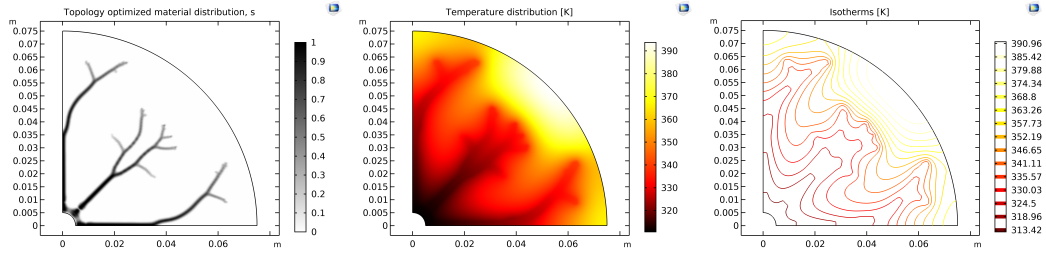


Figure 5.29: Second simulation at constant  $\Gamma_2$ . (a) Design variable (b) Temperature field (c) Temperature contour

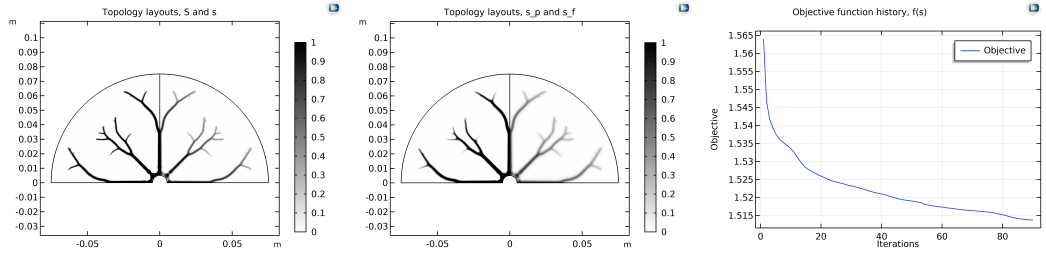


Figure 5.30: Second simulation at constant  $\Gamma_2$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

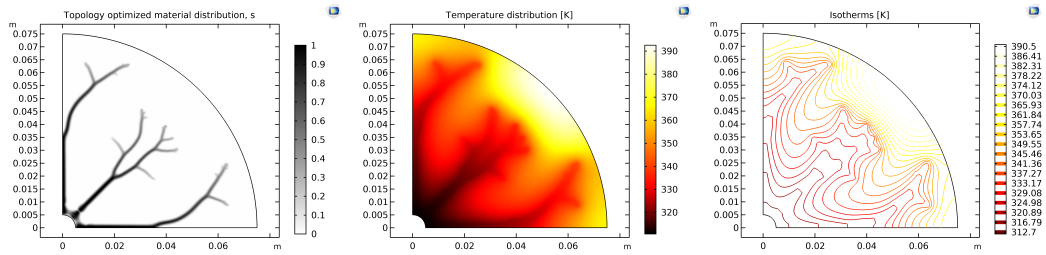


Figure 5.31: Third simulation at constant  $\Gamma_2$ . (a) Design variable (b) Temperature field (c) Temperature contour

It is useful for estimating the sensitivity of TO model to variations in the scale factor to visualize the numerical findings.

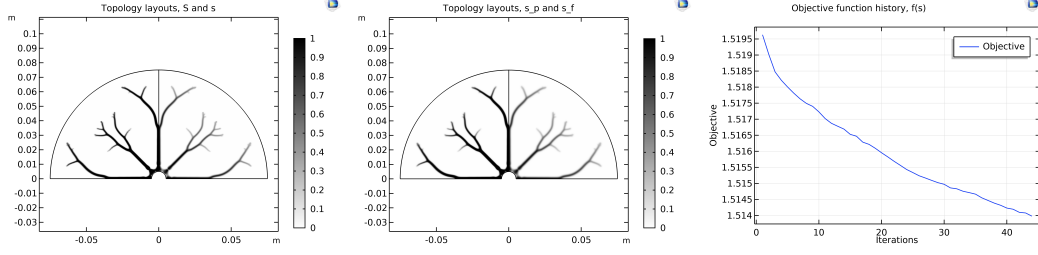


Figure 5.32: Third simulation at constant  $\Gamma_2$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

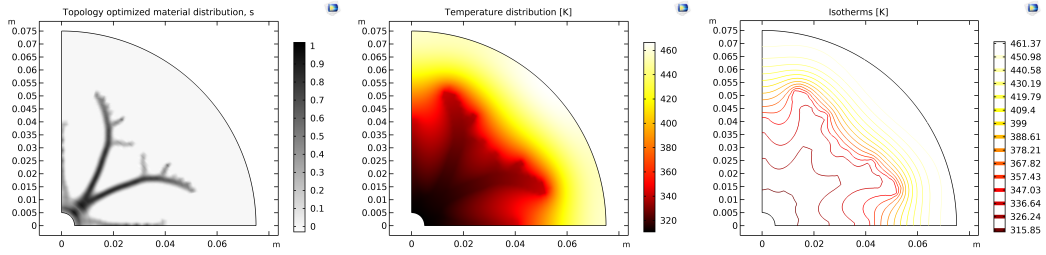


Figure 5.33: First simulation at constant  $\Gamma_3$ . (a) Design variable (b) Temperature field (c) Temperature contour

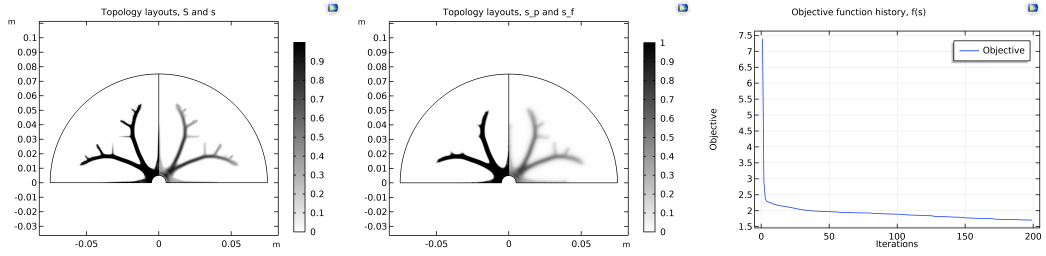


Figure 5.34: First simulation at constant  $\Gamma_3$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

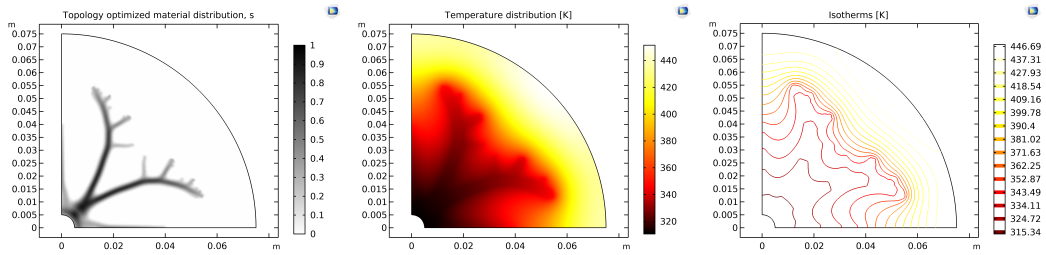


Figure 5.35: Second simulation at constant  $\Gamma_3$ . (a) Design variable (b) Temperature field (c) Temperature contour

It can be clearly observed from the above figures that both the overall mean temperature and final target function  $f(s)_{final}$  experiences a downward trend

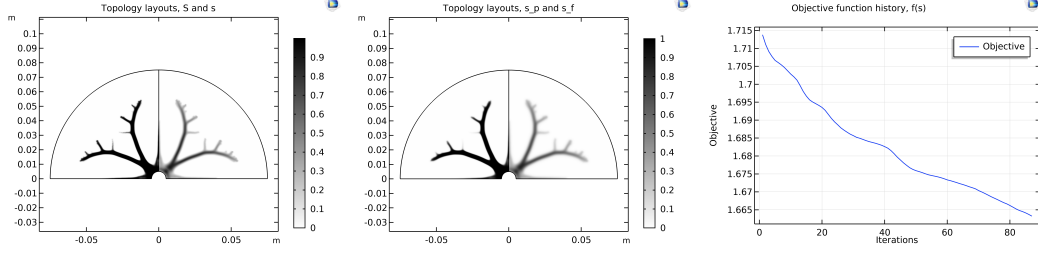


Figure 5.36: Second simulation at constant  $\Gamma_3$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

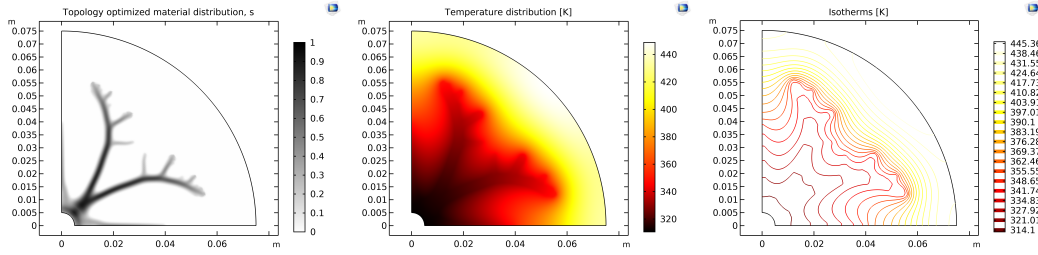


Figure 5.37: Third simulation at constant  $\Gamma_3$ . (a) Design variable (b) Temperature field (c) Temperature contour

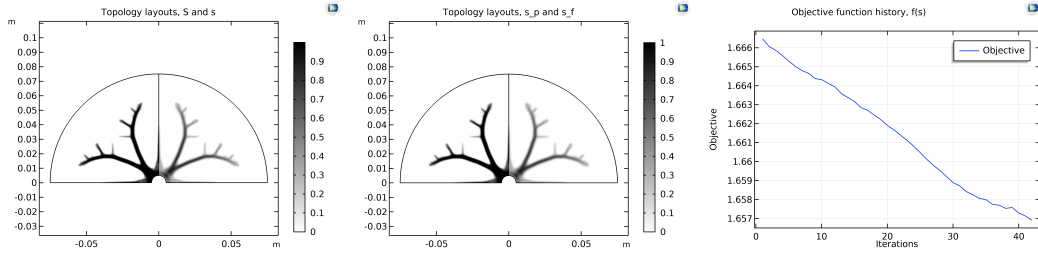


Figure 5.38: Third simulation at constant  $\Gamma_3$ . (a) Projected variable (b) Sequential coupling of filtering and projection (c) Target function

Table 5.12:  $T_{avg}$  varying values of  $\Gamma$

$T_{avg,2}$ for $\Gamma = 0.0042$	344.22[K]
$T_{avg,1}$ for $\Gamma = 0.0083$	362.19[K]
$T_{avg,3}$ for $\Gamma = 0.0167$	376.72[K]

with sharp fall of the scale factor. Therefore, a decline of this input parameter can be operated for exploring new concept designs that may better meet the design specifications. When a conceptual design is identified, it must be verified if its topology is manufacturable.

The percentage of performance improvement due to the scaling factor chosen

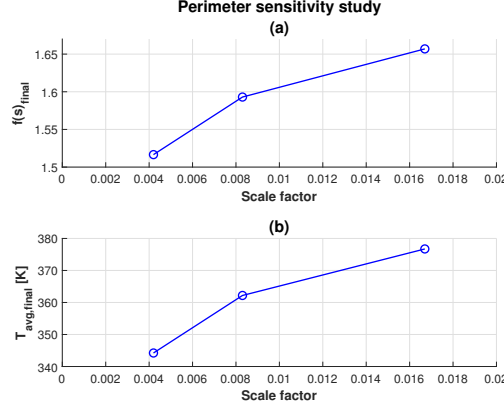


Figure 5.39: (a)  $f(s)_{final}$  (b)  $T_{avg}$

is calculated in the following manner:

$$\Delta_{T_{avg}, \Gamma_3 \rightarrow \Gamma_1} = T_{avg, \Gamma_3} - T_{avg, \Gamma_1} = 14.53[K] \quad (5.25)$$

$$\%change_{T_{avg}, \Gamma_3 \rightarrow \Gamma_1} = \frac{\Delta_{T_{avg}, \Gamma_3 \rightarrow \Gamma_1}}{T_{avg, \Gamma_3}} \cdot 100\% = 3.857\% \quad (5.26)$$

$$\Delta_{T_{avg}, \Gamma_1 \rightarrow \Gamma_2} = T_{avg, \Gamma_1} - T_{avg, \Gamma_2} = 17.97[K] \quad (5.27)$$

$$\%change_{T_{avg}, \Gamma_1 \rightarrow \Gamma_2} = \frac{\Delta_{T_{avg}, \Gamma_1 \rightarrow \Gamma_2}}{T_{avg, \Gamma_1}} \cdot 100\% = 4.961\% \quad (5.28)$$

## 5.5 Time-dependent heat conduction TO problem

### 5.5.1 Modified governing equations

The primary goal of this work is to implement a transient-state thermal diffusive TO model to search the best high conducting path, which minimizes the mean temperature over the whole computational domain.

In this time-dependent analysis the 2D geometric model remains the same as that illustrated in Fig. 5.2. What varies is the addition of one IC to impose:

$$\begin{cases} T = T_{Dir} & \text{on } \Gamma_D \\ k \nabla T \cdot n = 0 & \text{on } \Gamma_N \\ T = T_{init} & \text{in } \Omega \end{cases} \quad (5.29)$$

The time-dependent temperature field is modelled using the thermal balance equation:

$$\rho(s)C_p(s)\mathbf{u}_{trans} \cdot \nabla T + \nabla \cdot (\mathbf{q}(s) + \mathbf{q}_r) = Q + q_0 + Q_{ted} \quad (5.30)$$

with

$$\mathbf{q}(s) = -k(s)\nabla T \quad (5.31)$$

### 5.5.2 TO model

In this section, it will be proposed how to set-up the constrained TO of the transient-state heat conduction problem.

The optimization strategy is implemented in the Optimization Module of COMSOL using a quadratic discretization. Firstly, the optimization target must be selected. In the opt interface the design objective is formulated as Integral Objective function:

$$f(T, s, t) = \frac{1}{A_\Omega} \int_\Omega T(s, t) d\Omega \quad (5.32)$$

Secondly, the Control Variable Field defines the pseudo-density variable  $s = s(x, y)$  as a scalar field. The Control Variable Bounds introduce lower and upper limits corresponding to inequality constraints of the form

$$0 \leq s(x, y) \leq 1 \quad (5.33)$$

Furthermore, it is necessary to initialize the control variable to  $\gamma_{max}$ .

Thirdly, the optimization problem is subject to two inequality constraints:

$$\frac{s}{A_\Omega \cdot \gamma_{max}} = \frac{1}{A_\Omega \cdot \gamma_{max}} \int_\Omega s(x, y) d\Omega \quad (5.34)$$

$$\Gamma \cdot (d(s, x)^2 + d(s, y)^2) = \Gamma \cdot \int_\Omega |\nabla s(x, y)|^2 d\Omega \quad (5.35)$$

In this analysis, the parameters  $\gamma_{max}$  and  $\Gamma$  are tuned equal to  $1.55 \cdot \gamma_{max,1}$  and 0.0042, respectively. The volume constraint imposed is higher than  $\gamma_{max,1}$ , because if a lesser value was specified the optimization solver fails to generate topology changes. Moreover, it can be noticed that 0.0042 is the best value among those considered for the  $\Gamma$  sensitivity study.

As previously mentioned in Sec. 5.5.1, the heat conduction under time-dependent conditions can be studied using the next thermal balance equation:

$$\rho(s)C_p(T, s) \frac{\partial T}{\partial t} - k(s)\nabla^2 T = 0 \quad (5.36)$$

Here,  $\rho(s)$  and  $C_p(T, s)$  are the design-dependent mass density and effective specific heat capacity, respectively. These two material properties are formulated using the SIMP interpolation model as shown below:

$$\rho_{SIMP}(s) = \rho_{PCM} + (\rho_{Al} - \rho_{PCM})s^p \quad (5.37)$$

$$C_{p,SIMP}(T, s) = C_{p,PCM} + (C_{p,Al} - C_{p,PCM})s \quad (5.38)$$

To lighten the computational burden, this last thermophysical property is mathematically expressed employing a linear interpolation. These two mathematical models are displayed in the following figures.

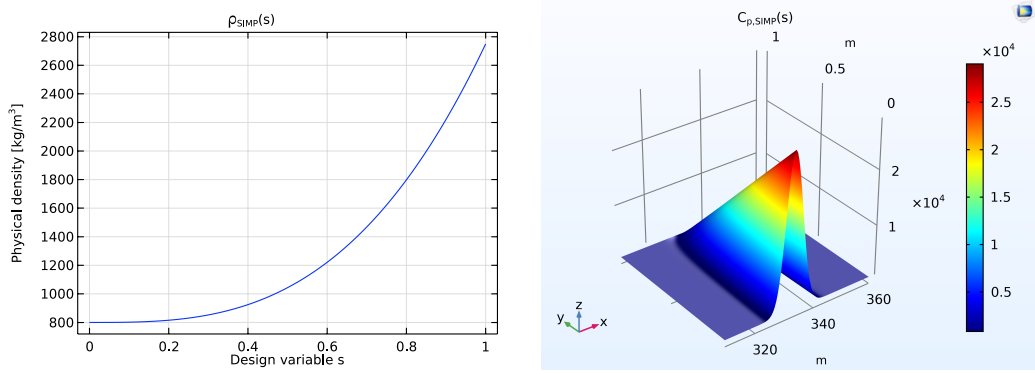


Figure 5.40: SIMP interpolation model. (a)  $\rho_{SIMP}(s)$  (b)  $C_{p,SIMP}(s)$

### 5.5.3 Double regularization strategy

In this section, it will be presented how the double regularization strategy must be implemented.

The parameters setting to employ this type of approach is reported in the next tables.

Table 5.13: Model parameters

Description	Symbol	Value in SI units
SIMP exponent	$p$	3
Volume fraction	$\gamma_{max}$	0.081851
Scale factor	$\Gamma$	0.0042
Threshold parameter	$\eta$	0.5

Table 5.14: Simulation (a) First (b) Second

Symbol	Value	Symbol	Value
$r_{fil}$	$0.5 \times h_{max}$	$r_{fil}$	$0.5 \times h_{int}$
$\beta$	32	$\beta$	16

### 5.5.4 Computational implementation

In this section, it will be discussed how to set-up the solver configuration.

The optimization problem is solved using the GCMMA. An Optimality tolerance of 0.02 and 0.01 is chosen for the first and second simulation, respectively. Moreover, a maximum allowed number of iterations is specified as second termination condition.

The MUMPS algorithm is chosen to calculate the numerical solution of the sparse linear system of discretized finite element equations.

A constant (Newton) non-linear solver is implemented. Furthermore, to promote the convergence trend of a non-linear problem the Jacobian matrix is updated at each iteration.

The time-dependent TO problem is solved using a time integration of the PDEs selecting a BDF method. This numerical scheme is chosen with a minimum and maximum order of 1 and 5, respectively. Moreover, the Nonlinear controller is implemented for an efficient time-step control.

## 5.6 Simulation results

Distributions of control variable, temperature and projected variable are presented in the next figures.

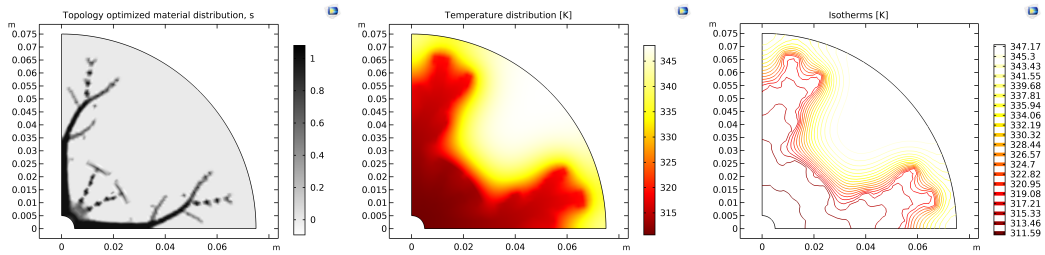


Figure 5.41: First simulation. (a) Design variable (b) Temperature field (c) Temperature contour



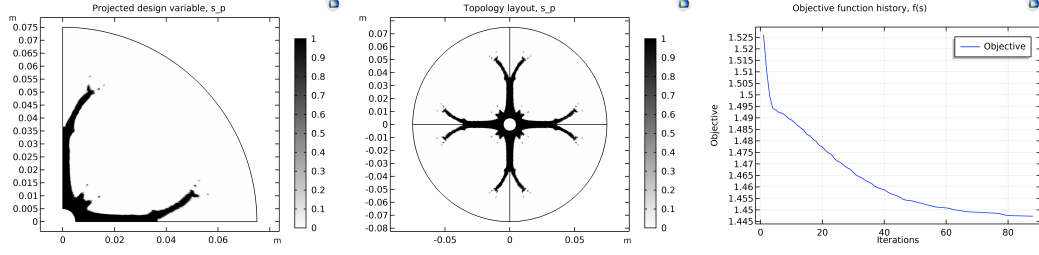


Figure 5.42: First simulation. (a) Projected variable (b) Whole layout (c) Target function

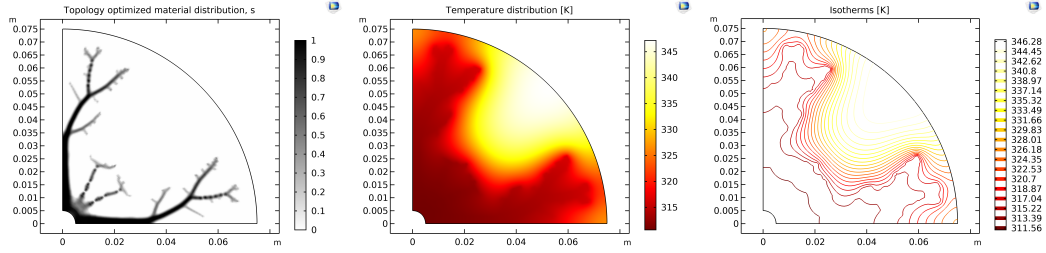


Figure 5.43: Second simulation. (a) Design variable (b) Temperature field (c) Temperature contour

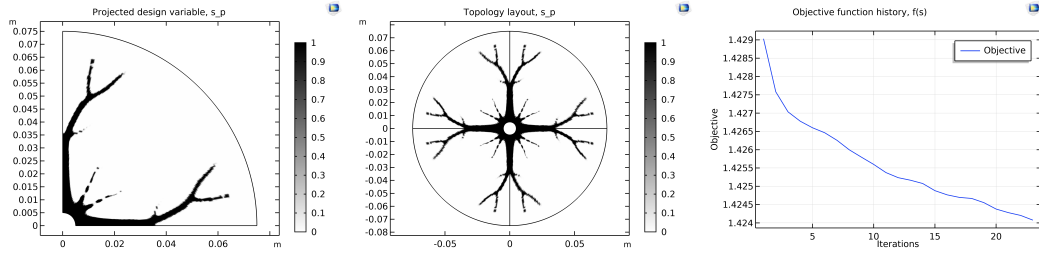


Figure 5.44: Second simulation. (a) Projected variable (b) Whole layout (c) Target function

## 5.7 Conclusion

The major results of the whole design process carried out can be summarized as follows:

- The values of the artificial density variable  $s$  are updated by using GCMMA.
- The SIMP method is used to model the thermophysical properties involved both in the steady-state and transient-state heat conduction study.
- The application of the triple regularization strategy allows to transform a topology with gray transition regions into a binary black and white optimal design, overcoming issues posed by the SIMP technique.

- The grid refinement during the TO process leads to greater resolution of the final high conductivity material distribution over the design domain.
- The TO problem results in performance enhancement due to minimization of the design target.
- A validation phase to guarantee the coral-like structure fitness is required. This further analysis will be performed in the next chapter.
- To better understand how variation of both the prescribed fin cross-sectional area fraction and scaling factor may affect the design objective a sensitivity study is conducted.
- At the end of the optimization procedure, a design engineer can find the optimum balance of performance targets to reach the desired goal in the best possible way.

Considering all points discussed above, it is possible to conclude that the stationary thermal diffusive TO model can be effectively used for running a topological design optimization and obtaining a numerical solution within a reasonable calculation time.

It is also important to remember that the discharging phase of a LHT-ESS is time-dependent. Therefore, the transient-state study that can deeply capture further details of the physics under investigation is carried out.

# Chapter 6

## Topology optimized design validation

### 6.1 Stationary validation protocol

#### 6.1.1 Finned geometry generation

In this section, it will be analyzed the validation of steady-state TO model used for obtaining the optimal tree-like topology.

It is necessary to establish if the numerical solution ensures that the topological modelling is accurate, considering all design constraints. For this reason, it is built a 2D geometric model of a LHTESS that includes:

- A high conductive metal path obtained from TO process.
- A storage tank filled with PCM, which generates a volumetric heat.

To build the fin geometry a **Filter Data Set** is applied to both the density-like field  $s$  and projected design field  $s_p$ , as shown below.

This tool works setting a lower bound equal to the projection threshold  $\eta$ . The amounts of solid materials make up around 5,46% and approximately 8,90% of the total cross-sectional area, respectively.

As can be seen in Fig. (a) and (b) 6.1, the primary difference between SIMP approach and triple regularization technique involves manufacturability of the topologically optimized structures obtained. In fact, the fin geometry corresponding to artificial density field is impossible to fabricate, because of empty space. Consequently, although this geometric configuration is cleaned and smoothed, may not be chosen for the subsequent validation phase. In contrast, when the Helmholtz filter and projection are sequentially applied three times, the resulting optimized topology is not constituted by

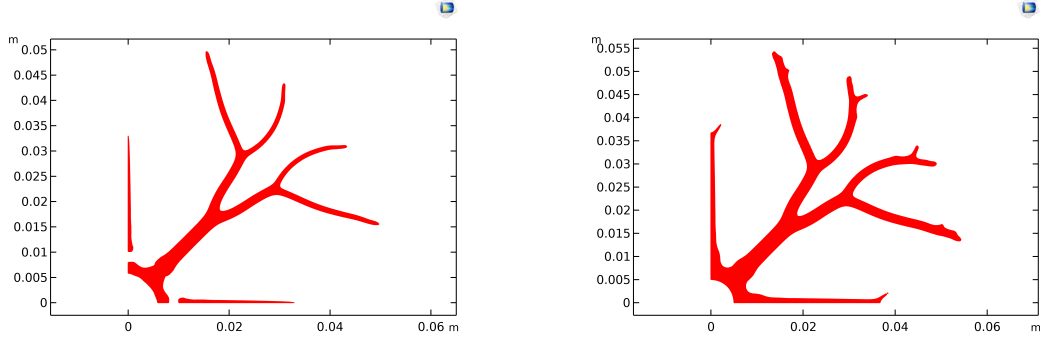


Figure 6.1: Filter applied to (a)  $s$  (b)  $s_p$

voids and significant geometric irregularities. Therefore, the validation study is conducted using this last coral-like structure.

After the application of the Filter Data Set a **Mesh Part** may be created.

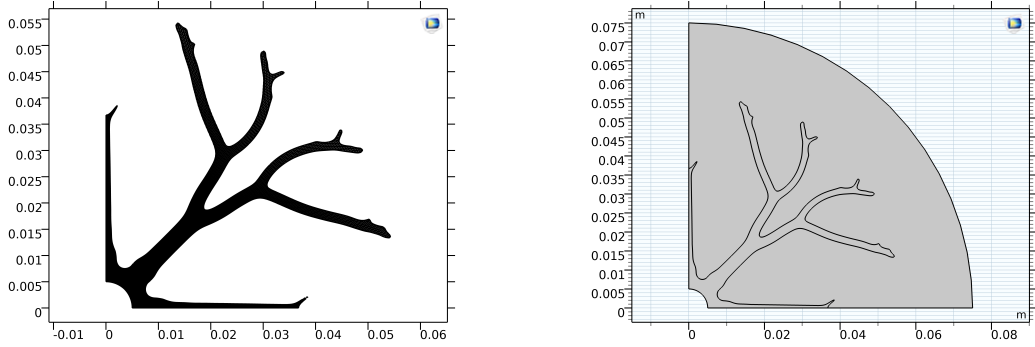


Figure 6.2: (a) Mesh part (b) Geometric configuration

This mesh part is imported in the **Geometry Interface** to generate the overall fin layout, as sketched in Fig. (b) 6.2.

### 6.1.2 Thermal field modelling

The mathematical modelling consists in using the energy balance equation for steady-state problem, imposing boundary conditions and assigning material properties. The reader can consult Sec. 3.3 to review how to implement a stationary heat conduction analysis.

The boundary conditions that must be set up are reported below.

$$\begin{cases} -k\nabla^2 T = Q_{GEN} & \text{in } \Omega \\ T = T_{Dir} & \text{on } \Gamma_D \\ k\nabla T \cdot n = 0 & \text{on } \Gamma_N \end{cases} \quad (6.1)$$

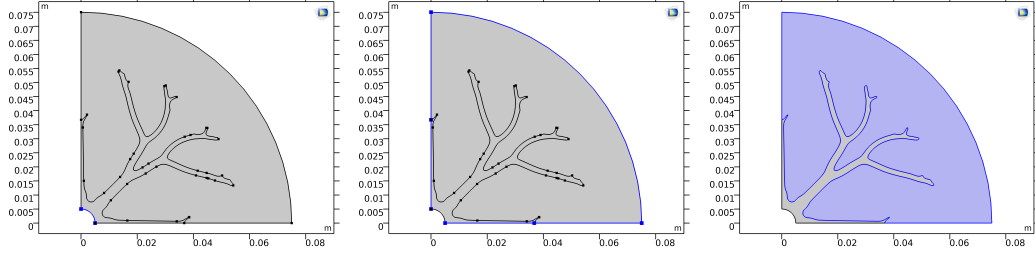


Figure 6.3: (a) Dirichlet BC (b) Neumann BC (c) Volumetric heat source

The solution domain is meshed with free triangular elements. For the finned region a mesh size of  $0.5[mm]$  is used. Conversely, for the storage space an element size of  $1[mm]$  is applied. Since in the main trunk of the tree-like structure greater temperature gradients are expected, a higher grid density is required. The meshing of the whole 2D computational domain is illustrated in the next figure.

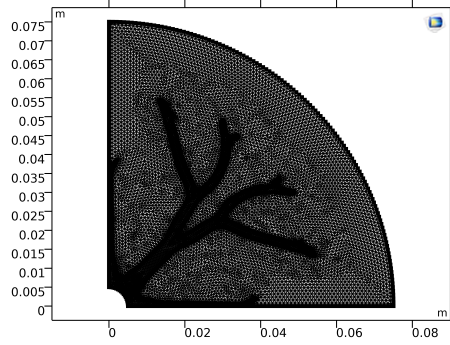


Figure 6.4: Mesh scheme

## 6.2 Simulation results

The validation study is carried out for benchmarking performance and hence to provide a measure of improvement in enhancing the heat transfer.

As a result of the TO process, the finned region produces a thermal performance improvement with respect to the SO procedure. The numerical results are compared in the table below.

The temperature profile and isothermal contours obtained from the validation analysis are depicted in the following figures.

The percent drop in mean temperature over the entire computational domain

Table 6.1: Numerical solutions

$T_{avg,TO}$	362.19 [K]
$T_{avg,val}$	350.85 [K]
$T_{avg,SO}$	369.64 [K]

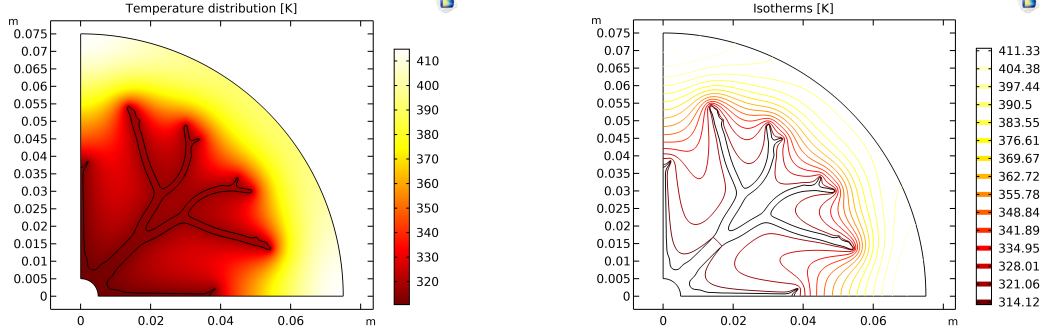


Figure 6.5: (a) Temperature distribution (b) Isotherms

can be estimated using the next formulas:

$$\%_{change, T_{avg}} = \left| \frac{T_{avg,TO} - T_{avg,val}}{T_{avg,val}} \right| 100\% = 3.23\% \Rightarrow \chi_{T_{avg}} = 96,77\% \quad (6.2)$$

Therefore, it can be concluded that the optimal layout structure determined by TO method is more thermally efficient than the optimum shape discovered performing the SO procedure. If the fin cross-sectional area obtained from the TO approach is greater than that maximum prescribed a further design optimization step will be necessary.

### 6.3 Sequential coupling of Topology and Shape Optimization

One major design challenge impacting on the performance target is the prescribed surface area that should be occupied by the metal fins. If it is not possible to satisfy the volume fraction constraint, a sequential coupling of **Topology and Shape Optimization (TSO)** is required. To evaluate this specific requirement, surface integral of the metal fins area is calculated. The numerical outcomes are tabulated in the table below.

Since finned zone area, extracted from TO, is jumped beyond the target volume of material  $\gamma_{max}$  by more than 150%, a strong reduction of this surface area, while conserving validity of the geometric model, is required.

Table 6.2: Fins surface area

Description	Symbol	Value
Design value	$FinSurf_{design}$	$2.4[cm^2]$
Surface area computed	$A_{fin,TO}$	$3.8472[cm^2]$

This integrated procedure allows to increase the domain region filled with PCM decreasing the domain area occupied by the resulting fin geometry obtained from TO to meet all the design requirements set.

### 6.3.1 Stationary heat conduction SO problem

The scope of SO is to adjust the geometric configuration of the final optimized topology derived from the combination of Filter Data Set, Mesh part and Import Geometry feature. The SO approach is based upon the COMSOL User's Guide [10].

To accomplish the purposes of the SO process, the following two features are required:

- **Free Shape Domain** to solve an equation that smoothly deforms the finned domain. The **Yeoh** model is selected as smoothing technique due to its numerical stability, even when large deformations occur.
- **Free Shape Boundary** to define a control variable for modifying the fins shape. Using this feature, it is possible to specify the finned region boundary and impose upper bounds of the optimization variables.

The prescribed volume fraction constraint can be fulfilled simply by tuning parameters. Meeting or exceeding the design specifications set depends on two factors. These include the maximum displacement  $d_{max}$  and filter radius  $R_{min}$ . These two parameters severely impact on success or failure of a numerical simulation. If they overcome certain values an error message appears, because  $d_{max}$  is too high and  $R_{min}$  is too low. Consequently, the TSO procedure helps find the best design, but it is necessary to consider the effects of influencing factors.

The features needed for implementing the SO approach are displayed in the figures below and simulation parameters tuned are tabulated in Tab. 6.3.

Since the current optimization problem aims to minimize the average temperature over the entire solution domain  $T_{avg}$  by declining surface area of metal fins, the integral objective is the same as that used for the TO

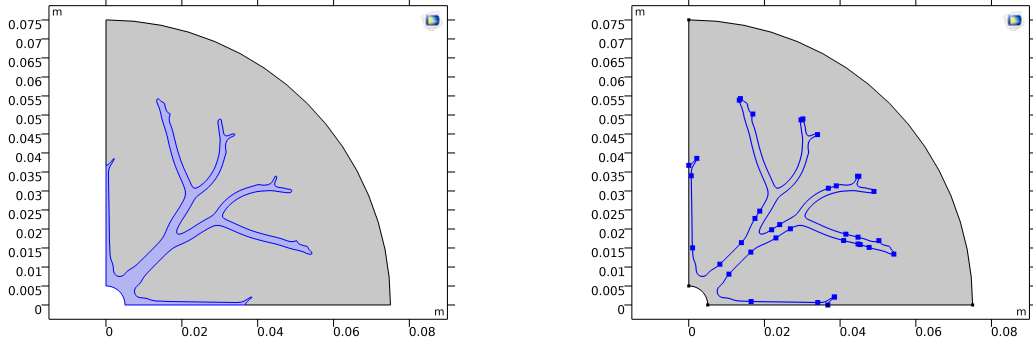


Figure 6.6: (a) Free Shape Domain (b) Free Shape Boundary

Table 6.3: Optimization factors

Symbol	Value
$d_{max}$	0.001 [m]
$R_{min}$	0.015 [m]

approach. To reach an amount of aluminium less than or equal to the prescribed volume fraction, which comprises about 5,46% of the total design domain, the following inequality constraint must be satisfied:

$$\frac{comp2.intop2(1)}{FinSurf} \leq 1 \quad (6.3)$$

Here,  $comp2.intop2(1)$  equals the resulting surface of the finned region obtained from SO.

Fig. (a) and (b) 6.7 depict the shape optimized design and objective function, respectively.

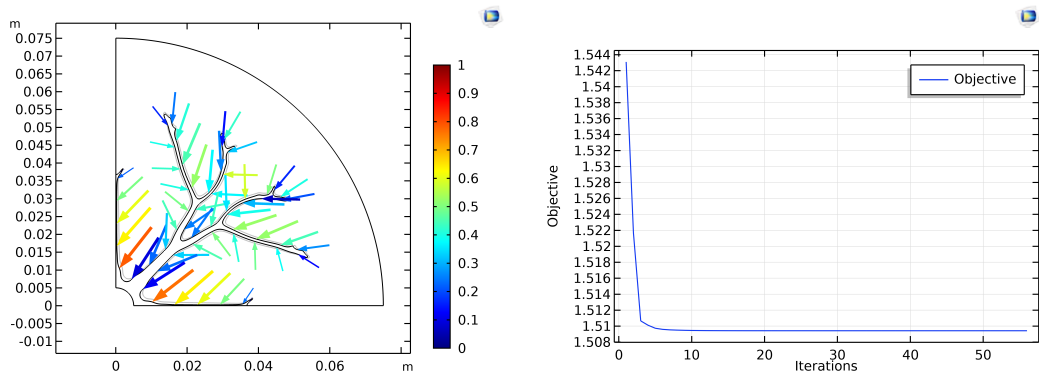


Figure 6.7: (a) Optimized shape (b) Target function

Once the SO process is completed, the surface integral over finned zone can be computed. Numerical result obtained is listed in the following table.



Table 6.4: Fins surface area

Description	Symbol	Value
Previous fins area	$A_{fin,TO}$	$3.8472[cm^2]$
Surface area calculated	$A_{fin,TSO}$	$2.3431[cm^2]$

As a result of the SO procedure, the surface area  $A_{fin}$  experiences a sharp fall. The percent decrease is estimated as follows

$$\Delta_{A_{fin}} = A_{fin,TO} - A_{fin,TSO} = 1.5041[cm^2] \quad (6.4)$$

$$\%_{change,A_{fin}} = \frac{\Delta_{A_{fin}}}{A_{fin,TO}} \cdot 100\% = 39.096\% \quad (6.5)$$

Undergoing reduction just over 39%, it may be concluded that adjusting optimization variables settings  $A_{fin,TSO}$  completely fulfills the volume fraction requirement. Therefore, the thermal system behaves as intended, while satisfying all the design constraints.

At this design phase, comparison of performance target achieved by implementing only SO or combining TO with SO is necessary to highlight the solution fitness. The design objectives obtained at the end of both the whole design procedure are compared in Tab. 6.5 .

Table 6.5: TSO versus SO

Symbol	Value
$T_{avg,TSO}$	$355.35[K]$
$T_{avg,SO}$	$369.64[K]$

The overall average temperature  $T_{avg,TSO}$  is dropped of nearly 3.87% with respect to  $T_{avg,SO}$ . The percent reduction is calculated using before Eq. 6.16 and after Eq. 6.11.

$$\Delta_{T_{avg}} = T_{avg,SO} - T_{avg,TSO} = 14.29[K] \quad (6.6)$$

$$\%_{change,T_{avg}} = \frac{\Delta_{T_{avg}}}{T_{avg,SO}} \cdot 100\% = 3.866\% \quad (6.7)$$

It is verified that the thermal system is well-designed, because it provides significant performance improvements when it is benchmarked with that optimized following SO approach, even though it uses a lower amount of aluminium.

## 6.4 Time-dependent validation protocol

### 6.4.1 Finned geometry generation

In this section, it will be investigated the validation of transient-state TO model used for obtaining the optimum coral-like topology.

As indicated in Sec. 6.1.1, to create the fin configuration a **Filter Data Set** is applied to both the pseudo density field  $s$  and projected design field  $s_p$ , as depicted below.

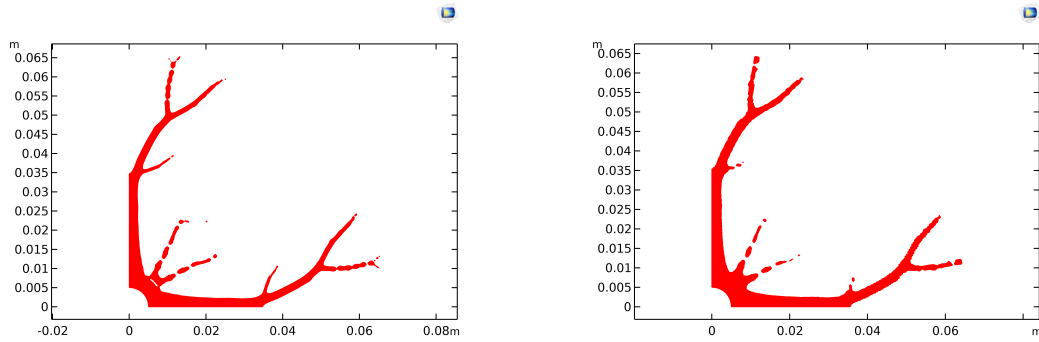


Figure 6.8: Filter applied to (a)  $s$  (b)  $s_p$

This tool works setting a lower bound equal to the projection threshold  $\eta$ . The amounts of solid materials constitute about 8,46% and nearly 8,51% of the total cross-sectional area, respectively.

The following validation analysis will be carried out employing the second finned layout.

After the application of the Filter Data Set a **Mesh Part** can be generated. This mesh part is imported in the **Geometry Interface** to build the overall fin geometry and then geometric irregularities are deleted using the **Collapse Edges** feature as illustrated in the following figures.

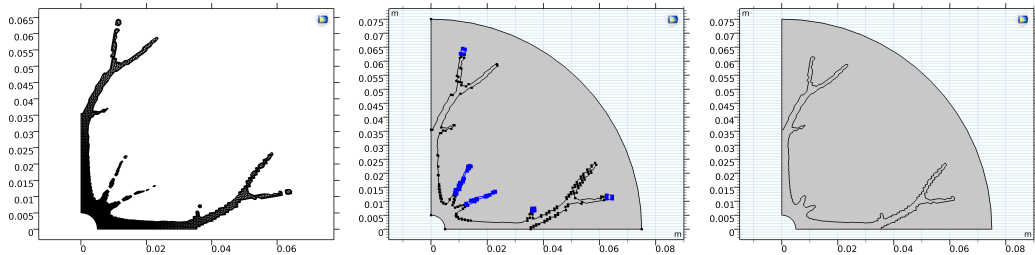


Figure 6.9: (a) Mesh part (b) Collapse Edges (c) Geometric configuration

### 6.4.2 Thermal field modelling

The mathematical modelling employed in studying temperature distribution involves the thermal energy balance for transient-state problem, application of boundary conditions and assignment of material properties. The Sec. 3.7.2 provides implementation of a time-dependent heat conduction analysis.

The boundary conditions that must be imposed are indicated below.

$$\begin{cases} T = T_{Dir} & \text{on } \Gamma_D \\ k \nabla T \cdot n = 0 & \text{on } \Gamma_N \\ T = T_{init} & \text{in } \Omega \end{cases} \quad (6.8)$$

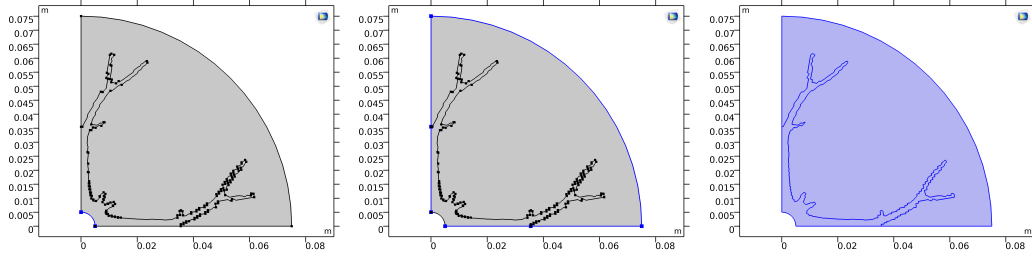


Figure 6.10: (a) Dirichlet BC (b) Neumann BC (c) Initial temperature

The solution domain is meshed with 4609 triangular elements. The grid sizes in the finned structure and storage tank are  $0.5[mm]$  and  $1[mm]$ , respectively. The mesh generated for the entire numerical domain is shown in the figure below.

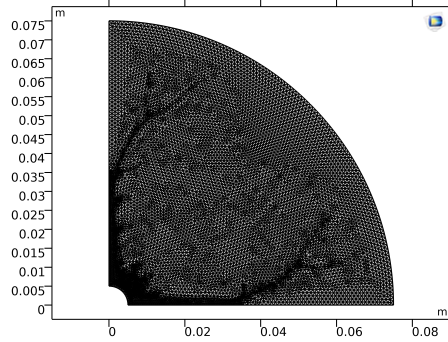


Figure 6.11: Mesh scheme

## 6.5 Simulation results

The primary aim of the whole validation study is to come up with a computational model, which is validated in terms of both the thermal response and total amount of material available.

The validation analysis confirms that the TO approach, compared to the SO process, achieves a thermal performance improvement. The computational outcomes are reported in the next table.

Table 6.6: Numerical solutions

$T_{avg,TO}$	323.79 [K]
$T_{avg,val}$	326.89 [K]
$T_{avg,SO}$	328.98 [K]

The resulting temperature field distribution and isothermal contours are displayed in the following figures.

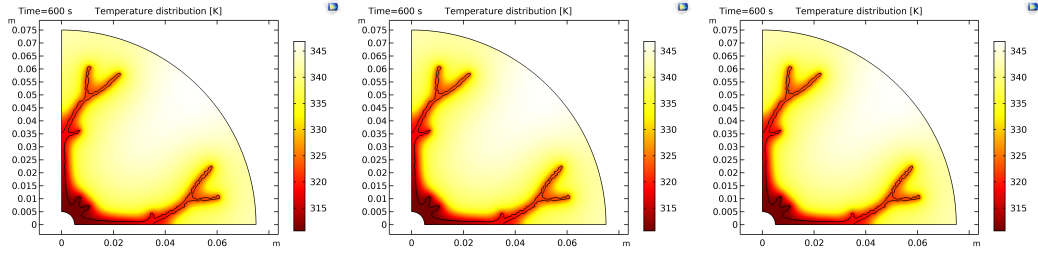


Figure 6.12: Temperature fields. (a)  $t = 600[s]$  (b)  $t = 1800[s]$  (c)  $t = 3600[s]$

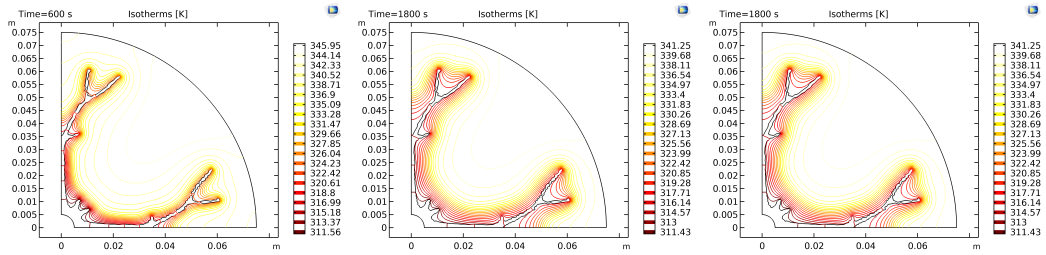


Figure 6.13: Temperature contours. (a)  $t = 600[s]$  (b)  $t = 1800[s]$  (c)  $t = 3600[s]$

The percent rise in mean temperature over the whole solution domain can be evaluated as follows:

$$\%_{change, T_{avg}} = \left| \frac{T_{avg, val} - T_{avg, TO}}{T_{avg, TO}} \right| 100\% = 0.948\% \Rightarrow \chi_{T_{avg}} = 99,052\% \quad (6.9)$$

In conclusion, simulation results show that the finned geometry corresponding to the optimum topology obtained following the workflow described in Sec. 6.4.1 provides an enhanced thermal penetration with respect to the optimal shape design reached at the end of the SO procedure. If the cross-sectional area covered by fins is higher than the maximum area fraction constraint a subsequent design optimization phase will be necessary.

### 6.5.1 Time-dependent heat conduction TSO problem

As discussed in Sec. 6.3.1, both the Free Shape Domain and Free Shape Boundary features are used to investigate design modification through SO.

To meet the fin cross-sectional area specified, it is necessary to set-up both the maximum displacement  $d_{max}$  and filter radius  $R_{min}$ .

The features required for employing the SO process are visualized in the next figures and simulation parameters tuned are listed in Tab. 6.7.

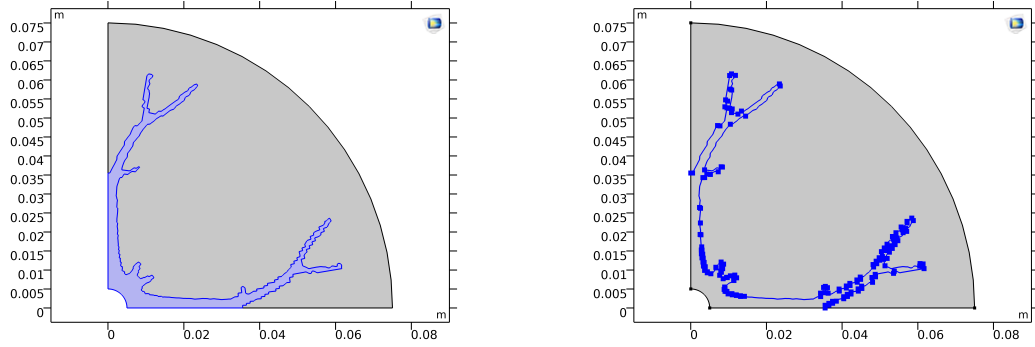


Figure 6.14: (a) Free Shape Domain (b) Free Shape Boundary

Table 6.7: Optimization factors

Symbol	Value
$d_{max}$	0.0015 [m]
$R_{min}$	0.035 [m]

Both the integral objective and inequality constraint selected for this SO problem are the same as those indicated in Sec. 6.3.1.

Fig. (a) and (b) 6.7 illustrate the shape optimized design and design objective, respectively.

The SO procedure completed reveals that the surface integral of the finned region fulfills the volume fraction constraint, as tabulated in the following table.

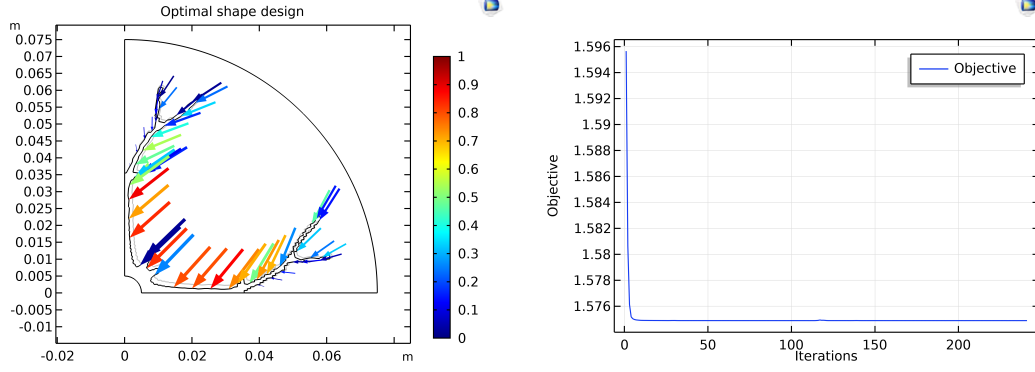


Figure 6.15: (a) Optimized shape (b) Target function

Table 6.8: Fins surface area

Description	Symbol	Value
Previous fins area	$A_{fin,TO}$	$3.6583[cm^2]$
Surface area calculated	$A_{fin,TSO}$	$2.3918[cm^2]$

As a result of the SO procedure, the surface area  $A_{fin}$  experiences a significant drop. The percent decrease is calculated using the next formulas

$$\Delta_{A_{fin}} = A_{fin,TO} - A_{fin,TSO} = 1.2583[cm^2] \quad (6.10)$$

$$\%_{change, A_{fin}} = \frac{\Delta_{A_{fin}}}{A_{fin,TO}} \cdot 100\% = 34.396\% \quad (6.11)$$

Undergoing reduction just above 34%, the maximum area fraction constraint is fully satisfied.

The comparison between SO and TSO demonstrates that the integrated T- and SO technique ensures to obtain the best possible outcome. The design objectives achieved at the end of both the entire design procedure are compared in Tab. 6.9.

Table 6.9: TSO versus SO

Symbol	Value
$T_{avg,TSO}$	$328.79[K]$
$T_{avg,SO}$	$328.98[K]$

The temperature field distributions and isothermal contours are depicted in the following figures. The mean temperature over the whole numerical

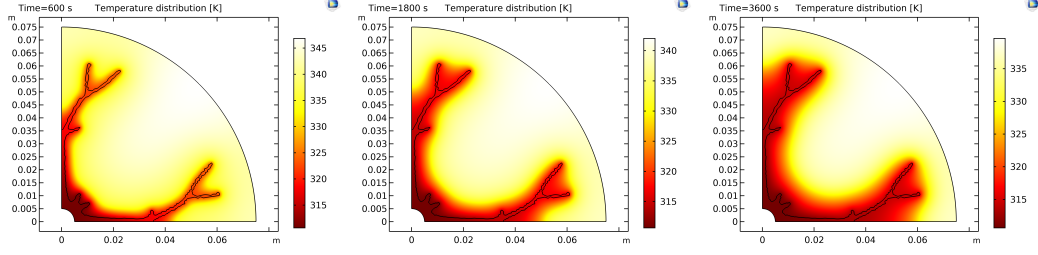


Figure 6.16: Temperature fields. (a)  $t = 600[s]$  (b)  $t = 1800[s]$  (c)  $t = 3600[s]$

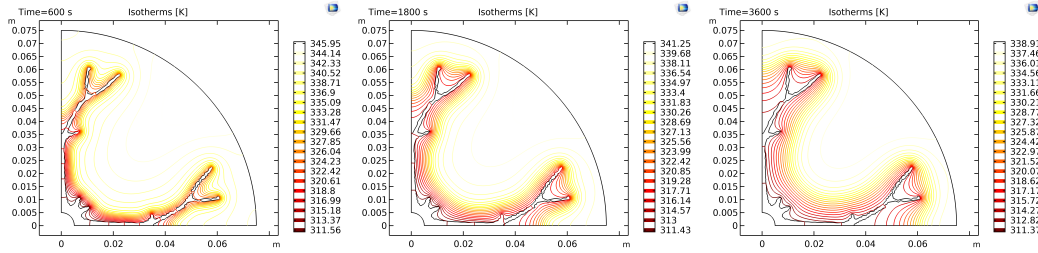


Figure 6.17: Temperature contours. (a)  $t = 600[s]$  (b)  $t = 1800[s]$  (c)  $t = 3600[s]$

domain is found to be 328.79 [K] after 3600 seconds from the beginning of the discharging mode.

The liquid fraction evolution during the solidification process is illustrated in the next figures.

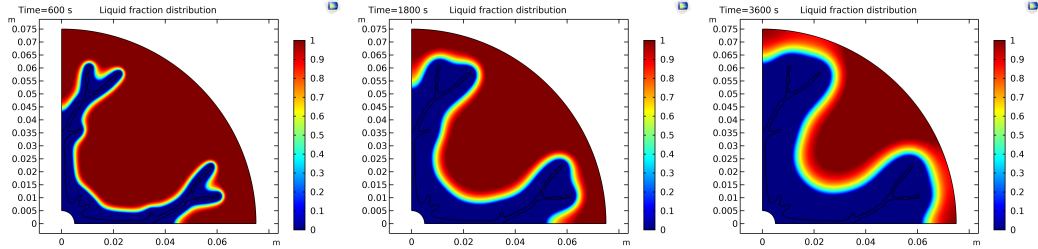


Figure 6.18: Liquid fraction fields. (a)  $t = 600[s]$  (b)  $t = 1800[s]$  (c)  $t = 3600[s]$

The heat transfer variables are investigated to map the thermal performance. Average temperature history and thermal energy storage density are chosen as key performance characteristics as observed in Sec. 3.8.

Additionally, to the previous performance indicators evaluated, the overall internal energy change and global energy release by the PCM, during its discharging mode, may be calculated as follows:

$$\Delta U_{tot,PCM} = \int_{T_{ref}}^T C_{p,PCM} dT + \sigma L_{PCM} \quad (6.12)$$

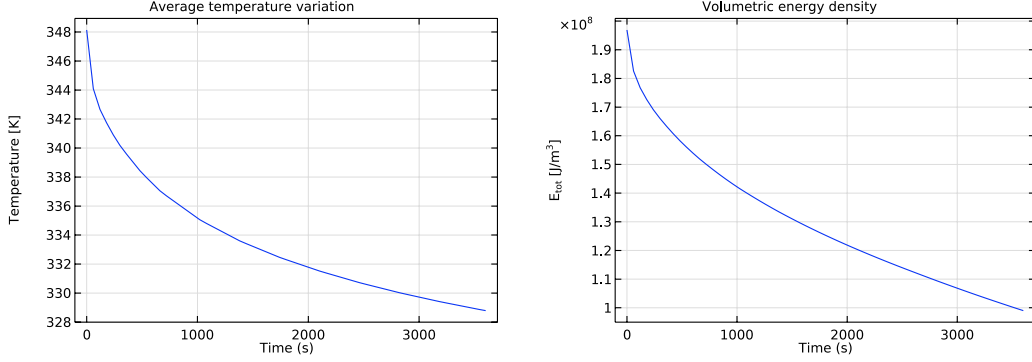


Figure 6.19: (a)  $T_{avg}$  evolution (b)  $E_{tot}$  released

$$\Delta E_{tot,PCM} = \rho_{PCM} \left( \int_{T_{ref}}^T C_{p,PCM} dT + \sigma L_{PCM} \right) \quad (6.13)$$

The overall internal energy change is equal to  $1.2238 \cdot 10^5 [J/kg]$ , while the global energy release is  $0.979 \cdot 10^8 [J/m^3]$ .

As a result of the TSO approach, both the overall internal energy change and global energy release show a growth. The percent increases are computed in the following manner

$$\Delta U_{tot,PCM} = U_{tot,PCM,TSO} - U_{tot,PCM,SO} = 1.2238 \cdot 10^5 - 1.1866 \cdot 10^5 = 0.0372 [J/kg] \quad (6.14)$$

$$\%_{change, U_{tot,PCM}} = \frac{\Delta U_{tot,PCM}}{U_{tot,PCM,SO}} \cdot 100\% = 3.135\% \quad (6.15)$$

$$\Delta E_{tot,PCM} = E_{tot,PCM,TSO} - E_{tot,PCM,SO} = 0.979 \cdot 10^8 - 0.9493 \cdot 10^8 = 0.0297 [J/m^3] \quad (6.16)$$

$$\%_{change, E_{tot,PCM}} = \frac{\Delta E_{tot,PCM}}{E_{tot,PCM,SO}} \cdot 100\% = 3.129\% \quad (6.17)$$

The reason of the differences in the numerical results with respect to those presented in Sec. 3.8 is caused by the use of one-quarter storage element.

## 6.6 Robustness of the TSO technique

Robustness is a very crucial attribute of a computational model under perturbations that must be guaranteed. In the context of design optimization,



robustness means that changing a certain input data the output variation is minimal or within a specified bound.

The TSO strategy developed ensures that both the stationary and time-dependent analyses are insensitive to changes of the maximum volume fraction specified and hence the study is robust. In fact, varying parameter  $\gamma_{max}$  the final fin cross-sectional area covered by fins is always less than the prescribed area fraction. This constraint is only overcome, but remaining within an error margin of +1%, in the case of  $\gamma_{max,3}$ .

The shapes of the other two resulting optimized designs are visualized below.

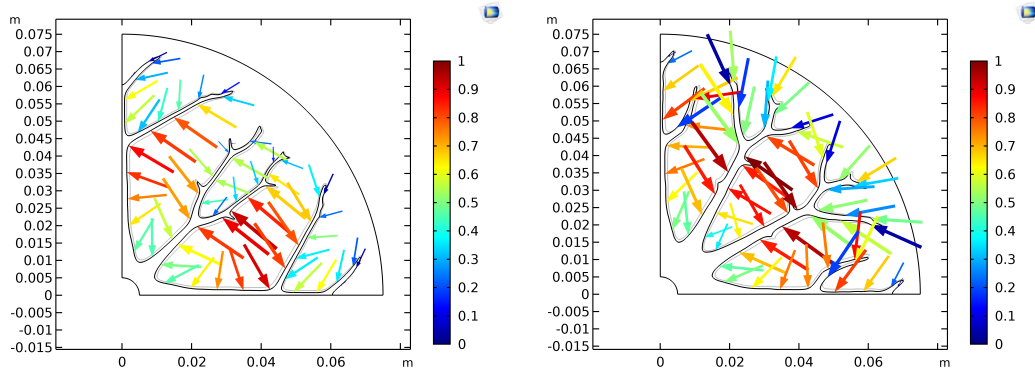


Figure 6.20: (a) Optimized shape for  $\gamma_{max,2}$  (b) Optimized shape for  $\gamma_{max,3}$

Surfaces areas of the previous finned configurations are listed in the next table.

Table 6.10: Fin surface area

$\gamma_{max,2} =$	$A_{TSO,2} = 4.6838cm^2$
$\gamma_{max,3} =$	$A_{TSO,3} = 7.2333cm^2$

## 6.7 Conclusion

Considering the computational outcomes of the TSO technique it can be concluded that the most important phenomena involved in the operation of the TES systems are captured. Therefore, this strategy developed may be used to conduct simulations of the operating modes of a LHTESS monitoring the evolution of the main energy indicators.

The whole strategy developed leads to an optimized topology, which is coherent with all the design specifications and thus with the study aims.

Indeed, if a TO process produces topologically optimized fins that exceed the volume fraction requirement, the subsequent SO fixes this problem in a satisfactory way. Consequently, this approach represents a helpful tool for design optimization of thermal systems with conductive fins, such as TES unit.

# Bibliography

- [1] Tobe F., Design optimization: Topology and much more, Design World September (2015) 54-56.
- [2] Pluke A, How to optimize performance and manufacturability for additive manufacturing, Design World August (2020) 94-97.
- [3] Sarkar D., Optimization in Chemical Engineering (NPTEL), Indian Institute of Technology Kharagpur (2018).
- [4] Maiti J., Design and Analysis of Experiments (NPTEL), Indian Institute of Technology Kharagpur (2017).
- [5] Giordano F.R., Fox W.P., Horton S.B., A First Course in Mathematical Modeling, Books/Cole Publishing Company 5th Edition (2013) 289-291.
- [6] Andersson B., Wei J., Optimization in one dimension, Uppsala University (2014) 4-19.
- [7] Heath M.T., Scientific Computing. Lecture on Optimization Problems, University of Illinois at Urbana-Champaign (2002) 2-63.
- [8] Sheikholeslami M., Applications of Nanofluid Transportation and Heat Transfer Simulation, IGI Global 1st Edition (2018) 249-262.
- [9] Heat Transfer Module User's Guide, COMSOL Multiphysics 5.5 (2019) 60-67.
- [10] Optimization Module User's Guide, COMSOL Multiphysics 5.5 (2019) 57-60.
- [11] Lohrasbi S., Gorji-Bandpy M., Gangi D.D., Response surface method optimization of V-shaped fin assisted latent heat thermal energy storage system during discharging process, Alexandria Engineering Journal 55 (2016) 2065-2076.

- [12] Yang Z., Chen L., Li Y., Xia Z., Wang C., Numerical investigation of heat transfer characteristics in a shell-and-tube latent heat thermal energy storage system, *Energy Procedia* 160 (2019) 475-482.
- [13] Darzi A.A.R., Jourabian M., Jourabian M., Melting and solidification of PCM enhanced by radial conductive fins and nanoparticles in cylindrical annulus, *Energy Conversion and Management* 118 (2016) 253-263.
- [14] Lohrasbi S., Gorji-Bandpy M., Ganji D., Thermal penetration depth enhancement in latent heat thermal energy storage system in the presence of heat pipe based on both charging and discharging processes, *Energy Conversion and Management* 148 (2017) 646-667.
- [15] Sheikholeslami M., Lohrasbi S., Domiri Ganji D., Numerical Analysis of Discharging Process Acceleration in LHTESS by Immersing Innovative Fin Configuration Using Finite Element Method, *Applied Thermal Engineering* 107 (2016) 154-166.
- [16] You Y., Zhao Y., Zhao C., Liu H., The topology optimization of the fin structure in latent heat storage (in English), *Chinese Science Bulletin* 64 (2019) 1191-1199.
- [17] Haertel J.H.K., Engelbrecht K., Lazarov B.S., Sigmund O., Topology Optimization of a Pseudo 3D Thermofluid Heat Sink Model, *International Journal of Heat and Mass Transfer* 121 (2018) 1073-1088.
- [18] Ikonen T.J., Marck G., Söbester A., Keane A.J., Topology optimization of conductive heat transfer problems using parametric L-systems, *Structural and Multidisciplinary Optimization* 58 (2018) 1899–1916.
- [19] Manuel M.C.E., Lin P.T, Heat Exchanger Design with Topology Optimization (Chapter 4), (2017) 61-72.
- [20] Deaton J.D., Grandhi R.V., A survey of structural and multidisciplinary continuum topology optimization: post 2000, *Structural and Multidisciplinary Optimization* 49 (2014) 1–38.
- [21] Verbart A., Van Keulen F., Langelaar M., Topology Optimization with Stress Constraints, TU Delft (2015).
- [22] Rojas Labanda S., Mathematical programming methods for large-scale topology optimization problems, DTU Wind Energy (2015).

- [23] Koga A., Lopes E., Villa Nova H., R. de Lima C., Silva E., Development of heat sink device by using topology optimization, *International Journal of Heat and Mass Transfer* 64 (2013) 759-772
- [24] Christensen P., Klarbring A., *An Introduction to Structural Optimization* (Chapter 4), (2009) 66-69.
- [25] Burger F.H., Dirker J., Meyer J., Topology Optimisation for the Volume-to-Surface Problem in a Three-Dimensional Cubic Domain Using Conduction Cooling, *International Journal of Heat and Mass Transfer* 67 (2013) 214-224.
- [26] Rojas P.A.M., *Computational Modeling, Optimization and Manufacturing Simulation of Advanced Engineering Materials*, (2016) 210-212.
- [27] Zhang B., Zhu J., Xiang G., Gao L. Design of nanofluid-cooled heat sink using topology optimization. *Chinese Journal of Aeronautics* (2020) 5-17.
- [28] Wang F., Lazarov B.S., Sigmund O., On projection methods, convergence and robust formulations in topology optimization. *Struct Multidisc Optim* 43 (2011) 767–784.
- [29] Soprani S., Haertel J., Lazarov B., Sigmund O., Engelbrecht K., A design approach for integrating thermoelectric devices using topology optimization. *Applied Energy* 176 (2016) 49-64.
- [30] Christiansen R.E., Lazarov B.S., Jensen J.S., Sigmund O., Creating geometrically robust designs for highly sensitive problems using topology optimization: Acoustic cavity design. *Structural and Multidisciplinary Optimization* 52 (2015) 737-754.