Computer Homework
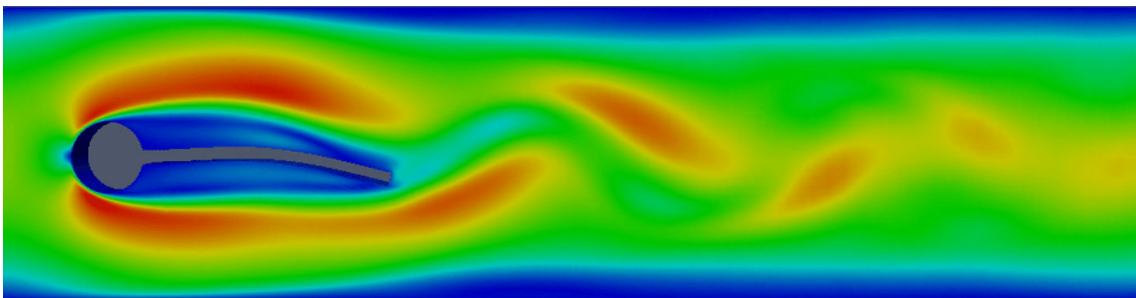## Coupled Problems

Sebastian Ares de Parga Regalado

Master in Numerical Methods in Engineering
Universitat Politècnica de Catalunya
17$^{\text{th}}$ June, 2020

Contents

1 Introduction

This report show the implementation of a Matlab code for 1D heat transfer coupled problems with its respective analysis, working with iterative and non-iterative schemes. The heat transfer equation is shown below:

$$-\kappa \frac{\partial^2 u}{\partial x^2} = f$$

Where κ stands for the thermal diffusive coefficient; u is the temperature of the domain (solution of the problem) and f represents the source term.

1.1 Tasks

1. Solve a single heat transfer problem. The domain is [0,1]. Fix u = 0 in both boundaries.

   (a) Study the effect of changing the value to the thermal diffusion coefficient kappa.

   (b) Study the effect of changing the source term value.

   (c) Study the effect of changing the number of elements, evaluate the convergence rate of the error in the maximum heat value in the domain.

2. Solve two independent heat transfer problems with kappa = 1, source = 1. The first problem subdomain is [0, 0.25]. The second problem subdomain is [0.25,1]. Fix u in x=0 and x=1, leave it free in the interface between sub-domains. Comment on the results.

3. Solve the previous problem in a Monolithic way.

   (a) Study **HP_SolveMonolithic.m** and relate it to what was explained in theory. Comment on the results.

   (b) Modify the kappa parameter of one of the sub-domains. Comment on the results.

4. Solve the previous problem (kappa = 1 in both sub-domains) in an iterative manner (Dirichlet Neumann). Apply Neumann boundary conditions at the interface in the first (left) subdomain, and Dirichlet boundary conditions at the interface in the second subdomain.

   (a) Evaluate the convergence of the iterative scheme (in terms of u at the interface).

   (b) Increase the value for kappa at subdomain 1 (x100). Comment on the convergence rate.

   (c) Diminish the value for kappa at subdomain 1(/100). Comment on the convergence rate.

   (d) Motivate the previous results in terms of the stability of the coupling scheme.

5. Implement a relaxation scheme

   (a) Relaxation scheme in terms of a fixed relaxation parameter w.

   (b) Aitken relaxation scheme.

2 Problem 1

For this first problem, a single domain [0,1] is to be computed and analyzed. The boundary conditions established for this specific problem are only Dirichlet, being $u = 0$ at $x = 0$ and $x = 1$. The analysis will consist in varying some of the problem parameters to show the physical and numerical behaviour.

2.1 Solution (a)

To show the response on the solution (temperature) of the problem when varying the thermal diffusion coefficient κ, 5 different values for alpha from 2 to 10 were simulated with a 100 element mesh and $f = 1$. The following image shows the results:
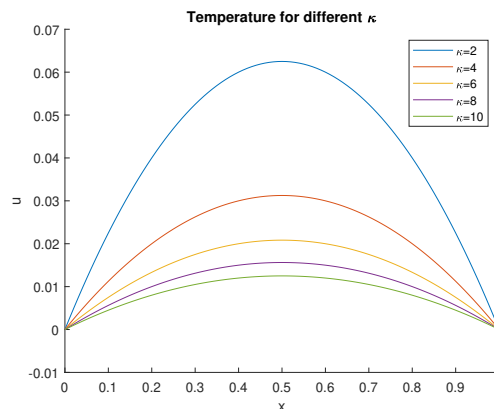

Figure 1: Solution for different values of κ

Notice that as κ becomes higher, there is a significantly reduction of the temperature through all the domain. This is physically acceptable since a material with high diffusivity moves the heat rapidly, dissipating the external heat sources (f) efficiently.

2.2 Solution (b)

Now, a variation of the source term will be simulated to analyze the effects on the temperature over the domain, the problem is simulated with a 100 element mesh and a thermal diffusion coefficient $\kappa = 1$, while the source term will be varying from 2 to 10. The results are shown in the following table:
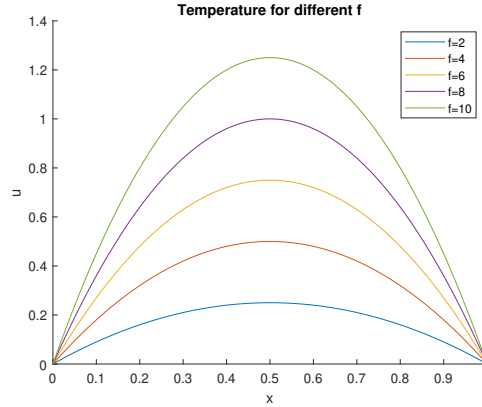
3

Figure 2: Solution for different values of f

We can notice that know, as the source term gets higher, the temperature over the domain increases. This complements the analysis made for the case where κ was varying show that as the source term gets higher, we will need a bigger diffusivity to present lower temperatures over the domain.

To show better the behaviour of the source term compared to the thermal diffusion coefficient, different simulations for the same values for both f and κ was carried out, and its depicted in the following graph:
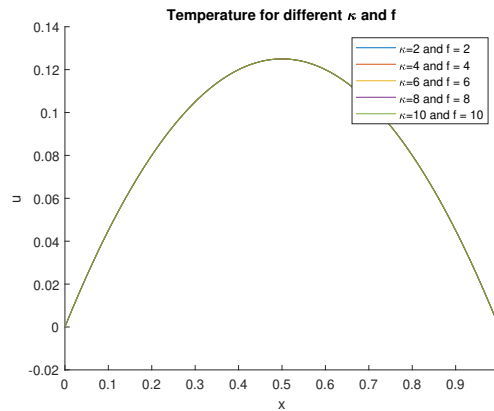

Figure 3: Solution for different values of f and κ

Notice that the solution for the 5 different cases is the same, proving the previously stated.

2.3   Solution (c)

In this case, we will perform a convergence analysis of the solution for different mesh sizes (number of elements in the domain). The chosen parameters to perform this analysis are $\kappa = 1$ and $f = 1$, and with the help of a 1000 elements size mesh we obtained the exact solution at the $x = 0.5$ where the maximum temperature is presented analytically.
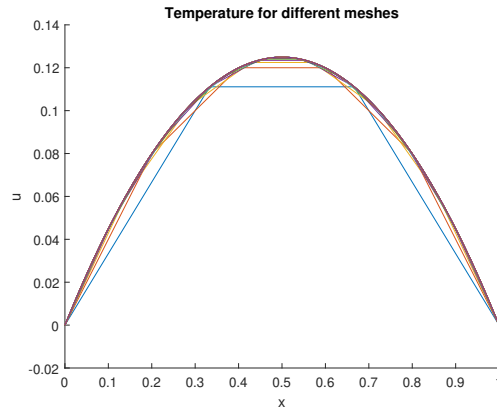
Figure 4: Solution for different meshes

Note: Odd numbers were used for the convergence analysis, because of the exact nodal solution, when using even numbers the convergence check would have been altered since no error will be presented for the maximum value because it lies exactly at the mid point of the domain.
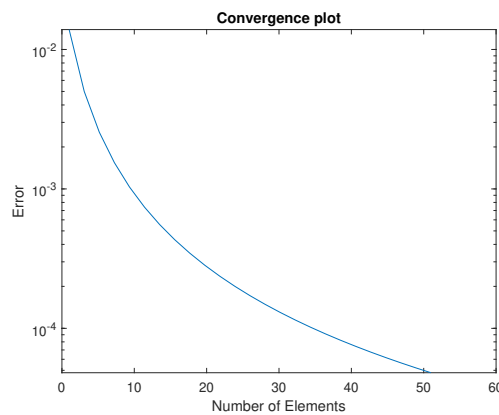

Figure 5: Convergence analysis: Error vs Number of elements

A semi-logarithmic convergence plot is presented in this figure, to show better the relative error obtained for certain number of elements, but if the graph was to be plotted in logarithmic scale for both the y and x axis, a linear behaviour was presented.

3 Problem 2

The second problem states to solve a two sub-independent problem, where the initial domain [0,1] is splitted into to two sub-domains:

- Sub-domain Ω_1:
  This domain goes from [0,0.25] and will be represented with 25 elements, with a fixed Dirichlet condition $u = 0$ at $x = 0$.

- Sub-domain Ω_2:
  This domain goes from [0.25,1] and will be represented with 75 elements, with a fixed Dirichlet condition at $x = 1$.

For both sub-domains, the thermal diffusion coefficient will be $\kappa = 1$ and the source term $f = 1$. It is worth to mention that the interface between sub-domains will be free of boundary conditions.

3.1 Solution:

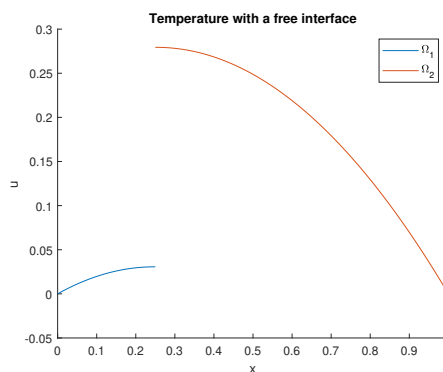This simulation was performed an the results obtained are shown in the following figure:

Figure 6: Solution for Ω_1 and Ω_2 with a free interface.

We can notice that a jump in the solution for both sub-domains is presented as expected, since both problems where solved for their own boundary conditions and no transmission conditions were imposed. The most important part of this problem is to notice that the problem needs the transmission conditions to be fully coupled, even if we impose Dirichlet conditions at the interface, the problem will have the proper solution.
To prove this, a simulation was performed with the same Dirichlet boundary conditions $x = 0$ and $x = 1$ ($u = 0$) and on the interface the Dirichlet boundary conditions were established as $u = 1$.
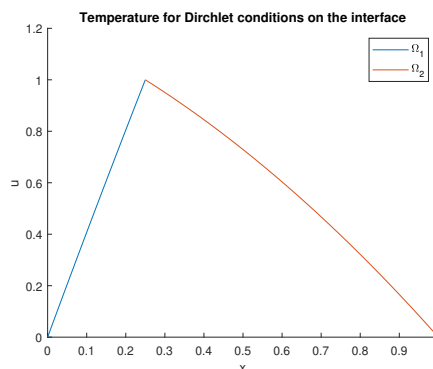

Figure 7: Solution for Ω_1 and Ω_2 with a Dirichlet conditions on the interface.

6

It can be noticed that the solution of each subdomain does not take any conditions from the other subdomain, this is solving the two problems by separated and coinciding on the interface due to the fixed Dirichlet condition. We can noticed this by solving separately as for the first problem of this assignment as a single problem.
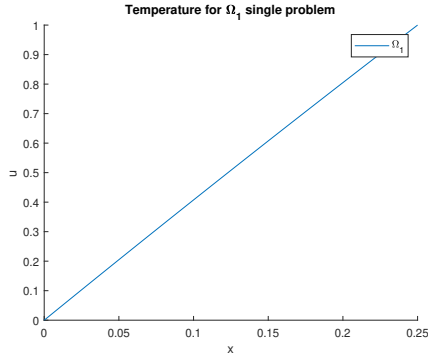
(a) Solution for Ω_1 single problem
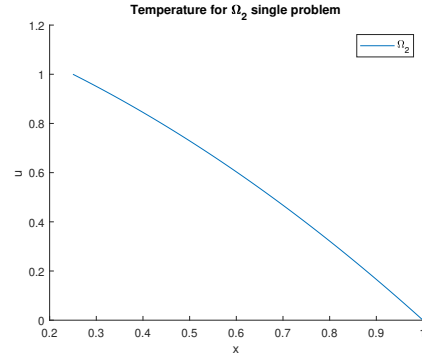
(b) Solution for Ω_2 single problem

Notice that no difference on the solution was presented, therefore, we have again proved that what we need to impose a solution of a coupled problem are the transmission conditions.

4   Problem 3

As stated on the previous problem, to solve it transmission conditions have to be applied. This can be done by applying a Neumann-Neumann coupling approach. Where the sub-domains are defined for the following weak forms:

- Sub-domain Ω_1:

$$\int_{\Omega_1} \frac{\partial w_1}{\partial x} \kappa_1 \frac{\partial u_1}{\partial x} - \int_{\Gamma} w_1 \kappa_1 \frac{\partial u_1}{\partial x} = \int_{\Omega_1} w_1 f$$

- Sub-domain Ω_2:

$$\int_{\Omega_2} \frac{\partial w_2}{\partial x} \kappa_2 \frac{\partial u_2}{\partial x} - \int_{\Gamma} w_2 \kappa_2 \frac{\partial u_2}{\partial x} = \int_{\Omega_1} w_2 f$$

Neumann-Neumann:

$$-\kappa_1 \frac{\partial u_1}{\partial x} = \kappa_2 \frac{\partial u_2}{\partial x}$$

Substituting this equation into the sub-domain Ω_2:

$$\int_{\Omega_2} \frac{\partial w_2}{\partial x} \kappa_2 \frac{\partial u_2}{\partial x} + \int_{\Gamma} w_2 \kappa_1 \frac{\partial u_1}{\partial x} = \int_{\Omega_1} w_2 f$$

Since the meshes match on the interface, the imposition of the transmission conditions is straightforward and the equations for the first domain is exactly the same, so there

is no need to integrate the boundary terms and we can simply assemble the global system and use the same unknowns at the interface. Therefore we can say that:

$$\int_\Gamma w_2 \kappa_1 \frac{\partial u_1}{\partial x} = \int_\Gamma w_1 \kappa_1 \frac{\partial u_1}{\partial x} = \int_{\Omega_1} \frac{\partial w_1}{\partial x} \kappa_1 \frac{\partial u_1}{\partial x} - \int_{\Omega_1} w_1 f$$

This exact case is done in the matlab code for the $HP\_SolveMonolithic.m$ function when assembling both sub-domains.

4.1 Solution (a)

In this case, the problem 2 initial problem is to be solved in a monolithic way. The parameters of the problem where stated the same:

- Sub-domain Ω_1:
  This domain goes from [0,0.25] and will be represented with 25 elements, with a fixed Dirichlet condition $u = 0$ at $x = 0$.

- Sub-domain Ω_2:
  This domain goes from [0.25,1] and will be represented with 75 elements, with a fixed Dirichlet condition at $x = 1$.

For both sub-domains, the thermal diffusion coefficient will be $\kappa = 1$ and the source term $f = 1$. The problem was solved and the results are presented in the figure below:
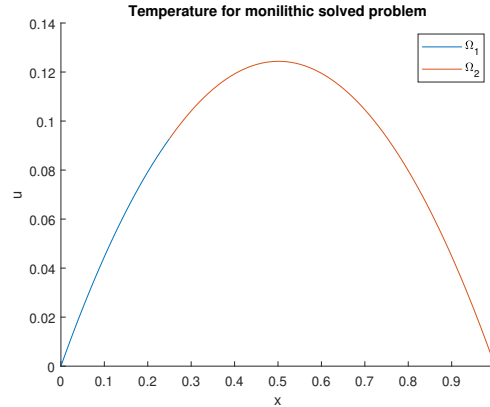

Figure 9: Solution for Ω_1 and Ω_2 in a monolithic way.

We now noticed that the transmission conditions make a whole difference on the solution and considers the input of both sub-domains to solve a coupled problem. Since the problem has the same parameters for the source term and κ, the diffusivity will behave similar for those cases, and sub-domain Ω_2 will behave as an extension of sub-domain Ω_1. If we decide to change the thermal diffusion coefficient, we expect to have a totally different solution (the amount of change will only depend on the of the change in κ).

4.2   Solution (b):

Now the same problem is solved, but a change in sub-domain Ω_2 was made for the thermal diffusion coefficient, which was multiplied by 100 times ($\kappa_2 = 100$). The results are presented on the following figure.
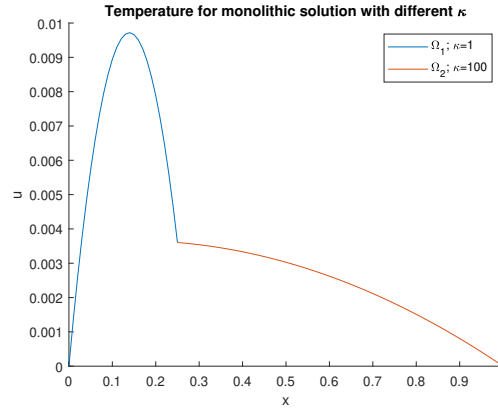

Figure 10: Solution for Ω_1 and Ω_2 in a monolithic way with different κ.

We can noticed a whole different solution, in which for the subdomain ω_1 which is the one with the lowest value of κ tends to be slower in terms of moving the heat through its domain, meanwhile for the case of sub-domain Ω_2, it moves diffuses a lot quicker. It is important that this behaviour creates a visible jump on the derivatives in the interface, but since the transmission condition is always satisfied, it can be proven that there is a continuity of the flux on the interface.

5   Problem 4

The same problem ($\kappa_1 = \kappa_2 = 1$) is to be solved with an iterative scheme (with Dirichlet-Neumann). Applying Neumann boundary ($\frac{\partial u}{\partial x} = 25$) conditions at the interface in the sub-domain Ω_1, and Dirichlet boundary conditions at the interface in the sub-domain Ω_2.

> **Sub-domain Ω_1 :**
>
> $$-k_1 \frac{\partial^2 u_1^{[k]}}{\partial x^2} = f_1 \longrightarrow in \;\; \Omega_1$$
>
> $$u_1 = 0 \longrightarrow on \;\; \partial\Omega_1$$
>
> $$-k_1 \frac{\partial u_1^{[k]}}{\partial x} = k_2 \frac{\partial u_2^{[k-1]}}{\partial x} \longrightarrow on \;\; \Gamma$$

> **Sub-domain Ω_2 :**
>
> $$-k_2 \frac{\partial^2 u_2^{[k]}}{\partial x^2} = f_2 \longrightarrow in \ \ \Omega_2$$
>
> $$u_2 = 0 \longrightarrow on \ \ \partial\Omega_2$$
>
> $$u_2^{[k]} = u_1^{[k]} \longrightarrow on \ \ \Gamma$$

To perform this, we will first solve the sub-domain Ω_1 with the established Neumann condition on the interface and set the Dirichlet condition of the sub-domain Ω_2 on the interface the solution obtained from Ω_1 as follows:

```
Data2.FixLeft = 1;
Data2.LeftValue = HeatProblem.Solution.uRight;
```

Then, we can solve the sub-domain Ω_2 and send the obtained value of the flux to the first subdomain and established as a Neumann boundary condition to solve it again. This will be done until a condition of tolerance is reached. The iteration process in Matlab was implemented as follows:

```
error=[];%Intiialize error
condition= true;%Initialize condition for iteration
tol=1e-4;%Tolerance
cont=0;%Initialize counter

    Comparacion1=HeatProblem.Solution.uRight;%First comparison
while condition
    cont=1+cont;
    Data.RightFluxes = -HeatProblem2.Solution.FluxesLeft;
    %Solve sub-domain 1
    HeatProblem = HP_Initialize(Data);
    HeatProblem = HP_Build(HeatProblem);
    HeatProblem = HP_Solve(HeatProblem);
    %Assign Dirichlet value to sub-domain 2
    Data2.LeftValue = HeatProblem.Solution.uRight;
    %Solve sub-domain 2
    HeatProblem2 = HP_Initialize(Data2);
    HeatProblem2 = HP_Build(HeatProblem2);
    HeatProblem2 = HP_Solve(HeatProblem2);
    Comparacion2=HeatProblem.Solution.uRight;%Second comparison
    error(cont)=abs(Comparacion1-Comparacion2);%Error
    condition=(error>tol);%Check condition (true/false)
    Comparacion1=Comparacion2;%New comparison
    %Break condition
    if cont>100
    break
    end
end
```

The simulation was done and the results are depicted in the following figures:
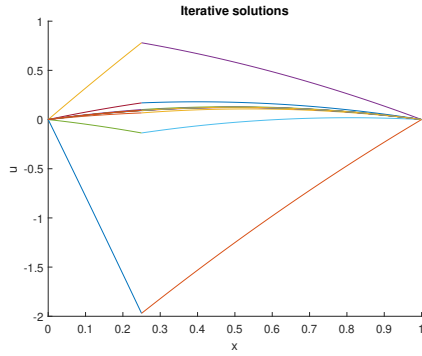
(a) Iterative solutions for Ω_1 and Ω_2.
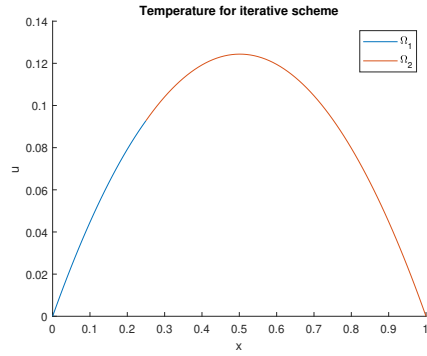
(b) Solution for Ω_1 and Ω_2 with an iterative scheme.
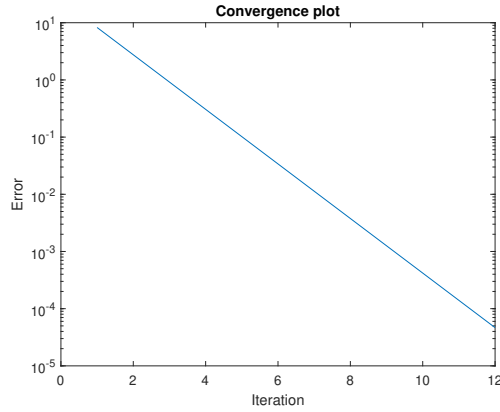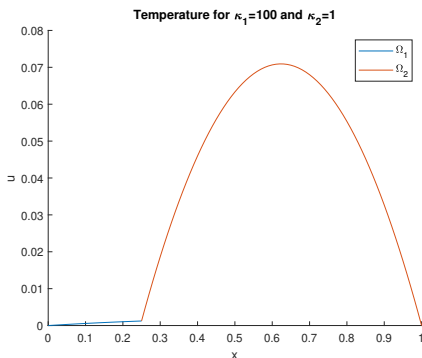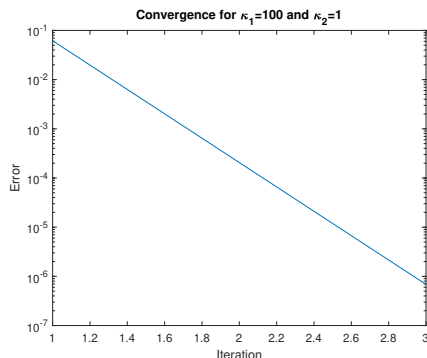

Figure 12: Convergence for Ω_1 and Ω_2 with a iterative scheme.

If we now solve this problem for with $\kappa_1 = 100$ and $\kappa_2 = 1$, the numerical solution obtained is the following:


(a) Solution for Ω_1 and ω_2 with $\kappa_1 = 100$ and $\kappa_2 = 1$ with an iterative scheme


(b) Convergence for Ω_1 and ω_2 with $\kappa_1 = 100$ and $\kappa_2 = 1$ with an iterative scheme

Notice that it behaved as previously discussed in other problems where κ is not the same in both sub-domains, but if we now solve this problem for with $\kappa_1 = 1$ and $\kappa_2 = 10$ (note that the ratio between κ_1 and κ_2 is lower than the last one), the numerical solution obtained is the following:
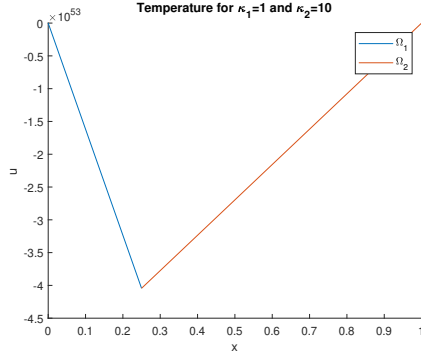
(a) Solution for Ω_1 and ω_2 with $\kappa_1 = 1$ and $\kappa_2 = 100$ with an iterative scheme
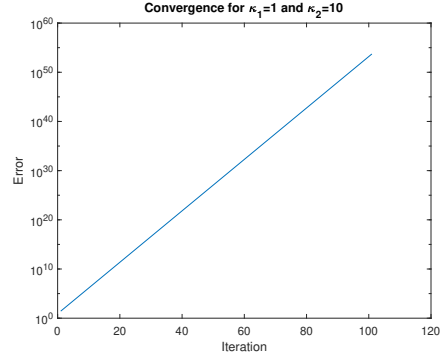
(b) Convergence for Ω_1 and ω_2 with $\kappa_1 = 1$ and $\kappa_2 = 100$ with an iterative scheme

Notice that now our problem diverges when iterating, and this behaviour can be explained through the following concept:

Continuity of the fluxes:

$$-k_1 \frac{\partial u_1^{[k]}}{\partial x} = k_2 \frac{\partial u_2^{[k-1]}}{\partial x}$$

$$\frac{\partial u_1^{[k]}}{\partial x} = -\frac{k_2}{k_1} \frac{\partial u_2^{[k-1]}}{\partial x}$$

This means that when as the ratio $\frac{k_2}{k_1}$ decreases, we converge faster, on the other hand as this ratio increases, our iterative scheme starts to present instabilities and will drag us to a divergence of the solution as presented when $\kappa_2 = 10$.

6   Problem 5

6.1   Solution (a)

For this problem, we are asked to implement an iterative scheme with a relaxation parameter, which will lead us to reduce the number of iterations implemented to converge and also will introduce stability to the scheme to solve problems in which the ratio between κ_2 and κ_1 decreases as in the previous problem. This relaxation will be introduced on the Dirichlet boundary of the interface using the transmission condition as follows:

$$u_{\Gamma_{21}}^{[k]} = \theta u_{\Gamma_{12}}^{[k]} + (1 - \theta) u_{\Gamma_{21}}^{[k-1]}$$

Therefore, the first step will be as the iterative method implemented on the previous problem, and then this relaxation on the transmission condition has to be added at the iteration loop, as follows:

12

```
theta=0.25;%Relaxation parameter
.
.
.

    %Assign Dirichlet value to sub-domain 2
    Data2.LeftValue = theta*HeatProblem.Solution.uRight...
    +(1-theta)*HeatProblem2.Solution.uLeft;
```

Once this was implemented, we solved the initial problem for different values of θ to check the number of iterations we need to solve the same problem with the implementation of the relaxation with respect to the normal iterative scheme, the results are presented below:
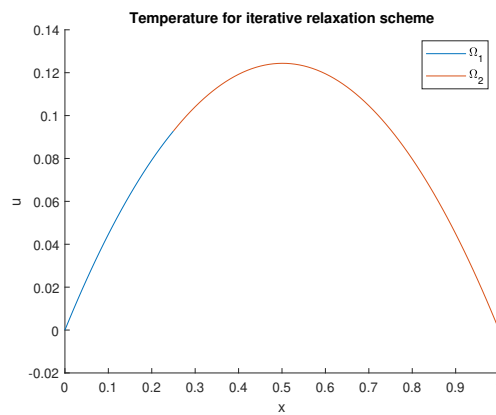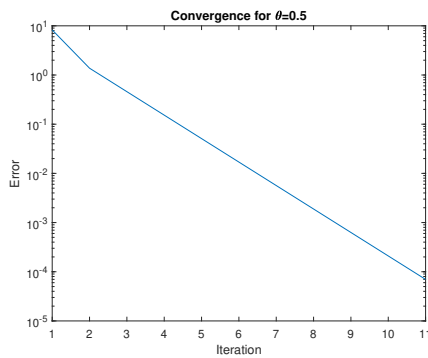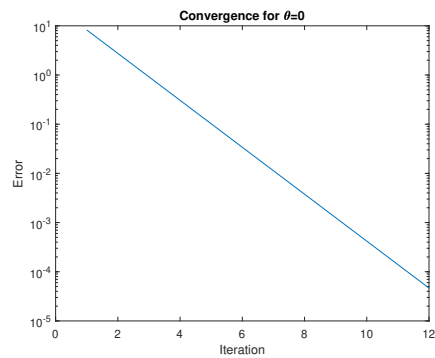

Figure 15: Temperature for iterative relaxation scheme.

(a) Convergence for $\theta = 1$

(b) Convergence for $\theta = 0.5$

Note that if $\theta = 1$, we recover the iterative scheme with no relaxation, and if $\theta = 0.5$ the number of iteration to converge to the tolerance decreased in 1. Now if we propose an optimal $\theta = 0.75$ the results obtained were the following:
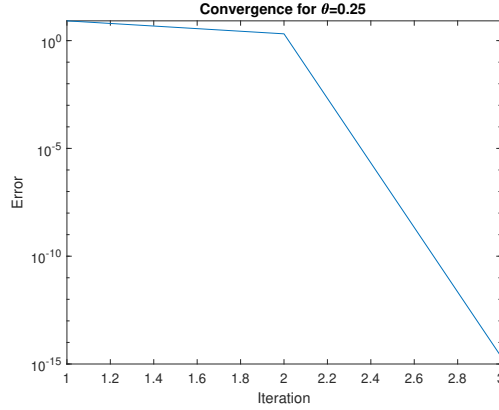
13

Figure 17: Convergence for $\theta = 0.75$

We decreased the number of iterations to converge to the same tolerance from 12 to 3 iterations.

If we now simulate the case of $\kappa_1 = 1$ and $\kappa_2 = 10$, which lead us to instability problems in the non-relaxed iterative scheme, will produce similar instabilities if the value of $\theta = 0.75$ is used, therefore, we solved for a new optimal value which was $\theta = 0.25$. The results are depicted in the following figures:
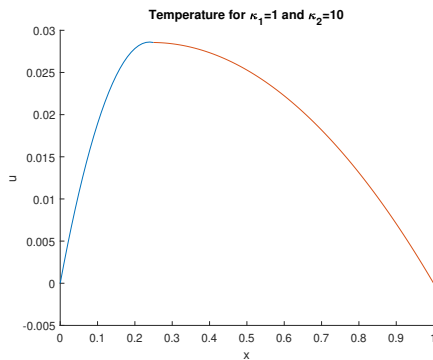
(a) Temperature for $\kappa_1 = 1$ and $\kappa_2 = 10$
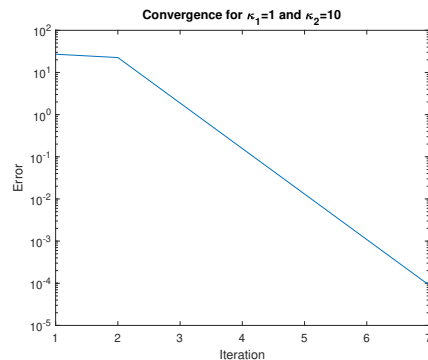
(b) Convergence for $\kappa_1 = 1$ and $\kappa_2 = 10$

We can notice that now we were able to successfully converge to the solution which was unstable without relaxation. We can also conclude that depending on the ratio of κ's we will have to use a different optimal relaxation parameter θ.

Solution (b): To solve this problem regarding the change of θ depending on the ratio of the κ's of the problem, an implementation of the Aitken relaxation scheme. This relaxation scheme, unlike the one previously used, offers a change of theta depending on the results obtained in the Dirichlet conditions on the interface for different steps, defined as follows:

$$\theta = \frac{u_{\Gamma_{21}}^{[k-2]} - u_{\Gamma_{21}}^{[k-1]}}{(u_{\Gamma_{21}}^{[k-2]} - u_{\Gamma_{21}}^{[k-1]}) + (u_{\Gamma_{12}}^{[k]} - u_{\Gamma_{12}}^{[k-1]})}$$

14

To implement this, we first solve for two iterations and save the Dirichlet values for those 2 steps in the left boundary and only the last Dirichlet value for the right domain (saved in variables named *valorxx*) and enter into an iterative scheme that follows the algorithm presented below:

```
while condition
    cont=1+cont;
    valor21=HeatProblem2.Solution.uLeft;
    Data.RightFluxes = -HeatProblem2.Solution.FluxesLeft;
    %Solve sub-domain 1
    HeatProblem = HP_Initialize(Data);
    HeatProblem = HP_Build(HeatProblem);
    HeatProblem = HP_Solve(HeatProblem);
    valor10=HeatProblem.Solution.uRight;
    %Aitken's relaxation parameter
    theta=(-valor21+valor22)/(-valor21+valor22+valor10-valor11);
    %Assign Dirichlet value to sub-domain 2
    Data2.LeftValue = theta*valor10+(1-theta)*valor21;%Relaxation
    valor11=valor10;
    valor22=valor21;
    %Solve sub-domain 2
    HeatProblem2 = HP_Initialize(Data2);
    HeatProblem2 = HP_Build(HeatProblem2);
    HeatProblem2 = HP_Solve(HeatProblem2);

    Comparacion2=HeatProblem.Solution.uRight;%Second comparison
    error(cont)=abs(Comparacion1-Comparacion2);%Error
    condition=(error>tol);%Check condition (true/false)
    Comparacion1=Comparacion2;%New comparison
    %Break condition
    if cont>100
    break
    end
end
```
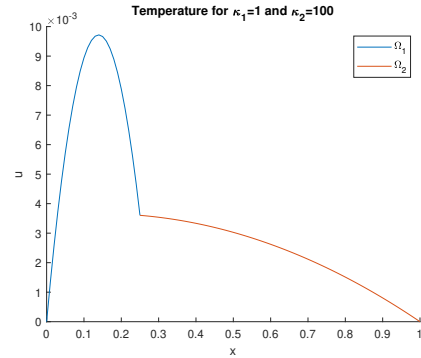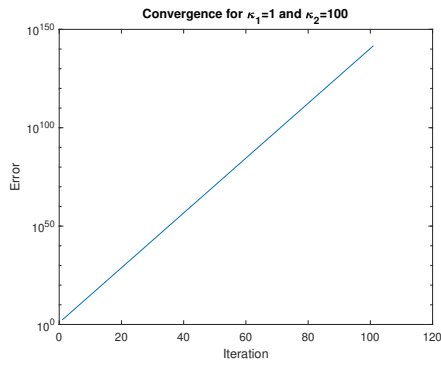
To prove this relaxation scheme works, we will simulate the case for $\kappa_1 = 1$ and $\kappa_2 = 100$ with the normal relaxation scheme vs the Aitken's relaxation scheme below:
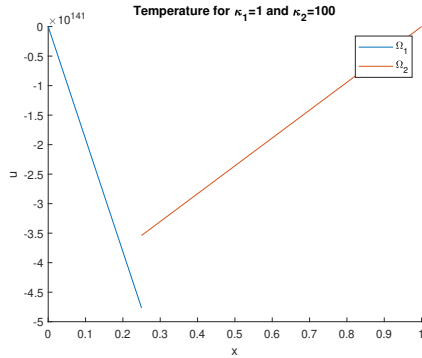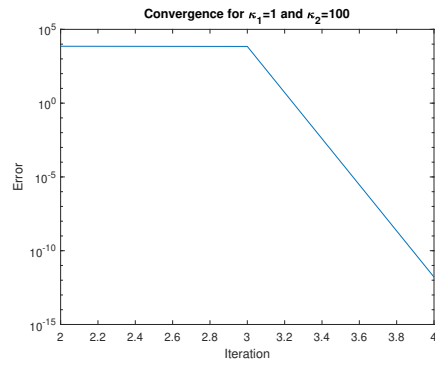

(a) Normal relaxation scheme. Temperature for $\kappa_1 = 1$ and $\kappa_2 = 100$

(b) Aitken's relaxation scheme. Temperature for $\kappa_1 = 1$ and $\kappa_2 = 100$

(a) Normal relaxation scheme. Convergence for $\kappa_1 = 1$ and $\kappa_2 = 100$

(b) Aitken's relaxation scheme. Convergence for $\kappa_1 = 1$ and $\kappa_2 = 100$

We can notice that Aitken's relaxation scheme solves the instabilities on the solution of this coupled problem when subjected to a high ratio of κ's.