**Western University, Department of Mechanical and Materials Engineering**

**MME 9710: Advanced CFD**

Assignment 2: Introduction to One Dimensional Transient Analysis

**Submitted By:** Alexander K. Kiar, 250731557

**Submitted To:** Dr. Straatman

**Date:** May 19th, 2017

**Table of Contents**

[1. **Fortran Solve**r 3](#_Toc483995189)

[1.1. Transient Discretization4 3](#_Toc483995190)

[1.2. Discretization Methods 4](#_Toc483995191)

[1.2.1. Explicit Discretization 4](#_Toc483995192)

[1.2.2. Implicit Discretization 4](#_Toc483995193)

[1.3. main.f 5](#_Toc483995194)

[1.4. difphi.f 5](#_Toc483995195)

[1.5. srct.f 5](#_Toc483995196)

[1.6. coeff.f 5](#_Toc483995197)

[2. **Code Verification** 5](#_Toc483995198)

[2.1. Analytic Solution 6](#_Toc483995199)

# Fortran Solver

,

Before completing the assignment problem, the Fortran code built for one dimensional diffusion analysis was updated to include routines for transient problems. The updated code includes a loop for a converged solution for each time step. The following outlines the discretization methods and the changes made to each file of the solver.

## Transient Discretization4

Transient discretization is an extension of the one-dimensional diffusion discretization. The transport equation over a set of control volumes (Figure 1) is expanded to integrate over the time domain in addition to volume:



Figure : Control Volume Set.

Where is the “new time”, , is the “old time”, , and is the weighting factor which controls the time discretization method. In general, the discretized transport equation takes the form:

The new transport equation caused the diffusion coefficient, source terms, and active coefficients to be updated to include a transient term:

## Discretization Methods

Different types of transient discretization methods are available. The transient discretization can either be implicit or explicit, or a combination of both. The type of discretization method used determines how the solver uses the previous iteration in the solution of the current iteration.

### Explicit Discretization

Explicit discretization generates the solution of the next iteration based on the previous solution. The solution in the explicit formulation does not require the solution of a system of equations since the formulation does not require iteration at . The computational requirements of the explicit formulation are less; however, it places a constraint the size of the time step. Since the source term is active, the coefficient of the previous must be positive to model the physical system correctly. This requires the time step size to be,

Where,

This restricts the time step based on the time step associated with the conduction through the end faces of the control volume of interest. This imposes a large restriction on time step size, and may not create an accurate solution.

### Implicit Discretization

,

A fully implicit solution removes the transient source term from the transport equation. This removes any restriction on the size of the time step. The fully implicit solution requires the solution of a system of equations based on the current iteration temperatures of each face of the control volumes.

A combination of discretization techniques can be used for . This modifies the amount of the previous source term included in the transport equation. When , the discretization is known as the Crank-Nicolson method. This method does not restrict the time step size, but can produce oscillatory solutions at large time steps since the solution may not converge.

## main.f

The main routine was updated to include initialization, time step, and temperature saving loops; error calculations were also included. A loop was added after the temperature field was initialized to reset the initial temperature field to the first time step of interest. This was implemented by assigning the analytic solution to each control volume node. Additionally, this loop initialized “TOLD”, the previous iteration temperature.

A time step loop was then added that encompassed the convergence process built for one dimensional diffusion analysis. This loop allowed a converged solution to be reached for each time step before moving to the next. Within this loop, the previous iteration temperature was updated after the converged solution for each time step was reached. This temperature field was used to update the active coefficients.

Lastly, an error calculation was added outside of the time step loop. Once the final time step had been reached, the temperature field was compared to the analytic solution. The error was calculated for each node and saved for analysis.

## difphi.f

The GAMA variable was the only updated to the difphi.f subroutine. Instead of passing the conductivity only, was passed. This was changed based on the derivation of the transient transport equation.

## srct.f

,

The srct.f subroutine was modified to include the transient term of source term. Additionally, the unsuccessful linearization techniques tested in assignment 1 were removed, leaving only the Newton-Raphson method.

## coeff.f

The coeff.f subroutine was modified to include the transient terms of the active coefficients. These terms were added to the equations used for the first assignment.

# Code Verification

The code was verified by analyzing transient conduction through a plane wall. The geometry of the problem is given in Figure 2, and the properties can be found in Table 1. Since dimensionless parameters were used for the transient points of interest, the material properties were selected to satisfy those parameters.



Figure : Problem geometry

Table : Problem Properties



## Analytic Solution

<

The analytic solution was used to initialize the temperature field and evaluate the error of the final solution of each time step size. The analytic solution was found using the one-term Fourier solution of a plane wall,

Where and are dimensionless parameters given by,

The analytic solution was evaluated for both values given in Table 1. The analytic solution took advantage of the symmetry of the plane wall, as only half of the distribution was solved for. The dimensionless parameter was varied from 0 to 1, based on the size of the control volume. The control volume size was kept constant for each time step trial since 40 control volumes were specified. The analytic solution can be seen in Figure 3.

Figure : Analytic solutions at time steps of interest

## Code Implementation

<

The problem was implemented by setting boundary conditions, and varying the size and number of time steps used. The problem was solved after building in the required loops for transient analysis.

To replicate the symmetry present in the plane wall, different boundary conditions were used for the beginning and end nodes. A Neumann boundary condition was used for the beginning node as it was modelled as an insulated surface, and a Robin condition was used for the end node since it was exposed to convection. Table 2 summarizes the boundary conditions used.

Discretization was performed in the spatial and time domains. The size of the control volumes was kept constant across all time step sizes. Time steps were varied from 2 to 32 by dividing up the difference between dimensionless values. A summary of the discretization used for each time step size can be found in Table 3.

Table : Boundary conditions



Table : Summary of transient and spatial discretization



## Results

The time steps were varied for two discretization schemes. A fully implicit scheme was used first and verified before moving to the Crank-Nicolson method. Once the