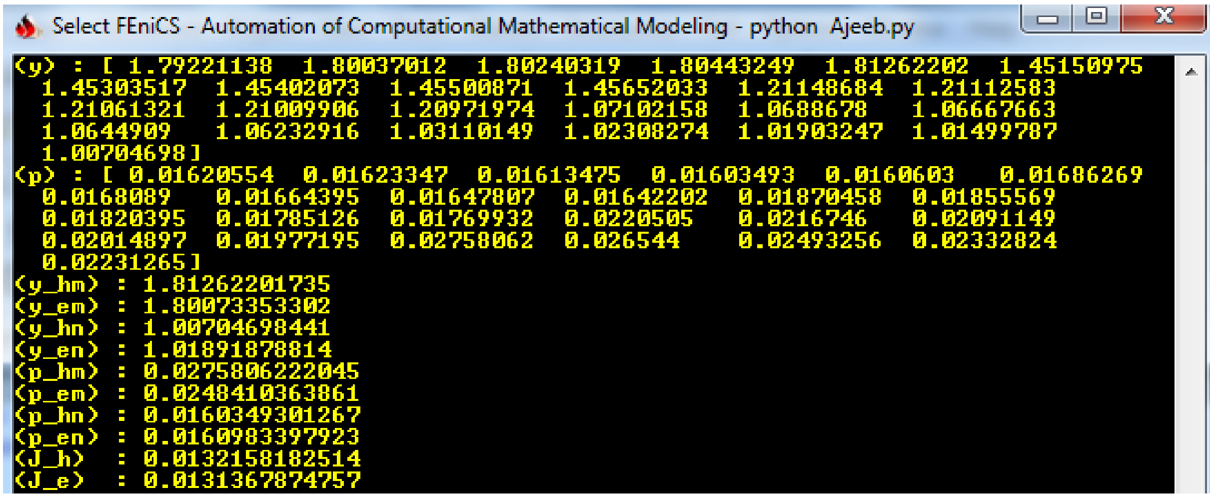

Master's Thesis for the degree of Engineering Mathematics and Computational Science

Finite Element Method for Optimal Control of Elliptic Partial Differential Equations

AJEEB UR REHMAN

arjunejo@hotmail.com & junejo@student.chalmers.se

Supervisor: Prof. Stig Larsson

Computational Mathematics Group
Department of Mathematical Science
Chalmers University of Technology
and University of Gothenburg

Gothenburg, Sweden, November 2012

And now that we may give final praise to the machine we may say that it will be desirable to all who are engaged in computations which, it is well known, are the managers of financial affairs, the administrators of others' estates, merchants, surveyors, geographers, navigators, astronomers ... For it is unworthy of excellent men to lose hours like slaves in the labor of calculations which could safely be relegated to anyone else if the machine were used.

Gottfried Wilhelm Leibniz(1646–1716)

Abstract

In this Master thesis numerical methods of optimal control of elliptic partial differential equations (EPDE) are studied. The problems considered in this work consist in minimizing a functional subject to EPDE constraints. The goal is to use FEniCS software to determine the state and control which minimize the corresponding cost functional, for this purpose we used numerical method based on an indirect approach, which means first derive the optimality system then solve numerically. In this work optimality system is derived by two ways, first way is based on Lagrange's method and other by reduced functional. Discretization of Optimality system of Optimal control for (EPDE) by finite element method is considered. Computation and comparison of exact and approximate solutions in two different ways, computation of error analysis and convergence rate by using FEniCS software are also considered.

Acknowledgements

First of all I am grateful to my supervisor Prof. Stig Larsson. Thank you for leading me to the world of Optimal Control Problems of PDEs and Python/FEniCS software. His well organised working process, excellent guidance and given valuable time for discussion will be great treasure for my life.
Master co-ordinator Johan Jonasson has been very helpful, thank you.
I would like to thanks all my teachers, staff and friends who assisted me whenever I asked them.
I would also like to thank and pay tribute to my family specially my parents for loving and for all support they have given me.
Last but not least I also wish to thank my sister Amina with core of my heart for always standing by me, for praying and for encouragement through these years.

Ajeeb ur Rehman, Gothenburg, 26$^{\text{th}}$ November, 2012

Contents

1

Introduction

Mathematical theory of optimal control has in the past few decades rapidly developed into an important and separate field of applied mathematics. One area of application of this theory lies in aviation and space technology: aspects of optimization come in to play whenever the motion of an air craft or a space vessel(which can be modelled by differential equation) has to follow a trajectory that is optimal in a sense to be specified.

Consider an aeroplane, trying to avoid an object that suddenly land on the road of airport. The pilot has some ability to maneuver the aeroplane by steering and breaking. Obviously the pilot faces the problem of finding an optimal way to maneuver the plane such that a collision is avoided (avoiding the obstacle and minimizing the final velocity). This is an example of *optimal control problem* which consists on a system of differential equations, describing the dynamics of plane and an objective functional that should be optimized (maximized or minimized).

The widespread applications of optimal control problems can be found in many fields of engineering such as mechanical, chemical, vehicle dynamics, aeronautics and in life sciences and also in many other disciplines.

Numerical solution of optimal control problems can be obtained by two (direct and indirect) approaches. In the direct method or approach, optimal control tries to minimize the cost function or objective function by working only on the set of *admissible functions* i.e., the pairs ($\mathbf{y}$,$\mathbf{u}$) satisfying the state equation, here $\mathbf{y}$ is the state variable and $\mathbf{u}$ is the control variable. The indirect method of optimization, in which the stationary points of the *Lagrangian functional*" are computed among which the possible local minima can be found. In this work we will discuss different model problems of optimal control of elliptic partial differential with different types of boundary conditions. We will take a brief look at Lagrangian functional and their associated stationary points, from which we will obtain the *Optimality system or Euler Lagrange system.* This optimality system will be solved by the finite element method, the interesting point is, from this optimality system we have write down the strong form, which we solved analytically for comparing the approximate and exact solutions, maximum and minimum of exact and approximate solutions and also for error analysis. We will also discuss about variational/weak form, variational calculus, reduced functional and discretization of Finite Element method. Furthermore we will discuss and compare the optimality systems, which we obtained by stationary points of Lagrangian functional and by the calculus of reduced functions that they are same optimality systems or in this section we conclude that optimality systems are same which we obtained by both techniques.

In the next section the obtained results of the successful implementation of both ways by FEniCS software of optimal control problem of elliptic partial differential equation whose cost functional is living on two different boundaries (*Euler-Lagrangian optimality system or optimality system derived by reduced functional*) [1, 2] are presented.

In conclusion section the exact results and approximate results of both techniques are compared and discussed and also error analysis calculated by four different ways and convergence rate is also discussed.

2

Preliminaries

In this chapter we will present a brief background and theory of optimal control problems (OCP), variational formulation (VF) and the corresponding Finite Element Method (FEM) of VF. We have mentioned here the corresponding FEM of VF, because some numerical/simulation softwares have the ability to generate automatic FEM solutions from VF. In this work we use the FEniCS software as a simulation tool for numerical results. Later the implemented results achieved by FEniCS software will be presented whose coding we have written in python language, the coding of variational formulation in python recognize FEniCS

So in this chapter we will also present VF of some model problems of partial differential equations(PDEs), later some of these PDEs will be used as a PDEs constraints or as a state equation in model problems of (OCP).

2.1 Optimal Control Problems of partial differential equations

Optimal control problems are concerned with finding the control functions that optimize cost functions for systems described by differential equations (ordinary differential equations or partial differential equations).

A variety of applications from different disciples and from different domains have witnessed the success stories of optimal control problems and also numerical methods have played main role for solving the application problems from different disciplines.
In this work we focused minimize cost functions subject to PDEs constrains only, later these optimal control problems will be solved and implemented by FEniCs software. In this work the cost functional will be denoted by J, the state function which governed by PDEs will be denoted by $\mathbf{y}$ and the control function will be denoted by $\mathbf{u}$. This notation for control is commonly used by the Russian word **u**pravlenie. The state function y is the quantity determined as the solution of PDEs, where as the control can be an

input function prescribed on boundary Γ called boundary control or an input function prescribed on volume domain Ω called distributed control. However, if both, a distributed control and a boundary control occur in a same problem, then $\mathbf{u}$ will denote the boundary control and $\mathbf{v}$ will denote the distributed control. In general J will be a quadratic function and depend on both state and control. However if there exists a unique state for each control (i.e., the map $S : u \longmapsto y = S(u)$), then J can be considered as a function of control u alone. There are two types of optimal control, each of them we will define below with the example of model problems and figures.

2.1.1 Distributed control/ Optimal heat source

Consider the model problem of optimal control with PDEs constraints, where the body which occupies the spatial domain $\Omega \subset R^3$ is heated or cooled. The optimal temperature distribution $y_\Omega : \Omega \to R^+$ is known and the heating elements can control the temperature u at each point of whole domain. Control is distributed over Ω and act as a heat source in the domain. Such type of problems occur, when the body is heated by electromagnetic induction or by microwaves, the entire optimal control problem cab be summarized in the following way:

$$\text{P.1}\begin{cases} \text{Minimize} \quad J(y,u) = \dfrac{1}{2}\displaystyle\int_\Omega (y(x) - y_\Omega(x))^2 dx + \dfrac{\lambda}{2}\int_\Omega u(x)^2 dx & (2.1a) \\ \text{subject to PDE constraints} \quad -\Delta y + y = u \quad \text{in} \quad \Omega & (2.1b) \\ \qquad n \cdot \nabla y = 0 \quad \text{on} \quad \Gamma & (2.1c) \\ \text{and control constraints} \quad u_a(x) \le u(x) \le u_b(x) \quad \text{on} \quad \Omega & (2.1d) \end{cases}$$

From model problem we see that the cost functional consists of state functional and control functional, where y is state and u is control functions which are defined on domain Ω and their corresponding integration can be performed on whole domain because the control is also living on Ω. In above problem $\lambda \ge 0$ is constant (will play same role in all model problems) which can be viewed as a measure of energy costs need to implement the control u. From mathematical point of view, λ is a regularization parameter, this term has the effect that possible optimal control show improved regularity properties.

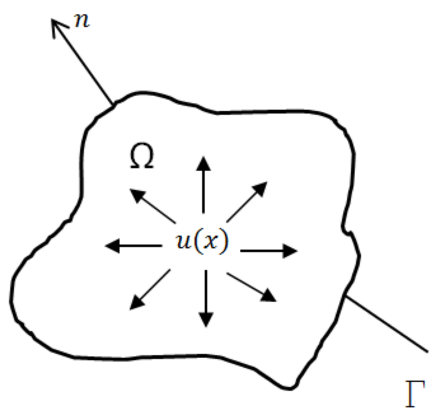

Figure 2.1: Distributed control

Due to physical and technical limitation of a heating or cooling device, one needs to impose some restrictions on the control because any device will be destroyed if its temperature becomes too low or too high, so (the point wise) control constraints are quite natural.

Consider an other more realistic model, one might not be able to control the temperature in whole domain Ω or might be has not resources to heat the whole domain, then the control can be imposed on a strict some part of domain i.e., Ω_c, where Ω_c is the subset of Ω. Then we obtain the following model

$$\text{P.2}\begin{cases} \text{Minimize} \quad J(y,u) = \dfrac{1}{2}\displaystyle\int_\Omega (y(x) - y_\Omega(x))^2 dx + \dfrac{\lambda}{2}\int_{\Omega_c} u(x)^2 dx & (2.2a) \\ \text{subject to PDE constraints} \quad -\Delta y + y = 0 \quad \text{in} \quad \Omega\backslash\Omega_c & (2.2b) \\ \qquad -\Delta y + y = u \quad \text{in} \quad \Omega_c & (2.2c) \\ \qquad n \cdot \nabla y = 0 \quad \text{on} \quad \Gamma & (2.2d) \\ \text{and control constraints} \quad u_a(x) \le u(x) \le u_b(x) \quad \text{on} \quad \Omega_c & (2.2e) \end{cases}$$

Now control u lives in Ω_c rather than Ω and here Ω_c is the heating element. Our goal is to find the control u (which acts in domain) in such a way that the corresponding temperature distribution$y = y(x)$ in Ω is the best possible approximation to a desired stationary temperature distribution $y_\Omega = y_\Omega(x)$ in Ω so that the cost functional in minimal. From model problems $P.1$ and $P.2$ we see that the cost functional is quadratic, the state equation is governed by a bilinear elliptic PDEs and control acts in the whole domain (Ω) or at the subset

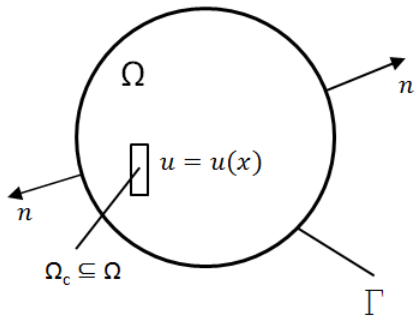

Figure 2.2: Distributed control living in sub domain

of whole domain (Ω_c), so such type of problems are called a *bilinear-quadratic elliptic distributed control problems.* Now let us consider a body $\Omega \subset R^3$ that is to be heated or cooled.

2.1.2 Boundary control/ Optimal boundary heating

In this section we will present the model problems, where the control is no more living in whole domain Ω but control only acts on boundary Γ. let us consider a body $\Omega \subset R^3$ that is to heated or cooled.

We apply control u on boundary Γ as a heat source, which depends on location x on the boundary that is $u = u(x)$ and is constant in time and now unlike in problem $P.1$ the whole integration of cost function has not be performed on Ω but integration of second part has to be performed on Γ rather than Ω. Similarly the control constraints or point wise control constraints now has to be imposed on Γ instead of Ω. This type of problems occur when the body is heated or cooled and the control temperature acts only on boundary Γ.

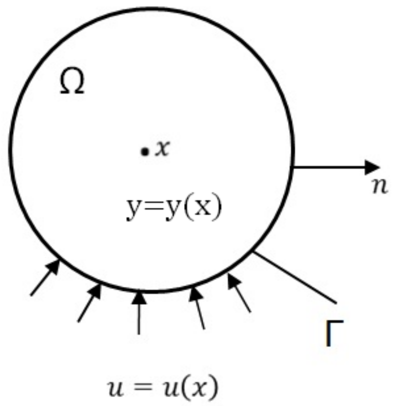

Figure 2.3: Control living on boundary

We can model such type of problems in the following way:

$$
\text{P.3}\begin{cases}
\text{Minimize} \quad J(y,u) = \dfrac{1}{2}\displaystyle\int_{\Omega} (y(x) - y_{\Omega}(x))^2 dx + \dfrac{\lambda}{2}\int_{\Gamma} u(x)^2 ds & \text{(2.3a)}\\
\text{subject to PDE constraints} \quad -\Delta y + y = f \quad \text{in} \quad \Omega & \text{(2.3b)}\\
\qquad n \cdot \nabla y = u \quad \text{on} \quad \Gamma & \text{(2.3c)}\\
\text{and control constraints} \quad u_a(x) \le u(x) \le u_b(x) \quad \text{on} \quad \Gamma & \text{(2.3d)}
\end{cases}
$$

In the similar way of P.2 more realistic control problem, from physical or technical point of view one might not be able to control the temperature on whole boundary Γ, but might be able to control the temperature on some part of boundary Γ_c, where $\Gamma_c \subset \Gamma$ [3] or might be one has not resources to control the temperature of whole boundary Γ. Such type of optimal control problems can be model in the following way:

$$
\text{P.4}\begin{cases}
\text{Minimize} \quad J(y,u) = \dfrac{1}{2}\displaystyle\int_{\Omega} (y(x) - y_{\Omega}(x))^2 dx + \dfrac{\lambda}{2}\int_{\Gamma_c} u(x)^2 ds & \text{(2.4a)}\\
\text{subject to PDE constraints} \quad -\Delta y + y = 0 \quad \text{in} \quad \Omega & \text{(2.4b)}\\
\qquad n \cdot \nabla y = u \quad \text{on} \quad \Gamma_c & \text{(2.4c)}\\
\qquad n \cdot \nabla y = 0 \quad \text{on} \quad \Gamma \backslash \Gamma_c & \text{(2.4d)}\\
\text{and control constraints} \quad u_a(x) \le u(x) \le u_b(x) \quad \text{on} \quad \Gamma_c & \text{(2.4e)}
\end{cases}
$$

The control temperature u acts only on some part of boundary Γ_c rather than whole boundary Γ. Now our goal from the models of boundary control problems is to find the control u which acts on boundary or on some part of boundary, in such a way that the actual y approximates y_Ω and the corresponding cost functional is minimized.

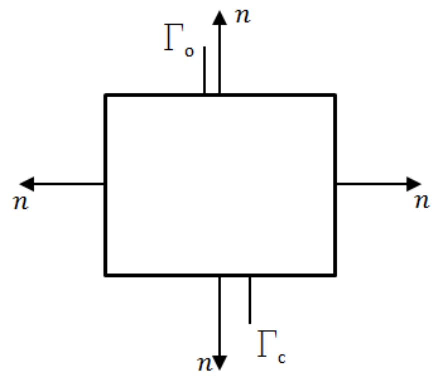

Figure 2.4: Control living on part of boundary

Models, P.3 and P.4 are called *linear-quadratic elliptic boundary control problems* because the state equations governed by elliptic PDEs are linear, cost functional is quadratic and the control acts on the boundary of domain or on some part of boundary of domain.

2.1.3 Optimal boundary control dependent on time

In above all model problems control u is independent of time, now we will present the problem when control is depend on time. Suppose the domain now $\Omega \subset R^3$ represent a potato, that is to be roasted over a fire with time $T > 0$. The temperature is denoted by $y = y(x,t)$, where $x \in \Omega$ and $t \in [0,T]$. Potato has, the initial temperature $y_0 = y_0(x) = y(x,0)$ and the final temperature is $y(x,T)$ and we want to serve it at a pleasant palatable temperature y_Ω at the final time T then the entire optimal control problem can be summarized in the following way:

$$
\text{P.5} \begin{cases}
\text{Minimize} \quad J(y,u) = \dfrac{1}{2}\displaystyle\int_\Omega (y(x,T) - y_\Omega(x))^2 dx + \dfrac{\lambda}{2}\int_0^T \int_\Gamma (u(x,t))^2 ds(x) dt & (2.5a) \\
\text{subject to PDE constraints} \quad y_t - \Delta y + y = 0 \quad \text{in } \Lambda & (2.5b) \\
\qquad n \cdot \nabla y = u \quad \text{on } \Upsilon & (2.5c) \\
\qquad y(x,0) = y_0(x) \quad \text{in } \Omega & (2.5d) \\
\text{and control constraints} \quad u_a(x,t) \leq u(x,t) \leq u_b(x,t) \quad \text{on } \Upsilon & (2.5e)
\end{cases}
$$

Where $\Lambda := \Omega \times (0,T)$ and $\Upsilon := \Gamma \times (0,T)$. Such type of problems are called *linear-quadratic parabolic boundary control problems*, because now the state equation is governed by linear parabolic PDE.

2.1.4 Optimal vibrations control dependent on time

Now Let the domain $\Omega \subset R^2$ represent a bridge and suppose that the group of pedestrians crosses a bridge, trying to excite oscillations in it, its transversal displacement denoted by $y = y(x,t)$, $u = u(x,t)$ is the force density acting in the vertical direction and $y_e = y_e(x,t)$ is a desired evolution of transversal vibrations. Then the optimal control model becomes:

$$\text{P.6}\begin{cases} \text{Min} \quad J(y,u) = \frac{1}{2}\int_0^T\int_\Omega (y(x,t) - y_\Omega(x))^2 dxdt + \frac{\lambda}{2}\int_0^T\int_\Gamma (u(x,t))^2 ds(x)dt & (2.6a) \\ \text{subject to PDE constraints} \quad y_{tt} - \Delta y = 0 \quad \text{in} \quad \Lambda & (2.6b) \\ y(0) = y_0 \quad \text{in} \quad \Omega & (2.6c) \\ y_t(0) = y_1 \quad \text{in} \quad \Omega & (2.6d) \\ y = u \quad \text{on} \quad \Upsilon & (2.6e) \\ \text{and control constraints} \quad u_a(x,t) \le u(x,t) \le u_b(x,t) \quad \text{on} \quad \Upsilon & (2.6f) \end{cases}$$

This is a *linear-quadratic hyperbolic boundary control problem*, because now the state equation is governed by linear hyperbolic PDE with non homogeneous Dirichlet boundary condition.

2.2 Variational formulation

Consider the elliptic PDE with Neumann boundary conditions

$$\begin{aligned} -\Delta y + y = f \qquad & \text{in} \quad \Omega \\ \partial_n y = u \quad \text{on} \quad \Gamma_c, \qquad & \partial_n y = 0 \quad \text{on} \quad \partial\Omega\backslash\Gamma_c \end{aligned} \tag{2.7}$$

where $f \in L^2(\Omega)$ is a given function and $u \in L^2(\Gamma)$, here $\partial_n y$ denotes the directional derivatives of y in the direction of outward unit normal n to Γ, generally assumed that the domain Ω is a bounded and Γ is the boundary of domain. For variational formulation, we multiply $\psi \in C^1$ with differential equation(2.7), not necessary to satisfy the boundary conditions, and then integrate over domain Ω, we obtain

$$-\int_\Omega \Delta y\psi dx + \int_\Omega y\psi dx = \int_\Omega f\psi dx$$

use Green's formula or integration by parts then yields

$$\int_\Omega \nabla y \cdot \nabla\psi dx - \int_\Gamma \partial_n y\psi ds + \int_\Omega y\psi dx = \int_\Omega f\psi dx$$

substitute the boundary conditions, i-e, $\partial_n y = u \quad \text{on} \quad \Gamma_c, \quad \partial_n y = 0 \quad \text{on} \quad \partial\Omega\backslash\Gamma_c$, then we obtain

$$\int_{\Omega} \nabla y \cdot \nabla \psi dx - \int_{\Gamma_c} u\psi ds + \int_{\Omega} y\psi dx = \int_{\Omega} f\psi dx, \quad \forall \psi \in C^1. \tag{2.8}$$

conversely if $y \in C^2$ satisfies (2.8), then apply Green's formula we have

$$\int_{\Omega} (-\Delta y + y - f)\psi dx + \int_{\Gamma_c} u\psi dx = 0, \quad \forall \psi \in C^1(\Omega)$$

Recall that $C^1(\Omega)$ is dense in $H^1(\Omega)$ and for fixed y all expressions in the equation depend continuously on $\psi \in H^1(\Omega)$, we conclude its validity $\forall \psi \in H^1(\Omega)$

Therefore, the variational formulation read as, Find $y \in H^1(\Omega)$ such that

$$\int_{\Omega} \nabla y \cdot \nabla \psi dx + \int_{\Omega} y\psi dx = \int_{\Omega} f\psi dx + \int_{\Gamma_c} u\psi ds, \quad \forall \psi \in H^1(\Omega). \tag{2.9}$$

above equation also called weak solution or weak form.[4] It is remarkable that variational formulation (weak) or first order derivatives are needed for a second order equation. More generally the abstract form of weak formulation can be treat as find $y \in H^1(\Omega)$, such that

$$\alpha(y,\psi) = L(\psi), \quad \forall \psi \in H^1(\Omega),$$

where in this problem

$$\alpha(y,\psi) = \int_{\Omega} \nabla y \cdot \nabla \psi dx + \int_{\Omega} y\psi dx \quad \text{and} \quad L(\psi) = \int_{\Omega} f\psi dx + \int_{\Gamma_c} u\psi ds \tag{2.10}$$

The bilinear form defined as $\alpha(\cdot,\cdot) : V \times V \to \mathbb{R}$, linear and continuous functional defined as $L(\cdot) : V \to \mathbb{R}$. Finally weak formulation in abstract form of general PDE is defined as below:

$$\alpha(y,\psi) = L(\psi) \quad \forall \psi \in V. \tag{2.11}$$

Note that equation(2.9) can also be read as, find $y \in H^1(\Omega)$ such that,

$$(\nabla y, \nabla \psi)_{\Omega} + (y,\psi)_{\Omega} = (f,\psi)_{\Omega} + (u,\psi)_{\Gamma_c} \;\; \forall \psi \in H^1(\Omega) \tag{2.12}$$

now we will talk about approximate solution, divide the polygonal domain Ω into finite triangles, make sure that the intersection of any two triangles is either, a node,a common edge or empty. Now the domain is denoted by $\bar{\Omega}$ means triangularized domain. Finite element or approximate solution of equation (2.11) is, find $y_h \in S_h$ such that,

$$(\nabla y_h, \nabla \phi)_{\Omega} + (y_h,\phi)_{\Omega} = (f,\phi)_{\Omega} + (u,\phi)_{\Gamma_c} \;\; \forall \phi \in S_h \tag{2.13}$$

Note that $S_h = \{\phi \in C(\bar{\Omega}) : \phi \text{ is linear in } T \text{ for each } T \in \pounds_h\}$ and also $S_h \subset H^1$, where $\pounds_h = \{T\}$ is a subset of closed triangles T. Let $\{N_i\}_{1=1}^{M_h}$ be the set of interior nodes, i.e., those that do not lie on boundary Γ. The function S_h then uniquely determined by its values at the K_j, and the set of pyramid functions $\{\Phi_i\}_{i=1}^{M_h} \subset S_h$, defined by

$$\Phi_i(K_j) = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}$$

forms a basis for S_h. If $\phi \in S_h$ we thus have $\phi(x) = \sum_{i=1}^{M_h} \phi_i \Phi_i(x)$, where $\phi_i = \phi(K_i)$ are the nodal values of ϕ. There fore S_h is a finite dimensional subspace of the Hilbert space H^1. Put $y_h(x) = \sum_{i=1}^{M_h} U_i \Phi_i(x)$ and $\phi = \Phi_j(x)$ in equation (2.13), we obtain

$$\sum_{i=1}^{M_h} (\nabla\Phi_i, \nabla\Phi_j) U_j + \sum_{i=1}^{M_h} (\Phi_i, \Phi_j) U_j = (f, \Phi_j) + (u, \Phi_j)$$

which produces the linear system of equations $AU + BU = b$ for the determination of U_j, where $U_i = U(x_i)$ are the nodal values of U.[2, 5] $A = \sum_{i=1}^{M_h} (\nabla\Phi_i, \nabla\Phi_j) U_j$ is stiffness matrix, $B = \sum_{i=1}^{M_h} (\Phi_i, \Phi_j) U_j$ is the mass matrix and the right hand side $b = (f, \Phi_j) + (u, \Phi_j)$ is the load vector. The matrices A and B are symmetric, positive definite and have a unique solution in S_h. Moreover the matrices, which we get from above equation are large and sparse, if mesh is fine i.e., large portion of its elements ar zero , because Φ_j vanishes except at the common node of triangles N_j, so that $a_{ij} = 0$ unless N_i and N_j are neighbour's. Note that (2.13) also called Galerkin approximation. Below figures show the triangularized domain.

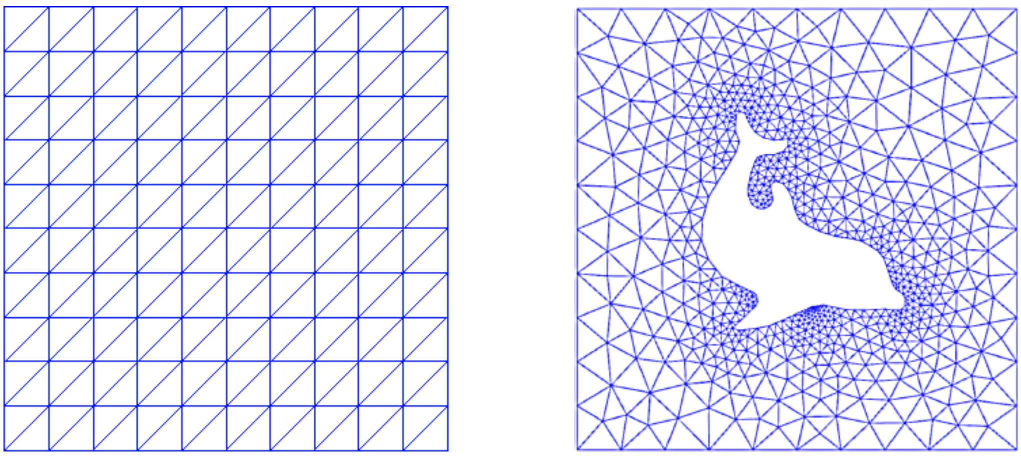

Figure 2.5: Figures show meshes of different domains

3

Methods for Optimal Control Problems

Present chapter will provide the information of two different approaches, indirect and direct approaches respectively, for finding the solution of optimal control problems. In both approaches different methods have been used for solution, but here we will discuss about the Lagrangian method which is type of indirect approach and conditioned gradient method, which is the type of direct approach respectively.

3.1 Lagrangian method

Consider the optimal boundary control problem

$$\begin{cases} \text{Minimize} \quad J(y,u) = \dfrac{1}{2}\displaystyle\int_{\Gamma_o} (y - y_0)^2 ds + \dfrac{\lambda}{2}\displaystyle\int_{\Gamma_c} u(x)^2 ds & (3.1a) \\ \text{subject to PDE constraints} \quad -\Delta y + y = f \quad \text{in} \ \Omega & (3.1b) \\ \qquad \partial_n y = u \quad \text{on} \ \Gamma_c & (3.1c) \\ \qquad \partial_n y = 0 \quad \text{on} \ \partial\Omega \backslash \Gamma_c & (3.1d) \\ \text{and control constraints} \quad u_a(x) \leq u(x) \leq u_b(x) \quad \text{on} \ \Gamma_c & (3.1e) \end{cases}$$

For finding the optimal solution of this problem, here four constraints have to be obeyed, three difficult ones of boundary value problem i.e., PDE constraints and a harmless one i-e box or pointwise constraint. But in Lagrangian method no need to obey these constraints, however one more variable have to be found. In this problem the generic solution spaces, for state variable y space denoted by $V := H^1(\Omega)$ and for control variable

u space denoted by $Q := L^2(\Gamma_c)$. We now apply Lagrangian functional by

$$\pounds(y,u,p) = J(y,u) - A(y)(p) - B(u,p) \tag{3.2}$$

for eliminating only the constraints equations by means of Lagrangian multiplier p. Here the Lagrangian multiplier (adjoint variable) p defined in Ω, so $p \in V$, the primal variable $y \in V$ and the control variable $u \in Q$, where as $A(\cdot)(\cdot)$ and $B(\cdot,\cdot)$ are the variational formulation of PDEs with a control form $B(\cdot,\cdot)$ that's why we have defined the variational formulation in last chapter. Now we seek stationary points $x := \{y,u,p\} \in X := V \times Q \times V$ of $\pounds(y,u,p)$ which are determined by differentiating the (3.2) with respect to y,u and p respectively and get the following system of equations

$$\begin{cases} \pounds'_y(y,u,p) = J'_y(y,u)(\phi) - A'(y)(\phi,p) = 0 \quad \forall \phi \in V, & (3.3a) \\ \pounds'_u(y,u,p) = J'_u(y,u)(\chi) - B(\chi,p) = 0 \quad \forall \chi \in Q, & (3.3b) \\ \pounds'_p(y,u,p) = -A(u)(\psi) - B(u,\psi) = 0 \quad \forall \psi \in V. & (3.3c) \end{cases}$$

This is called Optimality system or Euler-Lagrangian system:[6] Note that the last equation is just the state equation to be satisfied by any admissible pair $\{y,u\}$. The Galerkin approximation determines $x_h := \{y_h,u_h,p_h\} \in X_h := V_h \times Q_h \times V_h$ by a corresponding system of discrete equations

$$J'_y(y_h,u_h)(\phi_h) - A'(y_h)(\phi_h,p_h) = 0 \quad \forall \phi_h \in V_h, \tag{3.4a}$$

$$J'_u(y_h,u_h)(\chi_h) - B(\chi_h,p_h) = 0 \quad \forall \chi_h \in Q_h, \tag{3.4b}$$

$$-A(u_h)(\psi_h) - B(u_h,\psi_h) = 0 \quad \forall \psi_h \in V_h. \tag{3.4c}$$

We assume that both systems (3.3) and (3.4) have unique solution. But fortunately FEniCS software automatically generate approximate solution and no need to write down the coding of discrete equations i.e., (3.4) system, FEniCS just recognize the coding of, optimality system (3.3), which we will present in next chapter and that is our main goal. Note that it is also possible to eliminate the pointwise control constraints by incorporating them into the Lagrangian by means of additional multipliers μ_a and μ_b i.e., $\pounds(y,u,p,\mu_a,\mu_b)$. This can be done with the formation of *Karush-Kuhn-Tucker system* easily.

3.1.1 Lagrangian functional of boundary control problem

Construction of Lagrangian principle: Recall equation 2.12 which is also the variational formulation of, PDEs constraints i.e., (3.1b)-(3.1d) of problem (3.1), where $A(y)(\psi) = (\nabla y,\nabla\psi) + (y,\psi) - (f,\psi)$ and $B(u,\psi) = -(u,\psi)$, then we put these values in Lagrangian formula $\pounds(y,y,p) = J(y,u) - A(y)(p) - B(u,p)$, we obtain

$$\pounds(y,u,p) = J(y,u) - (\nabla y,\nabla p) - (y,p) + (f,p) + (u,p) \tag{3.5}$$

Now differentiate (3.5) with respect to (y,u,p) respectively, where $J(y,u)$ is also defined in equation $(3.1a)$ i.e., $J(y,u) = \frac{1}{2}\|y-y_0\|^2_{\Gamma_o} + \frac{1}{2}\lambda\|u\|^2_{\Gamma_c}$,

$$\mathcal{L}'_y(y,u,p)(\varphi) = (y-y_0,\varphi)_{\Gamma_o} - (\nabla p,\nabla\varphi)_\Omega - (y,p)_\Omega - (\varphi,p)_\Omega \quad \forall\varphi\in V \tag{3.6a}$$

$$\mathcal{L}'_u(y,u,p)(\chi) = \lambda(u,\chi)_{\Gamma_c} + (p,\chi)_{\Gamma_c} \quad \forall\chi\in Q \tag{3.6b}$$

$$\mathcal{L}'_p(y,u,p)(\psi) = -(\nabla y,\nabla\psi)_\Omega - (y,\psi)_\Omega + (f,\psi)_\Omega + (u,\psi)_{\Gamma_c} \quad \forall\psi\in V \tag{3.6c}$$

For stationary points putting derivative equals to zero in above system,we obtain

$$(y-y_0,\varphi)_{\Gamma_O} - (\nabla p,\nabla\varphi)_\Omega - (p,\varphi)_\Omega - (\varphi,p)_\Omega = 0 \quad \forall\varphi\in V \tag{3.7a}$$

$$\lambda(u,\chi)_{\Gamma_C} + (p,\chi)_{\Gamma_c} = 0 \quad \forall\chi\in Q \tag{3.7b}$$

$$-(\nabla y,\nabla\psi)_\Omega - (y,\psi)_\Omega + (f,\psi)_\Omega + (u,\psi)_{\Gamma_c} = 0 \quad \forall\psi\in V \tag{3.7c}$$

This is called Euler-Lagrangian system or optimality system (3.7)

3.1.2 Strong form of boundary control

Now we are interesting to state the strong form of above of (3.7) system, through strong form we will find the exact solution in next chapter, which will help us to compare the exact and approximate solutions, by exact and approximate solutions, we will also find the error analysis and convergence rate.

Now apply Green's formula on (3.7a) and on (3.7c) and we see that $\partial_n p = (y-y_0)$ on Γ_o and $\partial_n p = 0$ on rest boundary i.e., $\partial_n p = 0$ on $\partial\Omega\backslash\Gamma_o$ in (3.7a) and similarly $\partial_n p = u$ on Γ_c and rest boundary is zero in (3.7c), so when we apply Green's formula on all equations of the system (3.7), we obtain

$$-\Delta p + p = 0 \ \text{ in } \ \Omega, \quad \partial_n p = y - y_0 \ \text{ on } \ \Gamma_o, \quad \partial_n p = 0 \ \text{ on } \ \partial\Omega\backslash\Gamma_o \tag{3.8a}$$

$$p = -\lambda u \ \text{ on } \ \Gamma_c \tag{3.8b}$$

$$-\Delta y + y = f \ \text{ in } \ \Omega, \quad \partial y_n = u \ \text{ on } \ \Gamma_c, \quad \partial_n y = 0 \ \ on \ \ \partial\Omega\backslash\Gamma_c \tag{3.8c}$$

3.1.3 Lagrangian functional of optimal distributed control

Consider the following optimal distributed control problem:

$$\begin{cases} \text{Minimize} \quad J(y,u) = \dfrac{1}{2}\displaystyle\int_\Omega (y(x)-y_\Omega(x))^2dx + \dfrac{\lambda}{2}\int_\Omega u(x)^2dx & (3.9a) \\ \text{subject to PDE constraints} \quad -\Delta y + y = u \quad \text{in } \ \Omega & (3.9b) \\ \qquad\qquad\qquad\qquad\qquad\quad n\cdot\nabla y = 0 \quad \text{on } \ \Gamma & (3.9c) \end{cases}$$

First we find the variational form of PDE, multiply test function ϕ with (3.9b) and integrate over Ω, we obtain

$$-\int_\Omega \Delta y\phi dx + \int_\Omega y\phi dx = \int_\Omega u\phi dx \tag{3.10}$$

Applying Green's formula and using the value of boundary condition we get, find $y \in H_0^1(\Omega)$,

$$(\nabla y,\nabla\phi)_\Omega + (y,\phi)_\Omega - (u,\phi)_\Omega = 0 \quad \forall\phi \in H_0^1(\Omega), \tag{3.11}$$

now the solution spaces are $Q, V := H_0^1(\Omega)$. Here $A(y)(\phi) = (\nabla y,\nabla\phi)_\Omega + (y,\phi)_\Omega$ and $B(u,\phi) = -(u,\phi)_\Omega$, then the Lagrangian functional is,

$$\pounds(y,u,p) = J(y,u) - A(y)(p) - B(u,p) \tag{3.12}$$

Invoking the values of $J(y,u)$, $A(\cdot)(\cdot)$ and $B(\cdot,\cdot)$ in (3.12), we get:

$$\pounds(y,u,p) = \frac{1}{2}\|y - y_\Omega\|^2_{L^2(\Omega)} + \frac{\lambda}{2}\|u\|^2_{L^2(\Omega)} - (\nabla y,\nabla p)_\Omega - (y,p)_\Omega + (u,p)_\Omega \tag{3.13}$$

Now for optimality system first take the derivative of (3.13) with respect to (y,u,p) respectively and then putting derivative equal to zero, obtain

$$(y - y_\Omega, \varphi)_\Omega - (\nabla p,\nabla\varphi)_\Omega - (p,\varphi)_\Omega = 0 \quad \forall\varphi \in V \tag{3.14a}$$
$$\lambda(u,\chi)_\Omega + (p,\chi)_\Omega = 0 \quad \forall\chi \in Q \tag{3.14b}$$
$$-(\nabla y,\nabla\psi)_\Omega - (y,\psi)_\Omega + (u,\psi)_\Omega = 0 \quad \forall\psi \in V \tag{3.14c}$$

3.1.4 Strong form of distributed control

One can see that nothing is defined on boundary of system (3.14), so we conclude that $\partial_n p = 0$ and also $\partial_n y = 0$ on Γ respectively, for illustration and for exact solution we stated the strong form of (3.14) at following

$$\begin{cases} -\Delta p + p = y - y_\Omega \;\text{ in }\; \Omega, \quad \partial_n p = 0 \;\text{ on }\; \Gamma & (3.15a) \\ \quad p = -\lambda u \;\text{ in }\; \Omega & (3.15b) \\ -\Delta y + y = u \;\text{ in }\; \Omega, \;\; \partial_n y = 0 \;\text{ on }\; \Gamma & (3.15c) \end{cases}$$

3.2 Reduced functional

3.2.1 Reduced functional of optimal boundary control problem

Recall the model problem, where the control u is living on boundary Γ_c

$$\begin{cases} \text{Minimize } \; J(y,u) = \frac{1}{2}\|y - y_0\|^2_{L^2(\Gamma_o)} + \frac{\lambda}{2}\|u\|^2_{L^2(\Gamma_c)} & (3.16a) \\ \text{subject to PDE constraints} \quad -\Delta y + y = f \qquad \text{in } \; \Omega & (3.16b) \\ \qquad\qquad\qquad\qquad \partial_n y = u \qquad \text{on } \; \Gamma_c & (3.16c) \\ \qquad\qquad\qquad\qquad \partial_n y = 0 \qquad \text{on } \; \partial\Omega\backslash\Gamma_c & (3.16d) \\ \text{and control constraints} \;\; u_a(x) \le u(x) \le u_b(x) \quad \text{on } \; \Gamma_c & (3.16e) \end{cases}$$

We define the set of admissible controls by $Q_{\text{ad}} = \{u \in L^2(\Gamma_c) : u_a(x) \le u(x) \le u_b(x) \text{ for almost every } x \in \Omega\}$, where Q_{ad} is a nonempty, closed, and convex subset of $L^2(\Gamma_c)$. Our goal is to implement optimal control problem in FEniCS when $\lambda > 0$, then the control define as $Q^* = \{u \in L^2(\Gamma_c)\}$, it means when $u \in Q^*$ then pointwise constraints are excluded, i.e., except then equation (3.16e), so now and onward we will only consider the optimal control problems except than box constraints. We now rewrite the problem (3.16) without equation (3.16e)

$$\begin{cases} \text{Minimize} \quad J(y,u) = \frac{1}{2}\|y - y_0\|^2_{L^2(\Gamma_o)} + \frac{\lambda}{2}\|u\|^2_{L^2(\Gamma_c)} & (3.17a) \\ \text{subject to PDE constraints} \quad -\Delta y + y = f \quad \text{in} \quad \Omega & (3.17b) \\ \partial_n y = u \quad \text{on} \quad \Gamma_c & (3.17c) \\ \partial_n y = 0 \quad \text{on} \quad \partial\Omega\backslash\Gamma_c & (3.17d) \end{cases}$$

Now the set of control define by $Q^* = \{u \in L^2(\Gamma_c)\}$, to every $u \in Q^*$ there corresponds a unique weak solution $y \in H^1(\Omega)$ to the boundary value problem 3.17, called state associated with u and the state space is $Y := H^1(\Omega)$. If y is depend on u, then its denoted by $y = y(u)$.
Definition: The control $\bar{u} \in Q^*$ and the state $\bar{y}$ are optimal if

$$J(\bar{y}, \bar{u}) \leq J(y(u),u) \quad \forall u \in Q^*.$$

for the treatment of the existence question, we now rewrite the optimal control problem as an optimization problem in terms of u. The mapping $G : L^2(\Gamma_c) \to H^1(\Omega)$, $u \to y(u)$, defined by (3.17b)-(3.17d) have a unique solution $y \in H^1(\Omega)$, is called the control to state operator or instead of G, we consider the operator E_2G, where $E_2 : \quad H^1(\Omega) \to L^2(\Omega)$ denotes the embedding operator that assigns to each function $y \in Y = H^1(\Omega)$ the same function in $L^2(\Omega)$. In the problem of (3.17), we thus have

$$S : L^2(\Gamma_c) \to L^2(\Omega), \quad u \to y(u). \tag{3.18}$$

Definition: Let the real Hilbert spaces $X,(\cdot,\cdot)_X$ and $Y,(\cdot,\cdot)_Y$ as well as an operator $S \in (U,V)$ be given. An operator S^* is called the Hilbert space adjoint or adjoint of S if

$$(Sx,y)_Y = (x,S^*y)_X \quad \forall x \in X, \quad y \in Y. \tag{3.19}$$

Moreover put $y = Su$ in the cost functional i.e., (3.17a), which reduced to the following quadratic optimization problem in the Hilbert space $L^2(\Gamma)$:

$$\min_{u \in Q^*} f(u) = \frac{1}{2}\|Su - y_0\|^2_{L^2(\Gamma_o)} + \frac{\lambda}{2}\|u\|^2_{L^2(\Gamma_c)} \tag{3.20}$$

The use of S in equation (3.20) has the advantage that the adjoint operator S^* (as defined in (3.19) also acts in the space $L^2(\Omega)$. The equation (3.20) is called reduced cost functional.

3.2.2 Reduced functional of optimal distributed control problem

Now consider the model problem where the control is living in space Ω:

$$\begin{cases} \text{Minimize} \quad J(y,u) = \frac{1}{2}\|y - y_\Omega\|^2_{L^2(\Omega)} + \frac{\lambda}{2}\|u\|^2_{L^2(\Omega)} dx & (3.21a) \\ \text{subject to PDE constraints} \quad -\Delta y + y = u \quad \text{in} \quad \Omega & (3.21b) \\ \partial_n y = 0 \quad \text{on} \quad \Gamma & (3.21c) \end{cases}$$

Now we again define the set Q^* for (3.21) because now control is living in Ω, which is $Q^* = \{u \in L^2(\Omega)\}$, now for $u \in L^2(\Omega)$ the elliptic boundary value problem i.e., (3.21b)-(3.21c) has a unique weak solution $y = y(u) \in H^1(\Omega)$ and the operator $G : L^2(\Omega) \to H^1(\Omega)$, $u \to y(u)$, is continuous. We interpret G as a continuous linear operator mapping $L^2(\Omega)$ into $L^2(\Omega)$, that is we take $S = E_2 G$ and $S : L^2(\Omega) \to L^2(\Omega)$. Now the reduced cost functional of (3.21a) is define as below

$$\min_{u \in Q^*} f(u) = \frac{1}{2}\|Su - y_\Omega\|^2_{L^2(\Omega)} + \frac{\lambda}{2}\|u\|^2_{L^2(\Omega)}. \tag{3.22}$$

Variational equality of the reduced cost functional i.e., (3.20) and (3.22) will be denoted by

$$f'(\bar{u})(u - \bar{u}) = 0 \quad \forall u \in Q^*. \tag{3.23}$$

Here $\bar{u}$ is called optimal, note that box constrains are not included, if some one want to include box constrains then *variational inequality* should be use rather than *variational equality* and Q_{ad} will be used instead of Q^* i-e $f'(u)(u - \bar{u}) \geq 0 \quad \forall u \in Q_{ad}$.
Similarly the reduced cost functional of

$$\min J(y,u,v) = \frac{\lambda_\Omega}{2}\|y - y_\Omega\|^2_{L^2(\Omega)} + \frac{\lambda_\Gamma}{2}\|y - y_\Gamma\|^2_{L^2(\Gamma)} + \frac{\lambda_u}{2}\|u\|^2_{L^2(\Gamma_c)} + \frac{\lambda_v}{2}\|v\|^2_{L^2(\Omega)}$$

is following, where $y = S(u,v)$

$$\min J(y,u,v) = f(u,v) = \frac{\lambda_\Omega}{2}\|S(u,v) - y_\Omega\|^2_{L^2(\Omega)} + \frac{\lambda_\Gamma}{2}\|S(u,v) - y_\Gamma\|^2_{L^2(\Gamma)} + \frac{\lambda_u}{2}\|u\|^2_{L^2(\Gamma_c)} + \frac{\lambda_v}{2}\|v\|^2_{L^2(\Omega)}.$$

For further theory please see [1, 7, 8, 9, 10]

3.3 Comparison of optimality systems

In this section we will also treat the model problems of optimal boundary control problem and optimal distributed control problem, as we have already found the optimality systems of both types of problems by Lagrangian method in weak form and in strong form, now will find the optimality systems by gradient of reduced forms and then we will compare that the optimality systems. Before going to model problems first we are presenting the theorem and its proof, which will help us to determine the adjoint equation.

Theorem: Suppose that the functions ξ, $\kappa \in L^2(\Omega)$, σ, $\upsilon \in L^2(\Gamma)$, d_0, $\gamma_\Omega \in L^\infty(\Omega)$ and α, $\delta_\Gamma \in L^\infty(\Gamma)$ be given, where $\alpha \geq 0$ and $d_0 \geq 0$, almost everywhere, if the following two elliptic PDEs are

$$\begin{cases} -\Delta y + d_0 y = \gamma_\Omega \kappa \quad \text{in} \quad \Omega \\ \partial_n y + \alpha y = \delta_\Gamma \upsilon \quad \text{on} \quad \Gamma \end{cases} \qquad \begin{aligned} -\Delta p + d_0 p = \xi \quad \text{in} \quad \Omega \\ \partial_n p + \alpha p = \sigma \quad \text{on} \quad \Gamma \end{aligned} \tag{3.24}$$

then

$$\int_\Omega \gamma_\Omega p\kappa dx + \int_\Gamma \delta_\Gamma p\upsilon ds = \int_\Omega \xi y dx + \int_\Gamma \sigma y ds \tag{3.25}$$

Proof: This theorem can be proved by variational formulation, so multiply p i.e., $p \in H^1(\Omega)$ with left hand PDE of (3.24), by applying Green's formula and using its boundary condition, we obtain, find $y \in H^1(\Omega)$ such that

$$\int_\Omega \nabla y \cdot \nabla p dx + \int_\Omega d_0 y p dx + \int_\Gamma \alpha y p ds = \int_\Omega \gamma p\kappa dx + \int_\Gamma \delta p\upsilon ds \tag{3.26}$$

and similarly for right hand equation of (3.24), multiply y, after using Green's formula and boundary conditions obtain, we find that $y \in H^1(\Omega)$ such that,

$$\int_\Omega \nabla y \cdot \nabla p dx + \int_\Omega d_0 y p dx + \int_\Gamma \alpha y p ds = \int_\Omega \gamma \xi y dx + \int_\Gamma \sigma y ds \tag{3.27}$$

Now we see that the left hand sides of both (3.26) and (3.27) are same, so the right hand side of both equations should must be equal, i.e.,

$$\int_\Omega \xi y dx + \int_\Gamma \sigma y ds = \int_\Omega \gamma_\Omega p\kappa dx + \int_\Gamma \delta_\Gamma p\upsilon ds$$

and which is required result □

With equation above or (3.25) in hand it is now easy to treat the problems of finding optimality systems for Neumann boundary condition or for a Robin boundary conditions, for the sake of simplicity, we treat the same problems which we discussed in Lagrangian method.

3.3.1 Optimal stationary distributed temperature

Here the control u lives on Ω, so we define $Q^* = \{u \in L^2(\Omega)\}$. Recall the optimal distributed control problem:

$$\begin{cases} \text{Minimize} \quad J(y,u) = \frac{1}{2} \int\limits_\Omega (y(x) - y_\Omega(x))^2 + \frac{\lambda}{2} \int\limits_\Omega u(x)^2 dx & (3.28a) \\ \text{subject to PDE constraints} \quad -\Delta y + y = u \quad \text{in} \quad \Omega & (3.28b) \\ \qquad\qquad\qquad\qquad\qquad \partial_n y = 0 \quad \text{on} \quad \Gamma & (3.28c) \end{cases}$$

according to (3.22) we rewrite the reduced cost functional

$$J(y,u) = \min_{u \in V^*} f(u) = \frac{1}{2}\|Su - y_\Omega\|^2_{L^2(\Omega)} + \frac{\lambda}{2}\|u\|^2_{L^2(\Omega)}. \tag{3.29}$$

First we want to determine the variational equality i.e., (3.23). For every $u \in Q^*$ and $t \in [0,1]$, the convexity of Q^* yields $\bar{u} + t(u - \bar{u}) \in Q^*$). Since $\bar{u}$ is optimal and obviously f is not *Gâteaux differentiable* in $L^2(\Omega)$, it is a *Fréchet differentiable* in $L^2(\Omega)$, since $f(u + th)$ may be undefined for some $h \in L^2(\Omega)$ [11] even for small $t > 0$, however it is directionally differentiable in the direction $u-\bar{u}$, since $u+t(u-\bar{u}) \in Q^*$. The optimality of $\bar{u}$ gives

$$\frac{f(\bar{u} + t(u - \bar{u})) - f(\bar{u})}{t} = 0 \qquad \forall u \in Q^* \tag{3.30}$$

passing the limit $t \downarrow 0$ in equation (3.30) implies for directional derivative

$$f'(\bar{u})(u - \bar{u}) = 0 \qquad \forall u \in Q^* \tag{3.31}$$

with the definition of reduced cost functional or equation (3.22) at hand, (3.31) is equivalent to

$$(S^*(S\bar{u} - y_\Omega,u - \bar{u}))_{L^2(\Omega)} + (\lambda\bar{u}, u - \bar{u})_{L^2(\Omega)} = 0 \qquad \forall u \in Q^* \tag{3.32}$$

The *gradient* is

$$f'(\bar{u}) = (S^*(S\bar{u} - y_\Omega)) + \lambda\bar{u} \tag{3.33}$$

Note the difference between derivative and gradient, the directional derivative or derivative which is given by the rule is $f'(u)(u-\bar{u}) = (S^*(S\bar{u}-y_\Omega,u-\bar{u}))_{L^2(\Omega)}+(\lambda\bar{u}, u-\bar{u})_{L^2(\Omega)}$ and the gradient is equation (3.33). Since $S^*,S : L^2(\Omega) \to L^2(\Omega)$ so for avoiding the S^* in (3.32) we can write it as

$$(S\bar{u} - y_\Omega,Su - S\bar{u}))_{L^2(\Omega)} + (\lambda\bar{u}, u - \bar{u})_{L^2(\Omega)} = 0 \qquad \forall u \in Q^* \tag{3.34}$$

Now suppose that put $p = S^*(S\bar{u} - y_\Omega) = S^*(\bar{y} - y_\Omega)$ in (3.32), then (3.32) becomes

$$(p,u - \bar{u})_{L^2(\Omega)} + (\lambda\bar{u}, u - \bar{u})_{L^2(\Omega)} = 0 \qquad \forall u \in Q^* \tag{3.35}$$

where the adjoint p solves the following boundary value problem

$$-\Delta p + p = \bar{y} - y_\Omega \quad \text{in} \quad \Omega, \quad \partial_n p = 0 \qquad \text{on} \quad \Gamma \tag{3.36}$$

now recall the state equation

$$-\Delta y + y = u \quad \text{in} \quad \Omega, \quad \partial_n y = 0 \qquad \text{on} \quad \Gamma \tag{3.37}$$

now multiplying the $y \in H^1(\Omega)$ with equation (3.36) and similarly multiplying $p \in H^1(\Omega)$ with equation (3.37) ,using Green's formula and boundary conditions or applying the theorem on both equations i.e., (3.36), (3.37), we obtain

$$\int_\Omega pu dx = \int_\Omega (\bar{y} - y_\Omega) y dx, \tag{3.38}$$

The optimal state $S\bar{u} = \bar{y}$ is the weak solution to the state equation associated with $\bar{u}$, while $y = Su$. Hence by the linearity of the state, we have $S(u - \bar{u}) = y - \bar{y}$, now put $y = y - \bar{y}$ and $u = u - \bar{u}$ in (3.38), we get

$$\int_\Omega p(u-\bar{u})dx = \int_\Omega (\bar{y} - y_\Omega)(y - \bar{y})dx \qquad \forall u \in Q^*, \tag{3.39}$$

equation (3.39) is the same as equation(3.35), which prove the value of p too. Therefore equation (3.32) becomes

$$f'(\bar{u})(u-\bar{u}) = \int_\Omega (p + \lambda\bar{u})(u - \bar{u})dx \qquad \forall u \in Q^*, \tag{3.40}$$

Hence we get as a side result that the reduced gradient $f'(u)$ at an arbitrary u is of the form

$$f'(u) = p|\Omega + \lambda u \quad \text{or} \quad p + \lambda u = 0 \quad \text{in} \quad \Omega \tag{3.41}$$

where p solves the associated adjoint equation

$$-\Delta p + p = y - y_\Omega \quad \text{in} \quad \Omega, \quad \partial_n p = 0 \quad \text{on} \quad \Gamma \tag{3.42}$$

Now combine the (3.42), (3.41) and (3.37) respectively, we obtain

$$\begin{cases} -\Delta p + p = y - y_\Omega \quad \text{in} \quad \Omega, \quad \partial_n p = 0 \ \text{ on } \ \Gamma & (3.43\text{a}) \\ \quad p = -\lambda u \ \text{ or } \ u = -\dfrac{1}{\lambda}p \ \text{ in } \ \Omega & (3.43\text{b}) \\ -\Delta y + y = u \ \text{ in } \ \Omega, \quad \partial_n y = 0 \ \text{ on } \ \Gamma & (3.43\text{c}) \end{cases}$$

The system (3.43) is the same system which we got in strong form by Lagrangian method i.e., (3.15), its weak formulation will be

$$(y - y_\Omega, \varphi)_\Omega - (\nabla p, \nabla\varphi)_\Omega - (p,\varphi)_\Omega = 0 \quad \forall\varphi \in V \tag{3.44a}$$
$$\lambda(u,\chi)_\Omega + (p,\chi)_\Omega = 0 \quad \forall\chi \in Q \tag{3.44b}$$
$$-(\nabla y, \nabla\psi)_\Omega - (y,\psi)_\Omega + (u,\psi)_\Omega = 0 \quad \forall\psi \in V \tag{3.44c}$$

finally the system (3.44) is also same optimality system, to the system which we obtained by Lagrangian method i.e., (3.14). Hence we got the same optimality systems by reduced gradient and by Lagrangian method, and we are done. □

3.3.2 Optimal stationary boundary temperature

Now recall the boundary control problem (3.17) with $f = 0$,

$$\min \quad J(y,u) = \frac{1}{2}\|y - y_0\|^2_{L^2(\Gamma_o)} + \frac{\lambda}{2}\|u\|^2_{L^2(\Gamma_c)} \tag{3.45a}$$
$$\text{subject to PDE constraints} \quad -\Delta y + y = 0 \qquad \text{in} \quad \Omega \tag{3.45b}$$
$$\partial_n y = u \qquad \text{on} \quad \Gamma_c \tag{3.45c}$$
$$\partial_n y = 0 \qquad \text{on} \quad \partial\Omega\backslash\Gamma_c \tag{3.45d}$$

Here control u lives on Γ_c so we first define Q^*, which is $Q^* = \{u \in L^2(\Gamma_c)\}$. Now the control to state operator $G : u \mapsto y(u)$ is a continuous linear mapping from $L^2(\Gamma_c)$ into $H^1(\Omega)$, however we now consider G as an operator with range in $L^2(\Gamma_o)$, that is, $S = E_2 G : L^2(\Gamma_c) \to L^2(\Gamma_o)$ with embedding operator $E_2 : H^1(\Omega) \to L^2(\Gamma_c)$, so instead of using E_2G, we use only S. For the problem of stationary boundary control i-e (3.45), we thus have

$$S : L^2(\Gamma_\mathrm{c}) \to L^2(\Gamma_\mathrm{o}), u \mapsto y(u).$$

put $y = Su$,then the reduced cost functional attains the form

$$\min_{u\in Q^*} f(u) = J(y,u) = \frac{1}{2}\|Su - y_0\|^2_{L^2(\Gamma_\mathrm{o})} + \frac{\lambda}{2}\|u\|^2_{L(\Gamma_\mathrm{c})^2} \tag{3.46}$$

Let $\bar{u} \in Q^*$ denote optimal control and $\bar{y} = S\bar{u}$ denotes associated optimal state, then variational equality i.e., 3.31 of 3.46 is equivalent to

$$f'(\bar{u})(u - \bar{u}) = (S^*(S\bar{u} - y_0),u - \bar{u})_{L^2(\Gamma_\mathrm{o})} + \lambda(\bar{u}, u - \bar{u})_{L^2(\Gamma_c)} = 0 \qquad \forall u \in Q^* \tag{3.47}$$

here S^* is the adjoint operator $S^* : L^2(\Gamma_\mathrm{o}) \to L^2(\Gamma_\mathrm{c}), u \mapsto y(u)$, then equation (3.47) becomes

$$(S\bar{u} - y_0,Su - S\bar{u})_{L^2(\Gamma_\mathrm{o})} + \lambda(\bar{u}, u - \bar{u})_{L^2(\Gamma_\mathrm{c})} = 0 \qquad \forall u \in Q^* \tag{3.48}$$

or

$$(\bar{y} - y_0,y - \bar{y})_{L^2(\Gamma_\mathrm{o})} + \lambda(\bar{u}, u - \bar{u})_{L^2(\Gamma_\mathrm{c})} = 0 \qquad \forall u \in Q^* \tag{3.49}$$

Now suppose that

$$(\bar{y} - y_0,y - \bar{y})_{L^2(\Gamma_\mathrm{o})} = (p, u - \bar{u})_{L^2(\Gamma_\mathrm{c})} \tag{3.50}$$

For finding S^* and for proving the(3.50) and with the above considerations we are motivated to define p as the solution of

$$\begin{aligned} -\Delta p + p &= 0 \quad \text{in} \quad \Omega \\ \partial_n p &= \bar{y} - y_0 \quad \text{on} \quad \Gamma_\mathrm{o} \\ \partial_n p &= 0 \quad \text{on} \quad \partial\Omega\backslash\Gamma_\mathrm{o} \end{aligned} \tag{3.51}$$

this is called adjoint equation, its right hand side belongs to $L^2(\Gamma_o)$, since $y_0 \in L^2(\Gamma_o)$ by assumption and $\bar{y} \in Y = H^1(\Omega) \hookrightarrow L^2(\Gamma_{\Gamma_\mathrm{o}})$ so the (3.51) admits a unique weak solution $p \in H^1(\Omega)$ that satisfies

$$(\nabla p, \nabla y)_\Omega + (p,y)_\Omega = (\bar{y} - y_0,y)_{\Gamma_\mathrm{o}} \qquad \forall y \in H^1(\Omega). \tag{3.52}$$

Similarly multiply $p \in H^1(\Omega)$ with the (3.45b), then the (3.45b) -(3.45d) admits a unique weak solution $y \in H^1(\Omega)$ that satisfies

$$(\nabla y, \nabla p)_\Omega + (y,p)_\Omega = (p,u)_{\Gamma_\mathrm{c}} \qquad \forall p \in H^1(\Omega) \tag{3.53}$$

now we see that the left hand sides above of both equations i.e., (3.52)-(3.53) are same, then obviously right hand side of both equations should be equal too, or applying the theorem we obtain the following result

$$\int_{\Gamma_c} puds = \int_{\Gamma_o} (\bar{y} - y_0) y ds \tag{3.54}$$

then by putting $u = u - \bar{u}$ and $y = y - \bar{y}$ in above equation, it gives

$$\int_{\Gamma_c} p(u - \bar{u}) ds = \int_{\Gamma_o} (\bar{y} - y_0)(y - \bar{y}) ds. \tag{3.55}$$

Hence (3.50) is proved, now inserting the value of (3.55)in (3.49), then the updated variational equality will be

$$f'(\bar{u})(u - \bar{u}) = \int_{\Gamma_c} (p, u - \bar{u}) ds + \int_{\Gamma_c} \lambda(\bar{u}, u - \bar{u}) ds = 0 \quad \forall u \in Q^* \tag{3.56}$$

then the reduced gradient $f'(u)$ at any arbitrary u is of the form

$$f'(u) = p|\Gamma_c + \lambda u \qquad \text{or} \quad p + \lambda u = 0, \quad \text{on} \quad \Gamma_c \tag{3.57}$$

where p is the solution of following adjoint equation

$$\begin{aligned} -\Delta p + p = 0 \quad & \text{in} \quad \Omega \\ \partial_n p = y(u) - y_0 \quad & \text{on} \quad \Gamma_o \\ \partial_n p = 0 \quad & \text{on} \quad \partial\Omega \backslash \Gamma_o \end{aligned} \tag{3.58}$$

collect the adjoint, reduced gradient and state equations i.e., (3.58), (3.57), (3.45b-3.45d) respectively, which gives

$$\begin{cases} -\Delta p + p = 0 \ \text{in} \ \Omega, \ \ \partial_n p = y - y_0 \ \text{on} \ \Gamma_o, \ \ \partial_n p = 0 \ \text{on} \ \partial\Omega\backslash\Gamma_o & (3.59a) \\ u = -\dfrac{1}{\lambda} p \ \ \text{on} \ \Gamma_c & (3.59b) \\ -\Delta y + y = 0 \ \text{in} \ \Omega, \ \ \partial_n y = u \ \text{on} \ \Gamma_c, \ \ \partial_n y = 0 \ \text{on} \ \partial\Omega\backslash\Gamma_c & (3.59c) \end{cases}$$

the corresponding weak forms of (3.59) are following, where the generic solution spaces are $V := H^1(\Omega)$ for the adjoint, state variables p,y respectively and $Q := L^2(\Gamma_{\Gamma_c})$ for the control variable u

$$\begin{aligned} &(y - y_0, \upsilon)_{\Gamma_o} - (\nabla p, \nabla \upsilon)_\Omega - (p, \upsilon)_\Omega = 0 \quad \forall \varphi \in V && (3.60a) \\ &\lambda(u, \chi)_{\Gamma_c} + (p, \chi)_{\Gamma_c} = 0 \quad \forall \chi \in Q && (3.60b) \\ &-(\nabla y, \nabla \omega)_\Omega - (y, \omega)_\Omega + (u, \omega)_{\Gamma_c} = 0 \quad \forall \psi \in V && (3.60c) \end{aligned}$$

Hence we see that the optimality system (3.60) and the system of strong form (3.59) of boundary control problem (3.45) get by reduced gradient are respectively equal to the optimality system (3.7) and strong form's system (3.8) of by Lagrangian method □

4

Results

As mentioned earlier, in result section the optimality system derived by two different algorithms will be implemented by two ways (optimality system of two variables and optimality system of three equations). Before presenting the implementation results of these methods, first we will solve the system of strong form ,i.e., (3.8) which will give us the exact solution, this exact solution will help us for comparing the approximate and exact solutions and for finding the error analysis and convergence rate.

4.1 Exact solution

Recall the strong form of optimal boundary control ,i.e, (3.8)

$$-\Delta p + p = 0 \quad \text{in} \quad \Omega, \quad \partial_n p = y - y_0 \ \text{ on } \ \Gamma_\mathrm{o}, \ \ \partial_n p = 0 \ \text{ on } \ \partial\Omega\backslash\Gamma_\mathrm{o} \tag{4.1a}$$

$$p = -\lambda u \ \text{ on } \ \Gamma_\mathrm{c} \tag{4.1b}$$

$$-\Delta y + y = f \ \text{ in } \ \Omega, \ \ \partial y_n = u \ \text{ on } \ \Gamma_\mathrm{c}, \ \ \partial_n y = 0 \ \ on \ \ \partial\Omega\backslash\Gamma_\mathrm{c} \tag{4.1c}$$

From this system we have to find the values of (p,u,y) which satisfy all equations (PDEs) and boundary conditions. Before solving first we have to define, space Ω, boundary $\partial\Omega$, observation boundary Γ_o, control boundary Γ_c and some parts of boundary $\partial\Omega\backslash\Gamma_\mathrm{o}$ or $\partial\Omega\backslash\Gamma_\mathrm{c}$

$$\begin{aligned}
&\Omega := \{(x,y) \in (0,1) \times (0,1) \\
&\partial\Omega := \{(x,y) : (0,0), (1,0), (0,1), (1,1)\} \\
&\Gamma_\mathrm{c} := \{(x,y) : x \in [0,1] \quad \text{and} \quad y = 0\} \\
&\Gamma_\mathrm{o} := \{(x,y) : x \in [0,1] \quad \text{and} \quad y = 1\}
\end{aligned}$$

Now for simplicity we put $y - y_0 = A$ in (4.1a) then it becomes

$$-\Delta p + p = 0 \ \text{ in } \ \Omega, \quad \partial_n p = A \ \text{ on } \ \Gamma_\mathrm{o}, \ \ \partial_n p = 0 \ \text{ on } \ \partial\Omega\backslash\Gamma_\mathrm{o} \tag{4.3}$$

Keeping boundary conditions ($\partial_n p|_{\Gamma_o} = A$ and $\partial_n p|_{\partial\Omega\backslash\Gamma_o} = 0$) in mind and solving (4.3), we get the value of p

$$p = A\frac{\cosh(y)}{\sinh(1)} \tag{4.4}$$

Now put the value of (4.4) in the equation (4.1b) and plugging $y = 0$ at Γ_c we get

$$-\lambda u = \frac{A\cosh(0)}{\sinh(1)}$$

which is equivalent to

$$u = -\frac{1}{\lambda}\Big(\frac{A}{\sinh(1)}\Big) \tag{4.5}$$

Similarly keeping in mind the boundary conditions ($\partial_n y|_{\Gamma_c} = u$ and $\partial_n p|_{\partial\Omega\backslash\Gamma_c} = 0$) and the value of (4.5), after solving equation (4.1c), we obtain

$$y = \Big(\frac{1}{\lambda}\frac{A}{\sinh(1)} - 1\Big)\frac{\cosh(y-1)}{\sinh(1)} + \frac{y^2}{2} - y + 1 \tag{4.6}$$

Note that we have also assumed before that $A = y - y_0$ in (4.3), so now find the value of A at Γ_o

$$y(x,1) - y_0 = A \tag{4.7}$$

equation (4.7) gives the value of A with the presence of y_0

$$A = \frac{\lambda\sinh^2(1)}{1-\lambda\sinh^2(1)}\Big(y_0 + \frac{1}{\sinh(1)} - \frac{1}{2}\Big) \tag{4.8}$$

now its time to replace the value of A in equations (4.4), (4.5) and (4.6) which gives the corresponding values of (p,u,y) respectively:

$$\begin{cases} p = \dfrac{\lambda\sinh(1)\cosh(y)}{1-\lambda\sinh^2(1)}\Big(y_0 + \dfrac{1}{\sinh(1)} - \dfrac{1}{2}\Big) & \text{(4.9a)} \\ u = \dfrac{-\sinh(1)}{1-\lambda\sinh^2(1)}\Big(y_0 + \dfrac{1}{\sinh(1)} - \dfrac{1}{2}\Big) & \text{(4.9b)} \\ y = \dfrac{\cosh(y-1)}{1-\lambda\sinh^2(1)}\Big(y_0 + \dfrac{1}{\sinh(1)} - \dfrac{1}{2}\Big) - \dfrac{\cosh(y-1)}{\sinh(1)} + \dfrac{y^2}{2} - y + 1 & \text{(4.9c)} \end{cases}$$

Hence the system of (4.9) satisfies all equations (PDEs) and boundary conditions of (4.1) and we are done.

4.2 Discretization

We rewrite the optimality system (3.7)

$$(y - y_0, \varphi)_{\Gamma_o} - (\nabla p, \nabla\varphi)_\Omega - (p,\varphi)_\Omega = 0 \quad \forall\varphi \in V \tag{4.10a}$$
$$\lambda(u,\chi)_{\Gamma_c} + (p,\chi)_{\Gamma_c} = 0 \quad \forall\chi \in Q \tag{4.10b}$$
$$-(\nabla y, \nabla\psi)_\Omega - (y,\psi)_\Omega + (f,\psi)_\Omega + (u,\psi)_{\Gamma_c} = 0 \quad \forall\psi \in V \tag{4.10c}$$

for the Galerkin approximation of above optimality system, we consider a standard finite dimensional (bilinear finite elements) subspace V_h of V on triangulation meshes $T_h(\Omega)$, for the state and adjoint variables p_h and y_h respectively. For the approximation of the control variable u_h we use the space of traces of normal derivatives of functions in V_h on Γ_c ,i.e., piecewise linear shape functions. According to (3.4) the approximate optimality system of (4.10) reads:

$$-(\nabla y_h, \nabla\psi_h)_\Omega - (y_h,\psi_h)_\Omega + (f,\psi_h)_\Omega + (u_h,\psi_h)_{\Gamma_c} = 0 \quad \forall\psi_h \in V_h \tag{4.11a}$$
$$\lambda(u_h,\chi_h)_{\Gamma_c} + (p_h,\chi_h)_{\Gamma_c} = 0 \quad \forall\chi_h \in Q_h \tag{4.11b}$$
$$-(\nabla p_h, \nabla\varphi_h)_\Omega - (p_h,\varphi_h)_\Omega + (y_h - y_0, \varphi_h)_{\Gamma_o} = 0 \quad \forall\varphi_h \in V_h \tag{4.11c}$$

Let N denote the dimension of $V_h(T_h)$, and $\{\xi_n(x)\}$, $n = 1,...,N$ a basis of V_h then all test functions $\xi_n(x) \in V_h(T_h), n = \{1,...,N\}$. Then the test function defined as

$$\xi_n(x_m) = \begin{cases} 1, & \text{if } n = m \\ 0, & \text{if } n \neq m \end{cases} \tag{4.12}$$

forms a basis for V_h for the discretization of (4.11), we discretize y,u and p by the same basis of $\{\xi_n\}_{n=1}^N$ we write

$$y_h(x) = \sum_{n=1}^{N} y_n\xi_n(x), \quad u_h(x) = \sum_{n=1}^{N} u_n\xi_n(x) \text{ and } p_h(x) = \sum_{n=1}^{N} p_n\xi_n(x) \tag{4.13}$$

Moreover, we define the matrices

$$\left.\begin{aligned}
K_{mn} &= \int\limits_{\Omega} \nabla\xi_m(x)\nabla\xi_n(x)dx && \text{(4.14a)}\\
M_{mn} &= \int\limits_{\Omega} \xi_m(x)\xi_n(x)dx && \text{(4.14b)}\\
N_{mn} &= \int\limits_{\Gamma_c} \xi_m(x)\xi_n(x)ds && \text{(4.14c)}\\
L_{mn} &= \int\limits_{\Gamma_o} \xi_m(x)\xi_n(x)ds && \text{(4.14d)}\\
b_n &= \int\limits_{\Gamma_o} u_0\xi_n(x)dx && \text{(4.14e)}\\
b'_n &= \int\limits_{\Omega} f_n\xi_n(x) && \text{(4.14f)}
\end{aligned}\right\}$$

where K is known as the stiffness matrix, M,N,L as mass matrices and b, b' are load vectors. Keeping (4.12) in mind, then inserting (4.13) in (4.11a-4.11c), yields together with (4.14)

$$\begin{aligned}
(K+M)y - Nu+ &= b'_n f\\
\lambda Nu + Np &= 0\\
-Ly + (K+M)p &= -by_0
\end{aligned} \tag{4.15}$$

write (4.15) into matrix form, we obtain

$$\begin{pmatrix} K+M & -N & 0\\ 0 & \lambda N & N\\ -L & 0 & K+M \end{pmatrix} \begin{pmatrix} y\\ u\\ p \end{pmatrix} = \begin{pmatrix} b'f\\ 0\\ -by_0 \end{pmatrix}$$

coefficient matrix is symmetric, for simplicity interchange first and third row with each other, and multiplying minus with second row:

$$\begin{pmatrix} -L & 0 & K+M\\ 0 & -\lambda N & -N\\ K+M & -N & 0 \end{pmatrix} \begin{pmatrix} p\\ u\\ y \end{pmatrix} = \begin{pmatrix} -by_0\\ 0\\ b'f \end{pmatrix}. \tag{4.16}$$

for making blocks assume

$$A = \begin{pmatrix} -L & 0\\ 0 & -\lambda N \end{pmatrix} \quad B = \begin{pmatrix} K+M\\ -N \end{pmatrix} \quad B^T = \begin{pmatrix} K+M & -N \end{pmatrix}$$

inserting A, B and B^T in (4.16), we obtain the coefficient matrix

$$\bar{A} = \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix}$$

now we can easily find eigen values of above 2×2 matrix by using the characteristic equation $|\bar{A} - \mu I| = 0$, which gives

$$\begin{vmatrix} A - \mu & B \\ B^T & 0 - \mu \end{vmatrix} = 0$$

above equation produces, $\frac{A \pm \sqrt{A^2 + 4BB^T}}{2}$ and then $A < \sqrt{A^2 + 4BB^T}$, so clearly one eigenvalue is positive and other is negative.

Other way of discretization: According to (4.1b) or (4.10b) that, $(u = -\frac{1}{\lambda} p$ at $\lambda_c)$, then put the value of u in (4.10c) we obtain

$$-(\nabla y, \nabla \psi)_\Omega - (y, \psi)_\Omega + (f, \psi)_\Omega - \frac{1}{\lambda}(p, \psi)_{\Gamma_c} = 0 \quad \forall \psi \in V \tag{4.17a}$$

$$(y - y_0, \varphi)_{\Gamma_o} - (\nabla p, \nabla \varphi)_\Omega - (p, \varphi)_\Omega = 0 \quad \forall \varphi \in V \tag{4.17b}$$

now inserting (4.13) in (4.17) and by using (4.14), then the above optimality system produces

$$\begin{aligned} &\frac{1}{\lambda} Np + (K + M)y = b' f \\ &(K + M)p - Ly = -by_0 \end{aligned} \tag{4.18}$$

the coefficient matrix is symmetric, again using characteristic equation for checking the sign of eigenvalues

$$\begin{vmatrix} N - \mu & K + M \\ K + M & -L - \mu \end{vmatrix} = 0$$

as we already mentioned that λ is regularized parameter and we always keep $\lambda > 0$ in above

$$(N - \mu)(-L - \mu) - (K + M)^2 = 0$$

$$\frac{-(L - N) \pm \sqrt{(L - N)^2 + 4(NL + (K + M)^2)}}{2}$$

from above we see that one eigenvalue is positive and the other one is negative, hence in both ways the *coefficient matrix* is symmetric but not positive definite.

4.3 Numerical results

Now we are presenting the numerical results, however we have to write down the coding of the problems in Python language and run through "FEniCS software" [12, 13] which is the specific tool for automated solutions of differential equations by Finite Element Method (FEM). We have already evaluated the the exact solution of (4.1), now we will evaluate numerical solution and analysis of (4.1) in two ways.

4.3.1 Optimality system of two equations

Recall the optimality system of two variables where we have substituted $u = -\frac{1}{\lambda}p$ at Γ_c ,i.e., (4.17)

$$-(\nabla y,\nabla\psi)_\Omega - (y,\psi)_\Omega + (f,\psi)_\Omega - \frac{1}{\lambda}(p,\psi)_{\Gamma_c} = 0 \quad \forall\psi \in V \tag{4.19a}$$

$$(y - y_0,\varphi)_{\Gamma_o} - (\nabla p,\nabla\varphi)_\Omega - (p,\varphi)_\Omega = 0 \quad \forall\varphi \in V \tag{4.19b}$$

adding (4.19a) and (4.19b) we get

$$(\nabla y,\nabla\psi)_\Omega+(y,\psi)_\Omega+\frac{1}{\lambda}(p,\psi)_{\Gamma_c}+(\nabla p,\nabla\varphi)_\Omega-(y,\varphi)_{\Gamma_o}+(p,\varphi)_\Omega = (f,\psi)_\Omega-(y_0,\varphi)_{\Gamma_o} \quad \forall\varphi,\psi \in V \tag{4.20}$$

Now by using (4.20), regularizing the different values of $\lambda > 0$ and inserting the data as $y_0 = 1$, $f = y^2/2 - y$ and exact solutions in the coding window of Python. For error analysis we also interpolate the expressions of exact solution on fifth degree then compute the error analysis, for convergence rate we first perform the experiments with $h_1 > h_2 > h_3 \ldots$ and compute the errors $E1$, $E2$, $E3$ and so forth. Assuming two consecutive experiments, $E_i = Ch_i^r$ and $E_{i-1} = Ch_{i-1}^r$ for unknown constants C and r, then solve for r :

$$r = \frac{ln(E_i/E_{i-1})}{ln(h_i/h_{i-1})}$$

Analysis of y variable:
First we have evaluated and analysed y variable at different element sizes and at λ values. Tables 4.1, 4.2, 4.3, 4.4 show the error analysis and convergence rate of $||y_h - y||_{L^2}$ when $\lambda = 1$.

Tables 4.1, 4.2, 4.3 and 4.4 show that we approach the expected second-order convergence of linear Lagrange elements as the meshes become sufficiently fine and the error analysis in tables 4.1, 4.2, 4.3 and 4.4 is decreasing by the factor 1/4 as the size of meshes are decreasing 1/2.

Table 4.5 shows the maximum and minimum of exact and approximate values at different element sizes.

Table 4.6 shows the maximum and minimum of exact and approximate values at different regularize parameter. It also shows that if the value of regularization parameter λ converges to zero then the values of y also converges to zero.

Tables 4.7, 4.8, 4.9, 4.10 show the error analysis and convergence rate of $||y_h - y||_{l^2}$ when

Table 4.1: Error norm based on $y_h - y$ at $\lambda = 1$

Element size	Error	CR
h=1.25E-01	E=4.95E-02	r=1.94
h=6.25E-02	E=1.25E-02	r=1.98
h=3.13E-02	E=3.14E-03	r=2.00
h=1.56E-02	E=7.86E-04	r=2.00
h=7.81E-03	E=1.96E-04	r=2.00
h=3.91E-03	E=4.91E-05	r=2.00

Table 4.2: Error norm based on interpolation of y onto the same space as y_h at $\lambda = 1$

Element size	Error	CR
h=1.25E-01	E=4.95E-02	r=1.94
h=6.25E-02	E=1.25E-02	r=1.98
h=3.13E-02	E=3.14E-03	r=2.00
h=1.56E-02	E=7.86E-04	r=2.00
h=7.81E-03	E=1.96E-04	r=2.00
h=3.91E-03	E=4.91E-05	r=2.00

Table 4.3: Error norm based on interpolation of y to higher-order elements at $\lambda = 1$

Element size	Error	CR
h=1.25E-01	E=4.41E-02	r=1.94
h=6.25E-02	E=1.12E-02	r=1.98
h=3.13E-02	E=2.80E-03	r=1.99
h=1.56E-02	E=7.01E-04	r=2.00
h=7.81E-03	E=1.75E-04	r=2.00
h=3.91E-03	E=4.38E-05	r=2.00

Table 4.4: Error norm based on infinity norm (of nodal values) at $\lambda = 1$

Element size	Error	CR
h=1.25E-01	E=7.67E-02	r=1.87
h=6.25E-02	E=2.03E-02	r=1.92
h=3.13E-02	E=5.30E-03	r=1.94
h=1.56E-02	E=1.38E-03	r=1.94
h=7.81E-03	E=3.58E-04	r=1.95
h=3.91E-03	E=9.28E-05	r=1.95

Table 4.5: Maximum and minimum values of y_h and y at $\lambda = 1$

Element size	max(y_h)	min(y_h)	max(y)	min(y)
h=0.25E-00	-3.70711509408	-5.5864651043	-3.89572464983	-5.78295758313
h=1.25E-01	-3.84456209163	-5.73501213353	-3.89572464983	-5.78295758313
h=6.25E-02	-3.88228889337	-5.77165821923	-3.89572464983	-5.78295758313
h=3.13E-02	-3.89223236316	-5.78033270515	-3.89572464983	-5.78295758313
h=1.56E-02	-3.89482028553	-5.78235307828	-3.89572464983	-5.78295758313
h=7.81E-03	-3.89549088092	-5.78281946525	-3.89572464983	-5.78295758313
h=3.91E-03	-3.89566429792	-5.78292630535	-3.89572464983	-5.78295758313

$\lambda = 0.1$.

Tables 4.7, 4.8, 4.9 and 4.10 show that we approach the expected second-order convergence of linear Lagrange elements as the meshes become sufficiently fine and the error analysis in tables 4.7, 4.8, 4.9 and 4.10 decreasing by factor 1/4 as the size of mesh is

Table 4.6: Maximum and minimum values of y_h and y at element size $h = 6.25E-02$ and different values of λ

λ values	max(y_h)	min(y_h)	max(y)	min(y)
1	-3.88228889337	-5.77165821923	-3.89572464983	-5.78295758313
0.1	2.10766646514	1.21528938924	2.10557391269	1.21647189897
0.01	1.80199716388	1.01773682161	1.80073353302	1.01891878814
0.001	1.78090994491	1.00016222738	1.77442330197	1.00186833046
0.0001	1.78252795907	0.999716677066	1.77182825746	1.00018660078
0.00001	1.7805852458	0.999425711698	1.77156910783	1.00001865776
0.000001	1.78009556063	0.999398998038	1.77154319641	1.00000186575

Table 4.7: Error norm based on $y_h - y$ at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=1.36E-03	r=1.90
h=6.25E-02	E=3.47E-04	r=1.97
h=3.13E-02	E=8.74E-05	r=1.99
h=1.56E-02	E=2.19E-05	r=2.00
h=7.81E-03	E=5.47E-06	r=2.00
h=3.91E-03	E=1.37E-06	r=2.00

Table 4.8: Error norm based on interpolation of y onto the same space as y_h at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=1.36E-03	r=1.90
h=6.25E-02	E=3.47E-04	r=1.97
h=3.13E-02	E=8.74E-05	r=1.99
h=1.56E-02	E=2.19E-05	r=2.00
h=7.81E-03	E=5.47E-06	r=2.00
h=3.91E-03	E=1.37E-06	r=2.00

Table 4.9: Error norm based on interpolation of y to higher-order elements at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=3.54E-03	r=1.96
h=6.25E-02	E=8.94E-04	r=1.99
h=3.13E-02	E=2.24E-04	r=2.00
h=1.56E-02	E=5.61E-05	r=2.00
h=7.81E-03	E=1.40E-05	r=2.00
h=3.91E-03	E=3.51E-06	r=2.00

Table 4.10: Error norm based on infinity norm (of nodal values) at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=7.07E-03	r=1.68
h=6.25E-02	E=2.09E-03	r=1.76
h=3.13E-02	E=6.01E-04	r=1.80
h=1.56E-02	E=1.70E-04	r=1.83
h=7.81E-03	E=4.72E-05	r=1.85
h=3.91E-03	E=1.30E-05	r=1.86

decreasing by factor 1/2.
Tables 4.11, 4.12, 4.13, 4.14 shows the error analysis and convergence rate of $||y_h - y||_{l^2}$ when $\lambda = 0.001$

Table 4.11: Error norm based on $y_h - y$ at $\lambda = 0.001$

Element size	Error	CR
h=1.25E-01	E=7.47E-03	r=1.86
h=6.25E-02	E=1.91E-03	r=1.96
h=3.13E-02	E=4.82E-04	r=1.99
h=1.56E-02	E=1.21E-04	r=2.00
h=7.81E-03	E=3.02E-05	r=2.00
h=3.91E-03	E=7.54E-06	r=2.00

Table 4.12: Error norm based on interpolation of y onto the same space as y_h at $\lambda = 0.001$

Element size	Error	CR
h=1.25E-01	E=7.47E-03	r=1.86
h=6.25E-02	E=1.91E-03	r=1.96
h=3.13E-02	E=4.82E-04	r=1.99
h=1.56E-02	E=1.21E-04	r=2.00
h=7.81E-03	E=3.02E-05	r=2.00
h=3.91E-03	E=7.54E-06	r=2.00

Table 4.13: Error norm based on interpolation of y to higher-order elements at $\lambda = 0.001$

Element size	Error	CR
h=1.25E-01	E=7.85E-03	r=1.89
h=6.25E-02	E=2.01E-03	r=1.97
h=3.13E-02	E=5.04E-04	r=1.99
h=1.56E-02	E=1.26E-04	r=2.00
h=7.81E-03	E=3.16E-05	r=2.00
h=3.91E-03	E=7.89E-06	r=2.00

Table 4.14: Error norm based on infinity norm (of nodal values) at $\lambda = 0.001$

Element size	Error	CR
h=1.25E-01	E=2.45E-02	r=1.87
h=6.25E-02	E=6.49E-03	r=1.92
h=3.13E-02	E=1.64E-03	r=1.99
h=1.56E-02	E=4.11E-04	r=1.99
h=7.81E-03	E=1.03E-04	r=2.00
h=3.91E-03	E=2.57E-05	r=2.00

Tables 4.11, 4.12, 4.13 and 4.14 show that we approach the expected second-order convergence of linear Lagrange elements as the meshes become sufficiently fine and the error analysis in tables 4.11, 4.12, 4.13 and 4.14 is decreasing by the factor 1/4 as the size of mesh is decreasing by factor 1/2.
Tables 4.15, 4.16, 4.17, 4.18 show the error analysis and convergence rate of $||y_h - y||_{l^2}$ when $\lambda = 0.000001$.

Tables 4.15, 4.16, 4.17 and 4.18 shows that, we approach the expected second-order convergence of linear Lagrange elements as the meshes become sufficiently fine and the error analysis in tables 4.15, 4.16, 4.17 and 4.18 is decreasing by the factor 1/4 as the size of mesh is decreasing by the factor 1/2.

Table 4.15: Error norm based on $y_h - y$ at $\lambda = 0.000001$

Element size	Error	CR
h=1.25E-01	E=8.53E-03	r=1.83
h=6.25E-02	E=2.21E-03	r=1.95
h=3.13E-02	E=5.57E-04	r=1.99
h=1.56E-02	E=1.40E-04	r=2.00
h=7.81E-03	E=3.49E-05	r=2.00
h=3.91E-03	E=8.73E-06	r=2.00

Table 4.16: Error norm based on interpolation of y onto the same space as y_h at $\lambda = 0.000001$

Element size	Error	CR
h=1.25E-01	E=8.53E-03	r=1.83
h=6.25E-02	E=2.21E-03	r=1.95
h=3.13E-02	E=5.57E-04	r=1.99
h=1.56E-02	E=1.40E-04	r=2.00
h=7.81E-03	E=3.49E-05	r=2.00
h=3.91E-03	E=8.73E-06	r=2.00

Table 4.17: Error norm based on interpolation of y to higher-order elements at $\lambda = 0.000001$

Element size	Error	CR
h=1.25E-01	E=8.83E-03	r=1.83
h=6.25E-02	E=2.29E-03	r=1.95
h=3.13E-02	E=5.76E-04	r=1.99
h=1.56E-02	E=1.44E-04	r=2.00
h=7.81E-03	E=3.61E-05	r=2.00
h=3.91E-03	E=9.03E-06	r=2.00

Table 4.18: Error norm based on infinity norm (of nodal values) at $\lambda = 0.000001$

Element size	Error	CR
h=1.25E-01	E=3.16E-02	r=1.71
h=6.25E-02	E=8.55E-03	r=1.89
h=3.13E-02	E=2.23E-03	r=1.94
h=1.56E-02	E=5.74E-04	r=1.96
h=7.81E-03	E=1.48E-04	r=1.96
h=3.91E-03	E=3.79E-05	r=1.96

Analysis of p variable:
Tables 4.19, 4.20, 4.21, 4.22 show the error analysis and convergence rate of $||p_h - p||_{l^2}$ when $\lambda = 1$.

Table 4.19: Error norm based on $p_h - p$ at $\lambda = 1$

Element size	Error	CR
h=1.25E-01	E=4.94E-02	r=1.94
h=6.25E-02	E=1.25E-02	r=1.98
h=3.13E-02	E=3.13E-03	r=2.00
h=1.56E-02	E=7.83E-04	r=2.00
h=7.81E-03	E=1.96E-04	r=2.00
h=3.91E-03	E=4.90E-05	r=2.00

Table 4.20: Error norm based on interpolation of p onto the same space as p_h at $\lambda = 1$

Element size	Error	CR
h=1.25E-01	E=4.94E-02	r=1.94
h=6.25E-02	E=1.25E-02	r=1.98
h=3.13E-02	E=3.13E-03	r=2.00
h=1.56E-02	E=7.83E-04	r=2.00
h=7.81E-03	E=1.96E-04	r=2.00
h=3.91E-03	E=4.90E-05	r=2.00

Table 4.21: Error norm based on interpolation of p to higher-order elements at $\lambda = 1$

Element size	Error	CR
h=1.25E-01	E=4.30E-02	r=1.94
h=6.25E-02	E=1.09E-02	r=1.98
h=3.13E-02	E=2.74E-03	r=1.99
h=1.56E-02	E=6.84E-04	r=2.00
h=7.81E-03	E=1.71E-04	r=2.00
h=3.91E-03	E=4.28E-05	r=2.00

Table 4.22: Error norm based on infinity norm (of nodal values) at $\lambda = 1$

Element size	Error	CR
h=1.25E-01	E=7.67E-02	r=1.87
h=6.25E-02	E=2.03E-02	r=1.92
h=3.13E-02	E=5.30E-03	r=1.94
h=1.56E-02	E=1.38E-03	r=1.94
h=7.81E-03	E=3.58E-04	r=1.95
h=3.91E-03	E=9.28E-05	r=1.95

Tables 4.19, 4.20, 4.21 and 4.22 shows that we approach the expected second-order convergence of linear Lagrange elements as the meshes become sufficiently fine and the error analysis in tables 4.19, 4.20, 4.21 and 4.22 is decreasing by the factor 1/4 as the size of mesh is decreasing by the factor 1/2.

Tables 4.24, 4.25, 4.26, and 4.27 show the error analysis and convergence rate of $||p_h - p||_{l^2}$ when $\lambda = 0.1$.

Table 4.28 show the maximum and minimum of exact and approximate values at different values of regularize parameter. It also show that if the value of regularization parameter λ converges to zero then the values of p also converges to zero.

Table 4.23: Maximum and minimum values of p_h and p at $\lambda = 1$

Element size	$\max(p_h)$	$\min(p_h)$	$\max(p)$	$\min(p)$
h=0.25E-00	-3.97221056476	-6.23633923894	-4.16586085541	-6.42825921331
h=1.25E-01	-4.11297003668	-6.38176974782	-4.16586085541	-6.42825921331
h=6.25E-02	-4.15188046905	-6.41740814774	-4.16586085541	-6.42825921331
h=3.13E-02	-4.16220460157	-6.42576847145	-4.16586085541	-6.42825921331
h=1.56E-02	-4.16220460157	-6.42769389322	-4.16586085541	-6.42825921331
h=7.81E-03	-4.16561337759	-6.42813232194	-4.16586085541	-6.42825921331
h=3.91E-03	-4.16579664407	-6.42823110198	-4.16586085541	-6.42825921331

Table 4.24: Error norm based on $p_h - p$ at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=4.94E-02	r=1.94
h=6.25E-02	E=1.25E-02	r=1.98
h=3.13E-02	E=3.13E-03	r=2.00
h=1.56E-02	E=7.83E-04	r=2.00
h=7.81E-03	E=1.96E-04	r=2.00
h=3.91E-03	E=4.90E-05	r=2.00

Table 4.25: Error norm based on interpolation of p onto the same space as p_h at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=4.94E-02	r=1.94
h=6.25E-02	E=1.25E-02	r=1.98
h=3.13E-02	E=3.13E-03	r=2.00
h=1.56E-02	E=7.83E-04	r=2.00
h=7.81E-03	E=1.96E-04	r=2.00
h=3.91E-03	E=4.90E-05	r=2.00

Table 4.26: Error norm based on interpolation of p to higher-order elements at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=4.30E-02	r=1.94
h=6.25E-02	E=1.09E-02	r=1.98
h=3.13E-02	E=2.74E-03	r=1.99
h=1.56E-02	E=6.84E-04	r=2.00
h=7.81E-03	E=1.71E-04	r=2.00
h=3.91E-03	E=4.28E-05	r=2.00

Table 4.27: Error norm based on infinity norm (of nodal values) at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=7.67E-02	r=1.87
h=6.25E-02	E=2.03E-02	r=1.92
h=3.13E-02	E=5.30E-03	r=1.94
h=1.56E-02	E=1.38E-03	r=1.94
h=7.81E-03	E=3.58E-04	r=1.95
h=3.91E-03	E=9.28E-05	r=1.95

Table 4.28: Maximum and minimum values of p_h and p at element size $h = 6.25E-02$ and different values of λ

λ values	max(p_h)	min(p_h)	max(p)	min(p)
1	-4.15188046905	-6.41740814774	-4.16586085541	-6.42825921331
0.1	0.284765909005	0.184116169	0.284235241667	0.184199863088
0.01	0.0250710825262	0.0160935103468	0.0248410363861	0.0160983397923
0.001	0.00281864848522	0.00156641128559	0.00245318382367	0.00158979626101
0.0001	0.000289598277307	0.000155669005547	0.000245013412427	0.000158781989028

Table 4.29: Maximum values of y and p at different λ when $h = 0.25E-00$

λ	$max(y_h)$	$max(y)$	$max(p_h)$	$max(p)$
0.1	2.12826627615	2.10557391269	0.290463240571	0.284235241667
0.01	1.81262201735	1.80073353302	0.0275806222045	0.0248410363861

Table 4.30: Maximum values of y and p at different λ when $h = 1.25E-01$

λ	$max(y_h)$	$max(y)$	$max(p_h)$	$max(p)$
0.1	2.11264603657	2.10557391269	0.286114594077	0.284235241667
0.01	1.80472849737	1.80073353302	0.0256771693082	0.0248410363861

Analysis of cost function, i.e., J variable:

Now we are presenting the values $J(y,u)$, $J(y_h,u_h)$, J_{error} and CR, which represents exact values, approximate values, error analysis, and convergence rate respectively of cost functionals. In table 4.31 we see that, the exact values are same at every mesh and

Table 4.31: Value of cost function at different meshes when $\lambda = 1$

mesh	$J(y,u)$	$J(y_h,u_h)$	J_{error}	CR
(4,4)	20.6612582568	19.2165717538	1.44468650294	1.92
(8,8)	20.6612582568	20.2783732045	0.382885052288	1.98
(16,16)	20.6612582568	20.563913698	0.0973445587886	1.99
(32,32)	20.6612582568	20.6368079035	0.0244503532671	2.00
(64,64)	20.6612582568	20.6551378137	0.00612044310424	2.00
(128,128)	20.6612582568	20.659727611	0.00153064579687	2.00
(256,256)	20.6612582568	20.6608755554	0.000382701383131	

if the value of h is decreasing by factor 1/2, error is also decreasing approximately by factor 1/4.

Table 4.32: Value of cost function and its parts at different meshes when $\lambda = 0.1$

Element size	$\frac{1}{2}\|\|y_h - y_0\|\|_{\Gamma_o}$	$\frac{\lambda}{2}\|\|u_h\|\|_{\Gamma_c}$	$J(y_h,u_h)$	$J(y,u)$
h=0.25E-00	0.0239078753868	0.170979232203	0.194887107589	0.19307798933
h=1.25E-01	0.0235492016014	0.169993982706	0.193543184307	0.19307798933
h=6.25E-02	0.0234598388997	0.169735624063	0.193195462963	0.19307798933
h=3.13E-02	0.0234374945665	0.169669960761	0.193107455328	0.19307798933
h=1.56E-02	0.0234319052706	0.169653458209	0.19308536348	0.19307798933
h=7.81E-03	0.0234305075077	0.169649325938	0.193079833446	0.19307798933
h=3.91E-03	0.0234301580241	0.169648292385	0.193078450409	0.19307798933

Table 4.33: Error norm based on J_h- interpolate J at $\lambda = 0.1$

Element size	Error	CR
h=1.25E-01	E=4.65E-04	r=1.96
h=6.25E-02	E=1.17E-04	r=1.99
h=3.13E-02	E=2.95E-05	r=2.00
h=1.56E-02	E=7.37E-06	r=2.00
h=7.81E-03	E=1.84E-06	r=2.00
h=3.91E-03	E=4.61E-07	r=2.00

Table 4.34: Value of cost function and its parts at different meshes when $\lambda = 0.01$

Element size	$\frac{1}{2}\|\|y_h - y_0\|\|_{\Gamma_o}$	$\frac{\lambda}{2}\|\|u_h\|\|_{\Gamma_c}$	$J(y_h,u_h)$	$J(y,u)$
h=0.25E-00	0.000200220696535	0.0130155975549	0.0132158182514	0.0131367874757
h=1.25E-01	0.000181115363091	0.0129729890294	0.0131541043925	0.0131367874757
h=6.25E-02	0.000179244542248	0.0129616815577	0.0131409260999	0.0131367874757
h=3.13E-02	0.000179013620562	0.0129587960042	0.0131378096248	0.0131367874757
h=1.56E-02	0.000179013620562	0.0129587960042	0.0131378096248	0.0131367874757
h=7.81E-03	0.000178972445304	0.0129580698058	0.0131370422511	0.0131367874757
h=3.91E-03	0.0001789632411	0.0129578878839	0.013136851125	0.0131367874757

Table 4.37 show, if the value of regularizing parameter $\lambda \to 0$ then the value of cost function also $J \to 0$ and for convergence the tables 4.31, 4.33 and 4.35 shows that, we approach the expected second-order convergence of linear Lagrange elements as the

Table 4.35: Error norm based on J_h- interpolate J at $\lambda = 0.01$

Element size	Error	CR
h=1.25E-01	E=1.73E-05	r=2.19
h=6.25E-02	E=4.14E-06	r=2.06
h=3.13E-02	E=1.02E-06	r=2.02
h=1.56E-02	E=2.55E-07	r=2.00
h=7.81E-03	E=6.36E-08	r=2.00
h=3.91E-03	E=1.59E-08	r=2.00

$\frac{1}{2}||y_h - y_0||_{\Gamma_o}$ is the first part and $\frac{\lambda}{2}||u_h||_{\Gamma_c}$ is the second part of cost functional

Table 4.36: Value of cost function and its parts at different meshes when $\lambda = 0.001$

Element size	$\frac{1}{2}\|\|y_h - y_0\|\|_{\Gamma_o}$	$\frac{\lambda}{2}\|\|u_h\|\|_{\Gamma_c}$	$J(y_h,u_h)$	$J(y,u)$
h=0.25E-00	7.01178991024e-05	0.00130382707828	0.00137394497739	0.00126547140513
h=1.25E-01	6.59636225694e-06	0.00126748819829	0.00127408456054	0.00126547140513
h=6.25E-02	2.06393326165e-06	0.00126423034123	0.00126629427449	0.00126547140513
h=3.13E-02	1.76594036682e-06	0.00126382413882	0.00126559007919	0.00126547140513
h=1.56E-02	1.74671203616e-06	0.00126374883468	0.00126549554671	0.00126547140513
h=7.81E-03	1.74543720885e-06	0.00126373165593	0.00126547709314	0.00126547140513
h=3.91E-03	1.74534139545e-06	0.00126372746401	0.00126547280541	0.00126547140513

Table 4.37: Approximate and exact values of cost function at different λ when $h = 6.25E-02$

λ	$\frac{1}{2}\|\|y_h - y_0\|\|_{\Gamma_o}$	$\frac{\lambda}{2}\|\|u_h\|\|_{\Gamma_c}$	$J(y_h,u_h)$	$J(y,u)$
1	11.9313566801	8.63255701788	20.563913698	20.6612582568
0.1	0.0234598388997	0.169735624063	0.193195462963	0.19307798933
0.01	0.000179244542248	0.0129616815577	0.0131409260999	0.0131367874757
0.001	2.06393326165e-06	0.00126423034123	0.00126629427449	0.00126547140513
0.0001	3.68908838026e-08	0.000126116036747	0.000126152927631	0.000126076010126
0.00001	1.52006253071e-08	1.26077212776e-05	1.26229219029e-05	1.2602900441e-05
0.000001	1.50049393084e-08	1.26073896338e-06	1.27574390269e-06	1.26024304938e-06

meshes become sufficiently fine and the error analysis in tables 4.31, 4.33 and 4.35 is decreasing by the factor 1/4 as the size of mesh is decreasing by 1/2 factor.

4.3.2 Optimality system of three equations

$$-\Delta p + p = 0 \quad \text{in} \quad \Omega, \quad \partial_n p = y - y_0 \quad \text{on} \quad \Gamma_o, \quad \partial_n p = 0 \quad \text{on} \quad \partial\Omega\backslash\Gamma_o \tag{4.21a}$$

$$p \quad \text{on} \quad \Gamma_c = -\lambda u \quad \text{on} \quad \Gamma_c, \quad \text{or} \quad p = -\lambda u \quad \text{on} \quad \Gamma_c \tag{4.21b}$$

$$-\Delta y + y = f \quad \text{in} \quad \Omega, \quad \partial y_n = u \quad \text{on} \quad \Gamma_c, \quad \partial_n y = 0 \quad on \quad \partial\Omega\backslash\Gamma_c \tag{4.21c}$$

Consider the system (4.10) and simplifying it we get

$$(\nabla p, \nabla\varphi)_\Omega + (p,\varphi)_\Omega - (y,\varphi)_{\Gamma_o} - \lambda(u,\chi)_{\Gamma_c} - (p,\chi)_{\Gamma_c} + (\nabla y, \nabla\psi)_\Omega$$
$$+(y,\psi)_\Omega - (u,\psi)_{\Gamma_c} = (f,\psi)_\Omega - (y_0,\varphi)_{\Gamma_o} \quad \forall\psi,\varphi \in V \text{ and } \chi \in Q$$

we implemented the above equation at different meshes and at different regularize parameter (λ), some of the results are shown in below figures

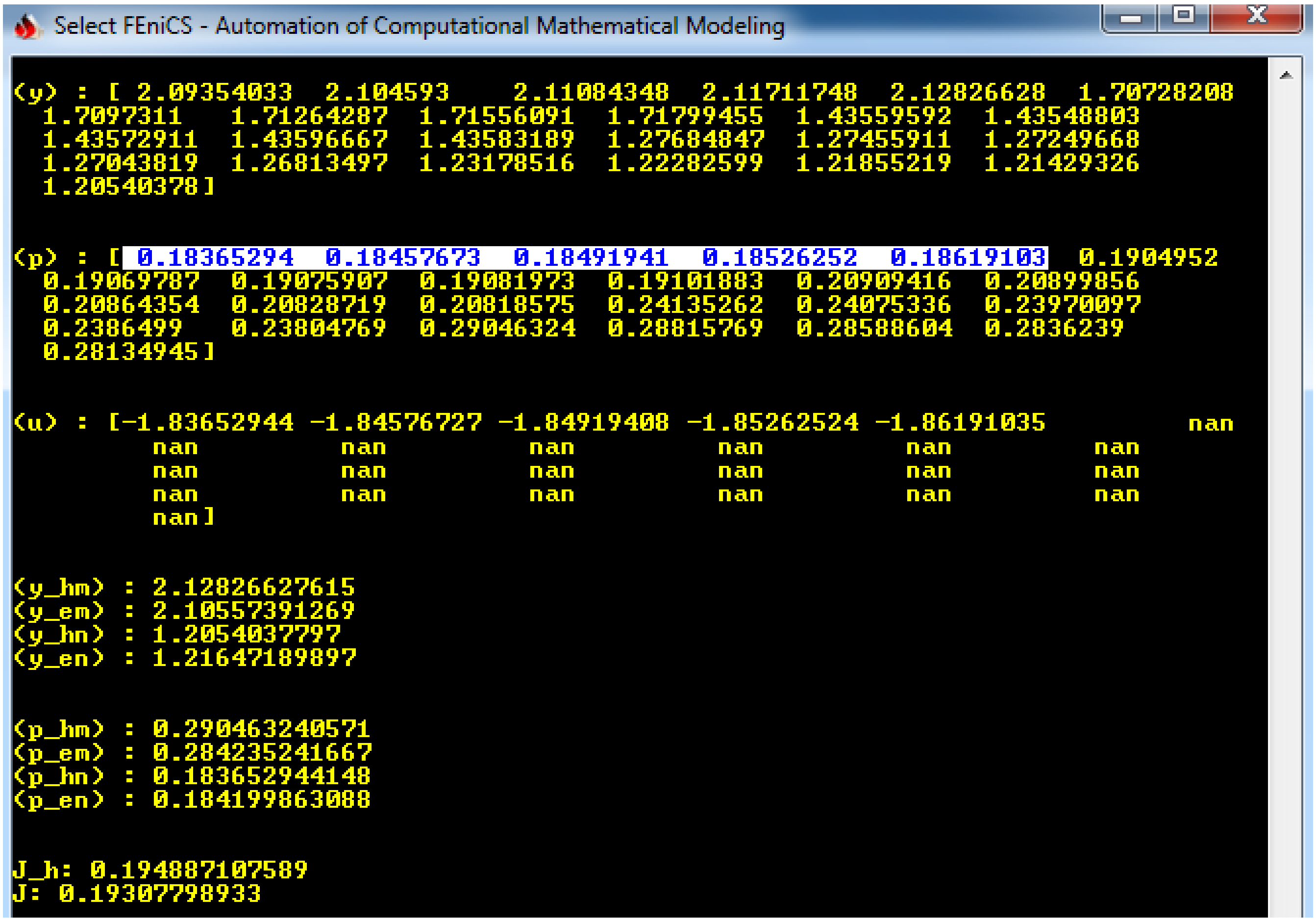

Figure 4.1: Output window of the implemented problem at mesh (4,4) when $\lambda = 0.1$

Now we compare the results of optimality system of two equations and optimality system of three equations, Fig (4.1) and Fig (4.2) show that the values of u is defined

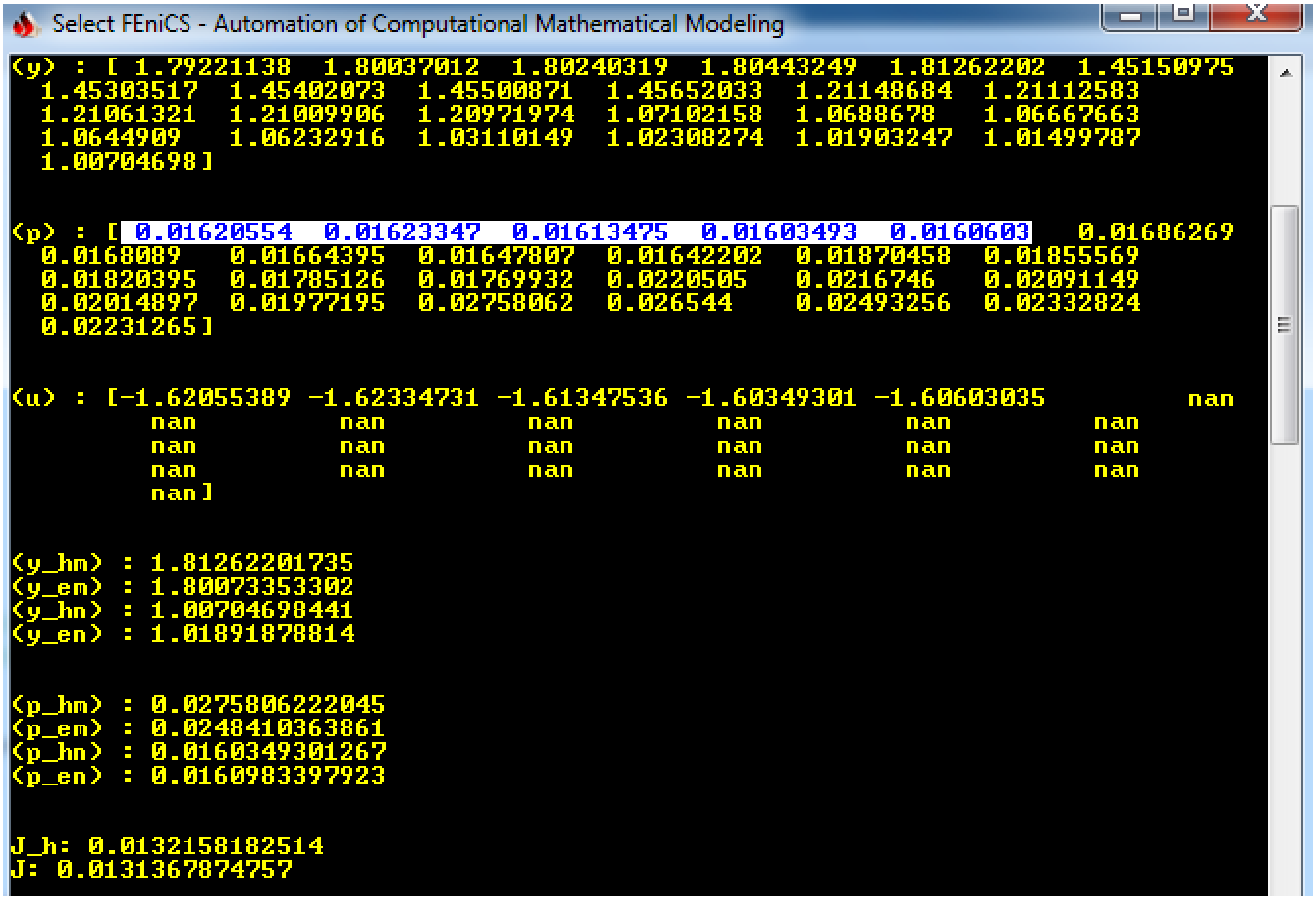

Figure 4.2: Output window of the implemented problem at mesh (4,4) when $\lambda = 0.01$

at 5 nodes when mesh is (4,4), these are because of our control u is living on control boundary (λ_c) and similarly Fig (4.3) and Fig (4.4) show the value of u is defined at 9 nodes when mesh is (8,8). More over above all figures also satisfied the condition

$$u = -\frac{1}{\lambda} p \quad \text{at} \quad \Gamma_c$$

which we used in optimality system of two equations ,i.e, (4.19a). The figures of optimality system of three equations ,i.e., (4.1), (4.2), (4.3) and (4.3) and the tables of optimality system of two equations (4.29), (4.30), (4.32) and (4.34) show that the evaluated exact and approximate values of cost functional, maximum and minimum values of y and p variables at different regularize parameters and at different length sizes are exactly same values. Then obviously the error analysis and convergence is the same in both ways.

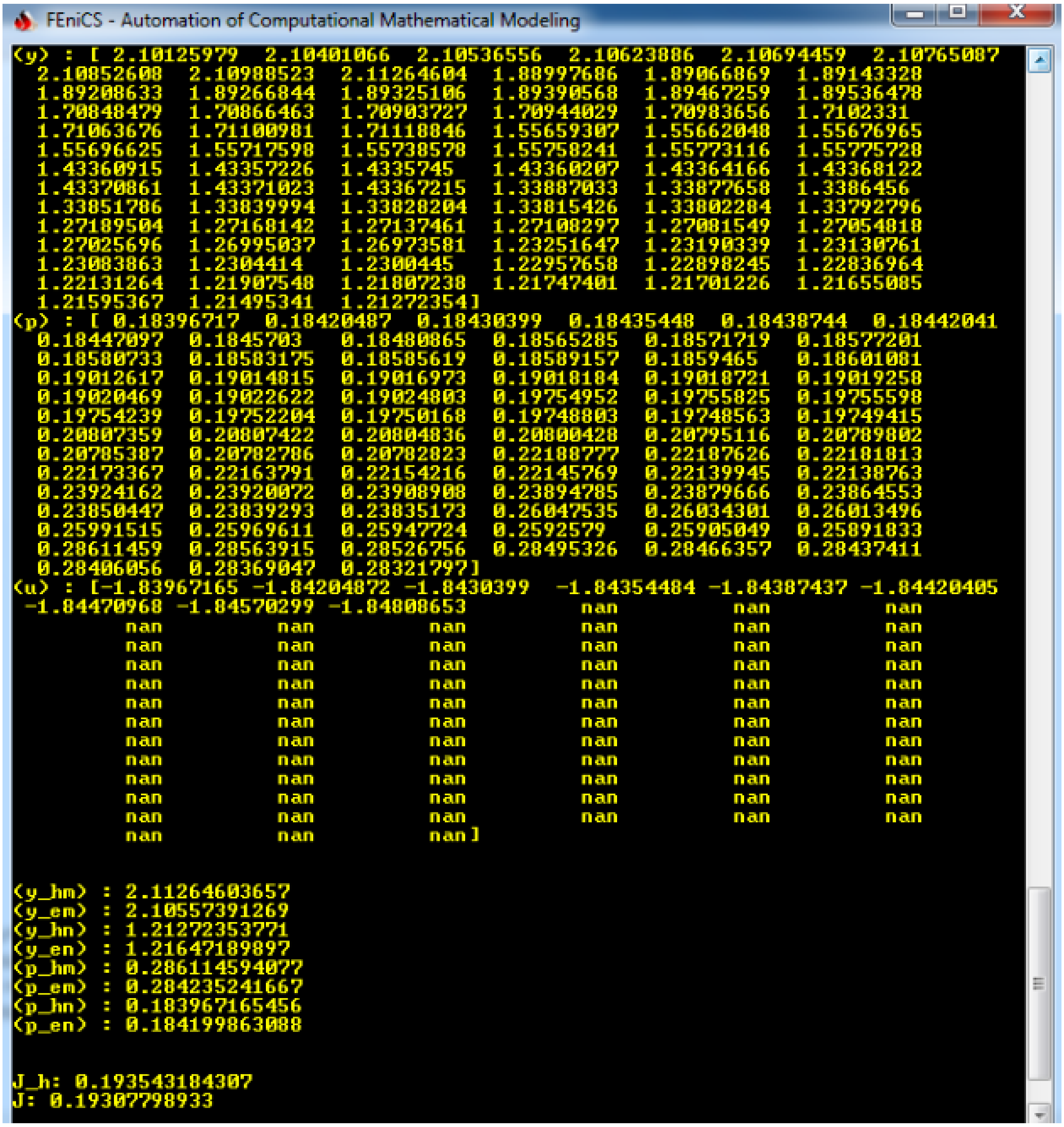

Figure 4.3: Output window of the implemented problem at mesh (8,8) when $\lambda = 0.1$

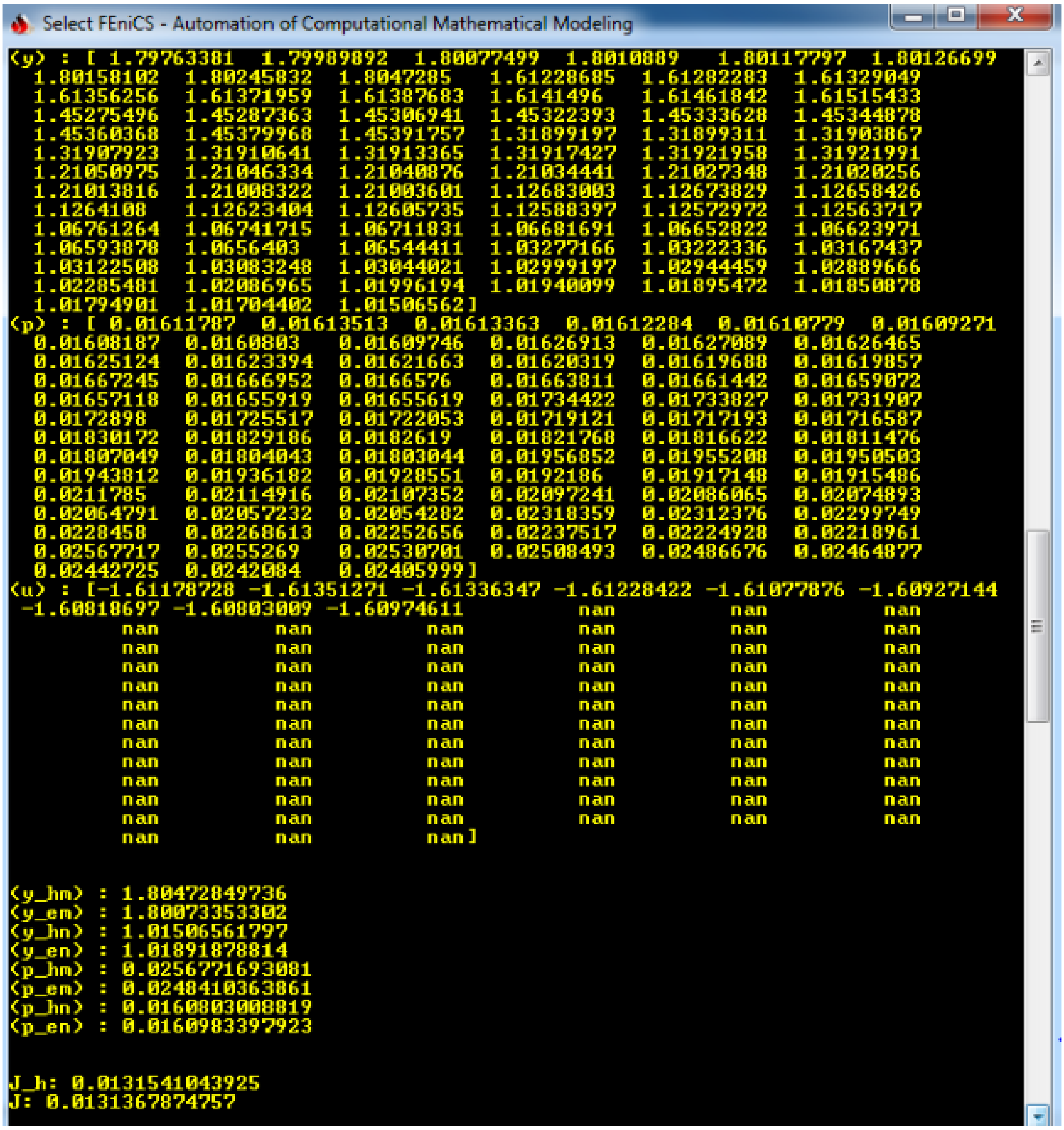

Figure 4.4: Output window of the implemented problem at mesh (8,8) when $\lambda = 0.01$

4.3.3 Coding for FEniCS software in Python Language

from dolfin import *
from math import *

```
import sys
mesh = UnitSquare (nx,ny)
X = FunctionSpace (mesh ,'Lagrange',1)
Y = FunctionSpace (mesh ,'Lagrange',1)
Z = X*Y
(y,p) = TrialFunctions(Z)
(si,phi) = TestFunctions(Z)

# Define Sourse values
f = Expression ('x[1] * x[1]/2 - x[1]')
y0 = Constant ('1')
d = Constant('1') # regularize parameter
lam_c = Expression ('-sinh(1)/(1 - p * sinh(1) * sinh(1)) * (u0 + (1/sinh(1)) - .5)',u0 =
1,d = 1)
a = inner(grad(phi),grad(p))*dx+phi*p*dx-phi*y*ds(1)+inner(grad(y),grad(si))*
dx + y * si * dx - (1/d) * p * si * ds(0)
L = f * si * dx - y0 * phi * ds(1)
#boundary
boundary_parts = MeshFunction ('uint', mesh, 1)
#Mark lower boundary facets as subdomain 0

class LowerNeumannBoundary(SubDomain):
def inside(self, x, on_boundary):
tol = 1E - 14 # tolerance for coordinate comparisons
return on_boundary and abs(x[1]) < tol

L = LowerNeumannBoundary()
L.mark(boundary_parts, 0)

# Mark upper boundary facets as subdomain 1
class UpperNeumannBoundary(SubDomain):
def inside(self, x, on_boundary):
tol = 1E - 14 # tolerance for coordinate comparisons
return on_boundary and abs(x[1] - 1) < tol

U = UpperNeumannBoundary()
U.mark(boundary_parts, 1)
#all of the Rest boundaries
class RestNeumannBoundary(SubDomain):
def inside(self, x, on_boundary):
tol = 1E - 14 # tolerance for coordinate comparisons
return on_boundary and (abs(x[0]) < tol or abs (x[0] - 1) < tol )
```

```
#Verification
Xe = FunctionSpace(mesh, 'Lagrange', 5)
y_e = interpolate(y_exact,Xe)
Ye = FunctionSpace(mesh, 'Lagrange', 5)
p_e = interpolate ( p_exact, Ye,)
ze = FunctionSpace(mesh, 'Lagrange', 5)
u_e = interpolate(u_exact ,ze)

error_sq = (y-y_e)*(y-y_e)*dx
error =sqrt(assemble(error_sq))
#print " y_error:",error

error_sq = (p-p_e)*(p-p_e)*dx
error =sqrt(assemble(error_sq))
#print "p_error:",error

error_sq = (u-u_e)*(u-u_e)*dx
error =sqrt(assemble(error_sq))

R = RestNeumannBoundary()
R.mark(boundary_parts,23)
M= assemble(y_0 * phi * ds(1),exterior_facet_domains=boundary_parts)
# Compute solution
A = assemble(a, exterior_facet_domains=boundary_parts)
b = assemble(L, exterior_facet_domains=boundary_parts)
s = Function(Z)
solve(A,s.vector(),b)
(y,p) = s.split()
print '(y,p) :' ,s.vector().array()
print ' n '
print '(y) :',y.vector().array()
print' n '
print '(p) :', p.vector().array()
print '(M) :', M.array()

#Verification
y_e = Expression ('(cosh(x[1] - 1)/(1 - d * sinh(1) * sinh(1))) * (y_0 + 1/sinh(1) - .5)
- (cosh(x[1] - 1)/sinh(1)) + x[1] * x[1]/2 - x[1] + 1',y_0 = 1,d = g)#0.1)
p_e = Expression('((d * sinh(1) * cosh(x[1]))/(1 - d * sinh(1) * sinh(1))) *
(y_0 + (1.0/sinh(1)) - .5)',y_0 = 1,d = g)#0.1)

B= (.5,.5)
print 'y_e at the centor:',y_e(B)
```

```
print 'y at the centor:',y(B)
print 'p_e at the centor:',p_e(B)
print 'p at the centor:',p(B)

cost=inner(y-y_0,y-y_0)*ds(1)+(1.0/d)*inner(p,p)*ds(0)
J_h=(1.0/2)*assemble(cost, exterior_facet_domains=boundary_parts)

coste=inner(y_e_Ve-y_0,y_e_Ve-y_0)*ds(1)+(1.0/d)*inner(p_e_Ve,p_e_Ve)*ds(0)
J_ex=(1.0/2)*assemble(coste, exterior_facet_domains=boundary_parts)
E5=abs(J_h-J_ex)
t_j=f_pv+s_pv

print '(y_m) :' , max (y.vector().array())
print '(y_em) :' , max (y_e_Ve.vector().array())
print '(y_n) :' , min (y.vector().array())
print '(y_en) :' , min (y_e_Ve.vector().array())
print " n"

print 'f_pv:', f_pv
print 's_pv:',s_pv
print 'tot:',t_j
print 'J_h(y,u):',J_h
print 'J_e(y,u):',J_ex

print '(p_m) :' , max (p.vector().array())
print '(p_em) :' , max (p_e_Ve.vector().array())
print '(p_n) :' , min (p.vector().array())
print '(p_en) :' , min (p_e_Ve.vector().array())

#Plot solution
plot(y,title="yplot")
plot(p,title="pplot")
plot(mesh,title="mesh")
interactive()
```

5

Discussion and Conclusion

The optimality system which we get by two different algorithms, i.e., "Lagrange's method" or "Reduced functional" has been successfully implemented in two different ways (optimality system of two equations and optimality system of three equations) in this work. The results achieved by optimality system of two equations and by optimality system of three equations are promising and exactly are same. The results in terms of approximate values or exact values of state variable y, the Lagrangian multiplier's variable p and the cost functional $J(y,u)$ considerably. More interestingly the condition

$$u = -\frac{1}{\lambda}p \quad \text{at} \quad \Gamma_{\text{c}}$$

which we have inserted in (4.10c) is also satisfied in implementation results of optimality system of three equations. As earlier we have mentioned that $\lambda > 0$ is a regularization parameter, so we see that in tables 4.6, 4.28 and 4.37 if the value of regularize parameter is decreasing then the value of y, p and J respectively is also decreasing with the same order. We mean if the regularize parameter converges to zero then according to tables 4.6, 4.28 and 4.37 the value of y, p and J respectively also converges to zero. Furthermore we have successfully implemented the error analysis and convergence rate in four different ways of variable's and achieved nice results, according to the tables of error analysis, if the element size is decreasing by the factor 1/2 then the error is decreasing by the factor 1/4 in each way, similarly we approach the expected second-order convergence of linear Lagrange elements as the meshes become sufficiently fine at different values of $\lambda > 0$ of every variable. We also evaluated the maximum and minimum values of variable and function at different $\lambda' s$ and at different element size's, and achieved same results in both implemented ways.

The implemented optimality system we derived by two methods 1. by Lagrange's method 2. by reduced functional, however by Lagrange's method we first get the optimality system in weak form and then for strong form we applied Green's formula, but in reduced functional we directly get the same optimality system in strong form.

FEniCS software is specific tool for automated computational solution of differential equations by Finite Element Method and good thing in FEniCS software is, it directly recognize the coding of variational formulation from Python or C++ languages, but we used to write the coding in Python language, however there is the need of improving so that we can define the variable at a boundary.

As in abstract we have mentioned that the goal in this work is to implement the optimal control problem in FEniCS when cost functional consists on two different boundaries or when the observed temperature and control temperature both living on two different boundaries (observed boundary Γ_o and control boundary Γ_c). Now our control temperature u is a variable living only on control boundary, i.e., Γ_c and according to the active member of FEniCS software's team that it's not possible to have variables living only on the boundary then we inserted the condition

$$u = -\frac{1}{\lambda} p \quad \text{at} \quad \Gamma_c$$

in (4.10c) and achieved above said nice results and the results of optimality system of three equations have been shown in the figures 4.1, 4.2, 4.3 and in 4.4, in first two figures u is defined at five nodal values at mesh (4,4) values and in last two figures the u is defined at nine nodal values at mesh (8,8), these are because of our control variable u lives only on Γ_c not in Ω, in these figures above condition is also satisfied and similarly error analysis and convergence are also same of both implemented ways.

Bibliography

[1] F. Tröltzsch, Optimal Control of Partial Differential Equations Theory, Methods and Applications, Vol. 112, American Mathematical Society, United States of America, 2010.

[2] S. Larsson, V. Thomée, Partial Differential Equations with Numerical Methods, Springer-Verlag, Berlin, 2003.

[3] P. Philip, Optimal Control of Partial Differential Equations, 2009.
URL `http://www.math.lmu.de/~philip/publications/lectureNotes`

[4] G. Da Prato, An introduction to infinite-dimensional analysis, Universitext, Springer-Verlag, Berlin, 2006.

[5] L. Beilina, M. V. Klibanov, Approximate Global Convergence and Adaptivity for Coefficient Inverse Problems, Springer, New York, 2012.

[6] W. Bangerth, R. Rannacher, Adaptive Finite Element Methods for Differential Equations, Birkhäauser Verlag, Basel, Switzerland, 2003.

[7] F. Tröltzsch, On Finite Element Error Estimates for Optimal Control Problems with Elliptic PDEs.
URL `http://page.math.tu-berlin.de/~troeltz/arbeiten/sozopol09.pdf`

[8] E. Casas, F. Tröltzsch, Second Order Analysis for Optimal Control Problems: Improving Results Expected from Abstract Theory, SIAM J. Optim. 22 (2012) 261–279.
URL `http://epubs.siam.org/doi/pdf/10.1137/110840406`

[9] V. Thomée, Galerkin Finite Element Methods for Parabolic Problems, 2nd Edition, Vol. 25 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 2006.

[10] N. Andréasson, A. Evgrafov, M. Patriksson, An Introduction to Continuous Optimization, 2005.

[11] C. Meyer, U. Prüfert, F. Tröltzsch, On two Numerical Methods for State-Constrained Elliptic Control Problems.
URL `http://page.math.tu-berlin.de/~troeltz/arbeiten/mpt05_small.pdf`

[12] A. Logg, K.-A. Mardal, G. N. Wells, Automated Solution of Differential Equations by the Finite Element Method, 2011.

[13] Logg, Mardal, Wells, Automated Scientific Computing, 2009.