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

**MME 9710: Advanced CFD**

Assignment 3: Introduction to Combined Diffusion/Convection Analysis

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

**Submitted To:** Dr. Straatman

**Date:** June 13th, 2017

**Table of Contents**

[1. **Fortran Solver** 3](#_Toc485125036)

[1.1. Advection Discretization 3](#_Toc485125037)

[1.2. main.f 5](#_Toc485125038)

[1.3. inital.f 5](#_Toc485125039)

[1.4. masflx.f 5](#_Toc485125040)

[1.4.1. coefcn 5](#_Toc485125041)

[1.4.2. masflx 6](#_Toc485125042)

[1.5. weight.f 6](#_Toc485125043)

[1.6. hoconv.f 6](#_Toc485125044)

[1.7. srct.f 6](#_Toc485125045)

[1.8. coeff.f 7](#_Toc485125046)

[2. **Problem 1** 8](#_Toc485125047)

[2.1. Code Implementation 8](#_Toc485125048)

[2.2. Results 9](#_Toc485125049)

[3. **Problem 2** 14](#_Toc485125050)

[3.1. Code Implementation 14](#_Toc485125051)

[3.2. Results 16](#_Toc485125052)

[4. **Appendix 1** 20](#_Toc485125053)

# Fortran Solver

Before completing the assignment problems, the Fortran code built for one dimensional diffusion and transient analysis was updated to include routines for advection problems. The updated code includes various subroutines to calculate each component of the advection discretization. The following sections outline the discretization method and the changes made to each subroutine of the solver. Printouts of the updated code can be found in Appendix 1.

## Advection Discretization

The advection problem adds a second heat flux through each control volume face (see Figure 1). In each problem the flow field is known. The convective heat flux is given by , and the diffusive flux is given by in the equations below.



Figure : Advection geometry

The inclusion of a convective term requires the discretization of both the mass and energy transport equations. The 1D mass equation is given by:

The 1D energy equation is given by:

The energy equation is integrated over a control volume and time step giving the following discretized equation:

Where it is assumed that:

To guarantee that mass is conserved over each control volume and remains independent of temperature level, the conservation of mass equation is multiplied by a reference temperature (assumed to be ), and subtracted from the energy equation. The discretized mass equation is derived by integrating over a control volume and time step giving:

Multiplying the above equation by and subtracting it from the energy equation gives:

If explicit time integration is assumed for both convection and diffusion, multiple restrictions are placed on the time step and control volume sizes. These restrictions stem from the assumption that the temperatures on the East and West faces of each control volume are derived from the central diffusion scheme, where and are given by:

To avoid these restrictions, an implicit scheme is implemented, changing the energy equation to:

This formulation allows for very large time steps, but still restricts the size of the control volumes due to the CDS temperature approximation. To remove this restriction, the unwinding scheme was implemented. The UDS scheme was used to remove the control volume restriction for all ranges of Peclet number. The Peclet number is given by:

The Peclet number measures the ratio between convective and diffusive heat transfer. The CDS approximation is only valid for Peclet numbers close to zero, where diffusion dominates the problem. This is because pure diffusion is linear, and the temperature on a control volume face can be approximated as the average of the temperatures between two control volumes centers. The UDS scheme removes this restriction because it carries the temperature from the previous control volume through the domain, which reflects the physical process of fluid flow.

The UDS scheme is given by:

Where is a weighting factor based on the direction of flow. Using the UDS scheme, the discretization becomes:

Where each coefficient is given by

This discretization ensures that all ranges of Peclet number can be modelled while keeping each coefficient positive.

## main.f

No major changes were made to the main routine. The only modifications made were call lines to the various subroutines required for the implementation of the advection scheme.

## inital.f

This subroutine was modified to initialize the flow field and the average velocity at the east face of each control volume. Variables were added to in.dat to set the initial velocities.

## masflx.f

,

### coefcn

<

The coefcn subroutine was built to calculate the coefficients of the mass conservation equation:

Where,

### masflx

This subroutine was built to calculate the mass fluxes through the east face of each control volume. The mass flux was given by,

The mass flux through the beginning boundary node was given by,

## weight.f

This subroutine calculates the weighting factors on each control volume face. The factors are used in the UDS temperature approximation. The boundary nodes are set outside of the interior face loop to ensure that the correct temperature is being passed into the domain based on the direction of flow. The interior faces are set by calling the subroutine prfl, which sets the value based on the direction of the mass flux for the control volume in question.

## hoconv.f

<,

The hoconv.f subroutine implemented the deferred correction approach for higher order temperature interpolation. A varaiable was added to in.dat to choose pure UDS or the corrected CDS, or QUICK schemes. The deferred correction coefficient was given by,

Where BETA was set to one for each trial, TE was calculated using the UDS scheme, and THOS was calculated based on the higher order correction selected. The CDS temperature was calculated using,

And the QUICK scheme was calculated using,

The higher order schemes are included in the correction term to keep the stability of UDS while adding the accuracy of a higher order scheme. Hardcoding the UDS scheme guarantees that positive coefficients will be found with a refined mesh.

## srct.f

The source routine was modified to add the deferred correction terms to QT. The linearized term RT was not modified. The QT term became,

## coeff.f

This subroutine was modified to include advection terms in the active coefficients. The only coefficients that were modified were,

The advection terms include the heat transfer that stems from the mass flux through the east face of each control volume, and the weight factor for the direction of flow.

# Problem 1

The first problem involved convection and diffusion driven by fluid flow through a duct, with prescribed temperatures on each end. The geometry of the problem can be found in Figure 2, and the properties can be found in Table 1.



Figure : Problem 1 geometry

Table : Problem 1 properties



## 2.1. Code Implementation

The problem was implemented by setting boundary conditions in bndct.f and weight.f, varying the number of control volumes, and testing each higher order correction scheme. Table 2 shows the discretization used for each correction scheme, and Table 3 shows the boundary conditions for the beginning and end nodes. One time step was used for each trial, and was implemented by creating a very large delta t, and limiting the time loop to one iteration. This removed any transient effect from the solution. Dirichlet boundary conditions were used on both ends of the duct to specify each temperature. The boundary conditions were set to ensure that the temperature was carried through the domain correctly since the UDS scheme was implemented. The positive IB-1 boundary condition ensured that the west face temperature of the IB node was equal to the west temperature of IB-1. The negative IE boundary condition ensured that the east face temperature of the IE node was equal to the east temperature of the boundary node.

Table : Problem 1 spatial discretization



Table : Problem 1 boundary conditions



## 2.2. Results

<

For each number of control volumes the temperature distribution was plotted. Composite plots of each number of control volumes can be found in Figures 3 through 6. The average residual was calculated for each iteration, and every solution converged to the residual criteria. Each scheme created negative coefficients for various control volume sizes. For each scheme, it was determined that close to 25 control volumes caused all coefficients to become positive. The coefficients were negative because of the error approximation assumed for each scheme. The error is a result of the truncated Taylor series expansion, and is a function of the order of the scheme. Using the deferred correction method reduces the error, but still causes negative coefficients and affects the stability of the solution. For low numbers of control volumes, the UDS scheme keeps the trend of producing large negative temperatures in its solution. However, the CDS and QUICK schemes oscillate for low control volumes in addition to producing negative temperatures.

After 20 control volumes were used, the deferred correction terms became small and each solution traced the same distribution. All solutions converged on the analytic when 40 control volumes were used. This verifies that each scheme is valid for modelling the combined problem. However, all three schemes had difficulty correctly modelling the extreme gradient when the distribution begins to turn towards the IE + 1 boundary condition. This difficulty stems from the interpolation techniques used to calculated the temperature between control volumes. The gradient at that point is highly non-linear, but a second order scheme was the highest order used to try and model the distribution. Had a higher order scheme been used the solution would have approached the analytic at the high gradient area.

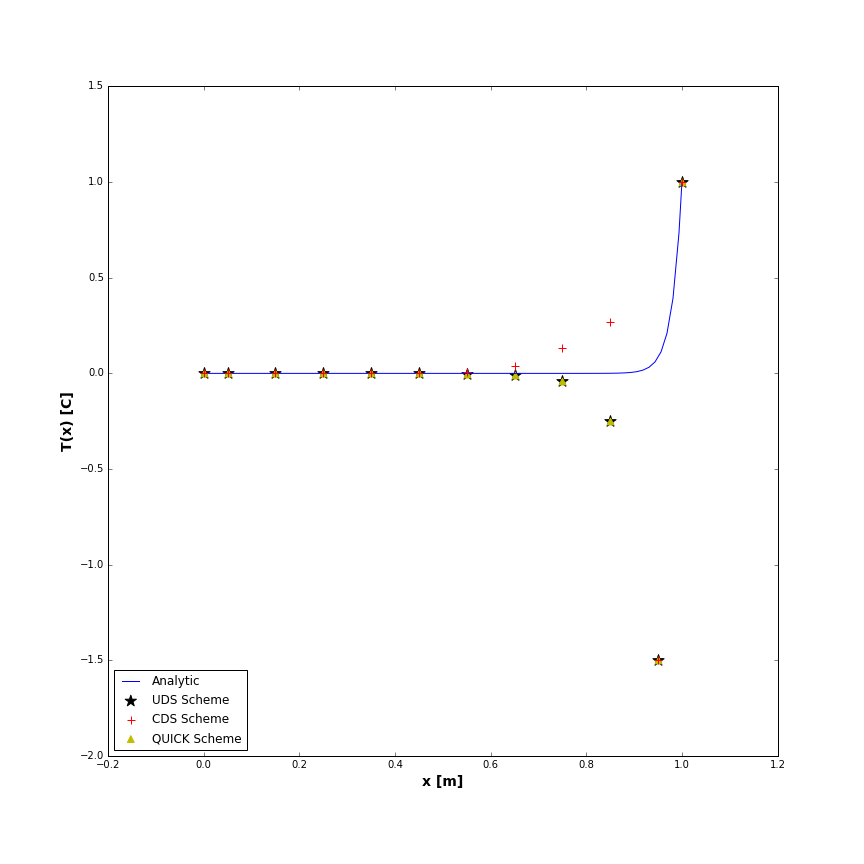


Figure : Problem 1, 10 control volumes for all schemes

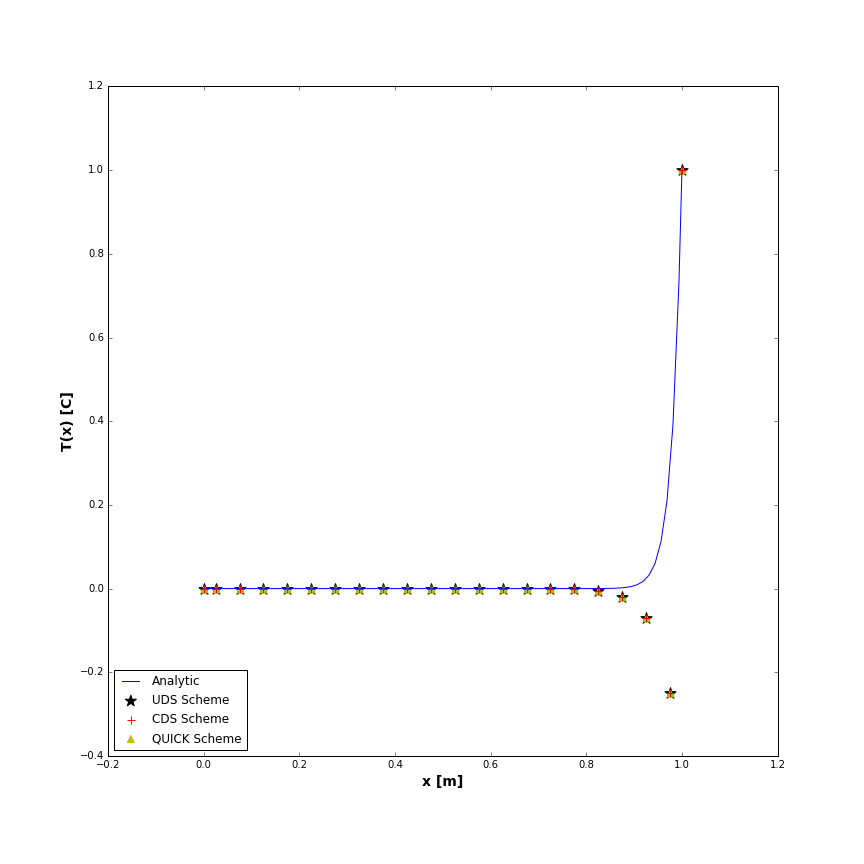


Figure : Problem 1, 20 control volumes for all schemes

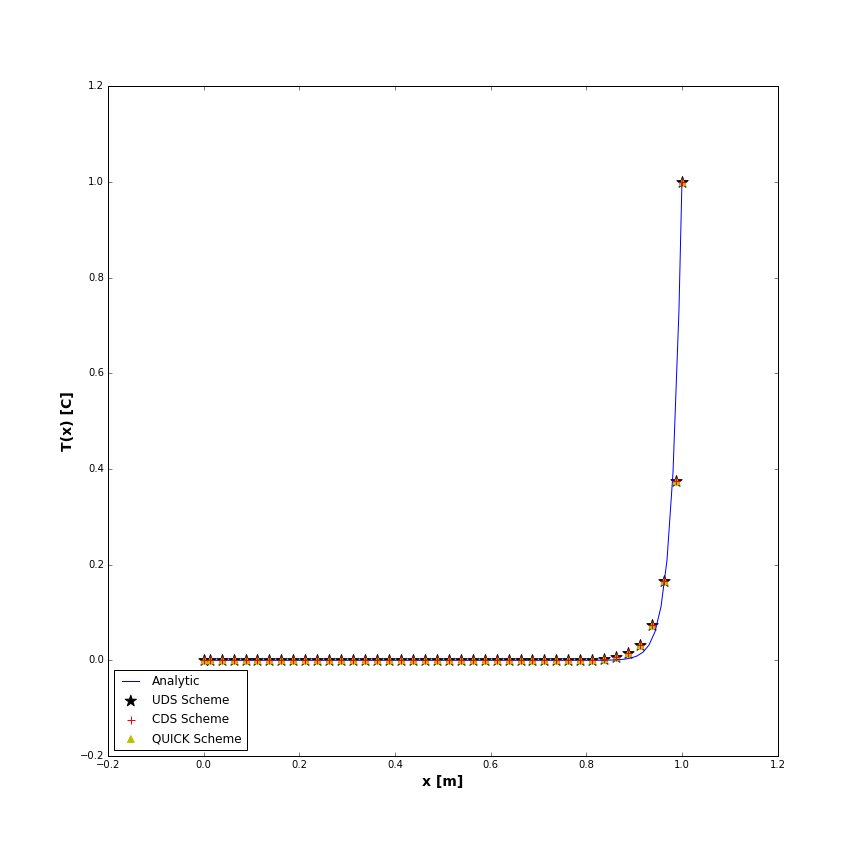


Figure : Problem 1, 40 control volumes for all schemes

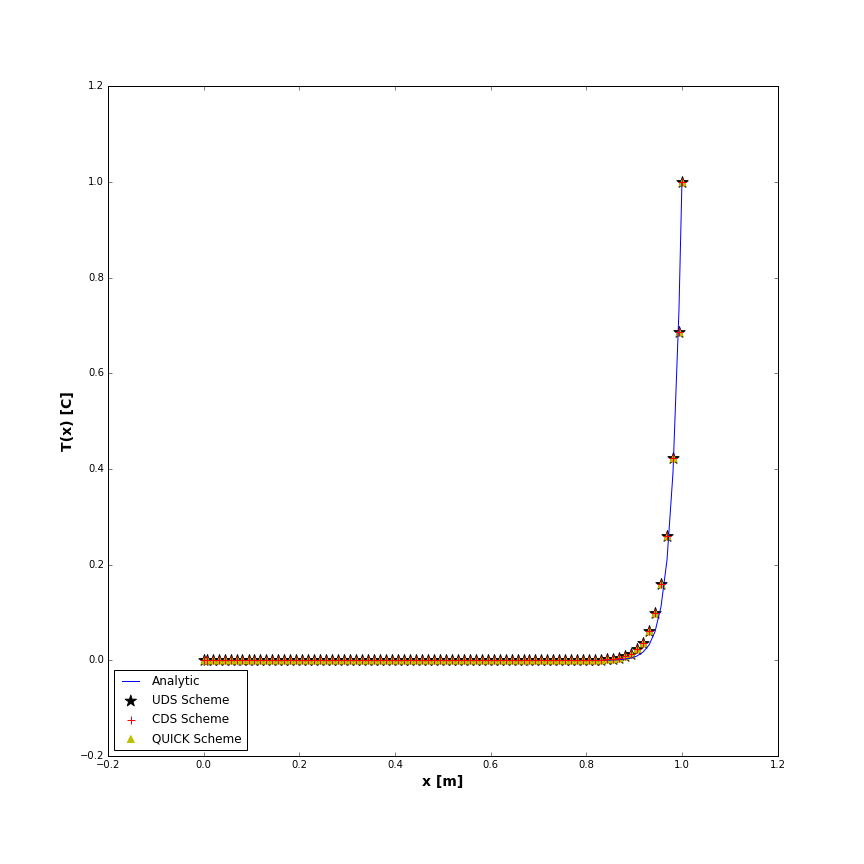


Figure : Problem 1, 80 control volumes for all schemes

# Problem 2

The second problem was a combined convection/diffusion problem with convection around the duct. Table 4 outlines the properties of the problem and the geometry is the same as problem 1. The duct has a prescribed temperature on the left end, but not on the right, resulting in two different boundary conditions. The cross section of the duct was specified and was set in makgrd.f.

Table : Problem 2 properties



## Code Implementation

The problem was implemented by setting boundary conditions in bndct.f and weight.f, and testing each higher order correction scheme. The direction of flow was also tested by reversing the direction of the flow field and switching the boundary conditions. Table \_\_ shows the discretization used for each correction scheme, and Table \_\_ shows the boundary conditions for the beginning and end nodes for both the forward and reversed flow directions.

The sign of the right boundary condition was also tested. Changing the sign of changes which temperature the east face of IE is assigned to. When is -1, the east face temperature is equal to the East temperature on the IE node. When is +1, the east face temperature is equal to the center node temperature:

However, regardless of the sign of the end boundary , the Neumann boundary condition on the end node sets the IE+1 temperature regardless of the value. Since the Neumann condition allows zero heat flux through the IE+1 node, it implies that the IE and IE+1 temperatures had to be the same, overwriting the effect of . The physically realistic value of is -1, the value that sets the east face temperature to the east temperature of the control volume. This value reflects the up-winding scheme being implemented, and carries the correct temperature through the domain. If the flow was reversed, the value should be reversed as well, to reflect the up-winding scheme.

One time step was used for each trial, and was implemented by creating a very large delta t, and limiting the time loop to one iteration. This removed any transient effect from the solution. Dirichlet boundary conditions were used on one end of the duct to specify the temperature, and a Neumann boundary condition was used on the other end to specify zero heat flux through the end node. The end that each was applied to depended on the direction of flow. The Neumann condition was used to allow the solution to predict the end duct temperature. The boundary conditions were set to ensure that the temperature was carried through the domain correctly since the UDS scheme was implemented. The positive IB-1 boundary condition ensured that the west face temperature of the IB node was equal to the west temperature of IB-1. The negative IE boundary condition ensured that the east face temperature of the IE node was equal to the east temperature of the boundary node. This was reversed for the reversed flow field.

Table : Problem 2 discretization



Table : Problem 2 boundary conditions for both forward and reversed flow



## Results

The temperature distribution of all schemes was plotted against the analytic solution for the forward and reverse flow directions. Figure 7 illustrates the forward direction, and Figure 8 illustrates the reversed direction. For both cases, the sign of the end did not influence the distribution. The residuals of each solution converged to the required criteria. UDS converged after one iteration, and QUICK converged faster than CDS.

It was found that the QUICK and CDS schemes traced the analytics solution more accurately than UDS. This was because the accuracy of the higher orders schemes was clearly noticeable since the gradient of the distribution was not as extreme as problem 1. Even though the higher order schemes were more accurate than pure UDS, the coarse mesh of only five control volumes did not accurately reflect all components of the distribution. Figure 9 illustrates the composite for 20 control volumes and confirms that as the mesh is refined, all schemes converged to the analytic solution. The final temperatures were equal because of the Neumann boundary condition, and each scheme came within 2 % of the analytic ending temperature.

When the flow direction and boundary conditions were reversed, the distribution was flipped. The schemes could accurately reflect the change in direction, and the same behavior was exhibited.

Negative coefficients did not occur for any scheme for low control volumes since the Peclet number was low relative to the first problem. This resulted in high stability for each solution, since the Peclet number originally restricted the size of the control volume. The Peclet number also reflects the slope of the distribution, which is why each scheme is able to model the analytic solution well.

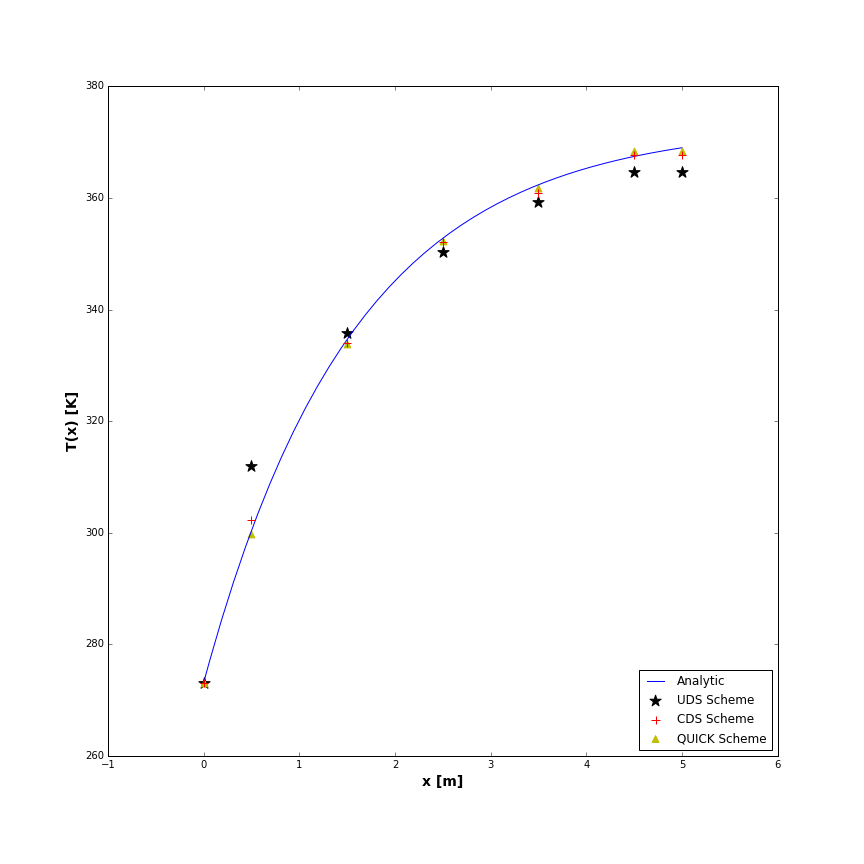


Figure : Problem 2, 5 control volumes, forward flow for all schemes

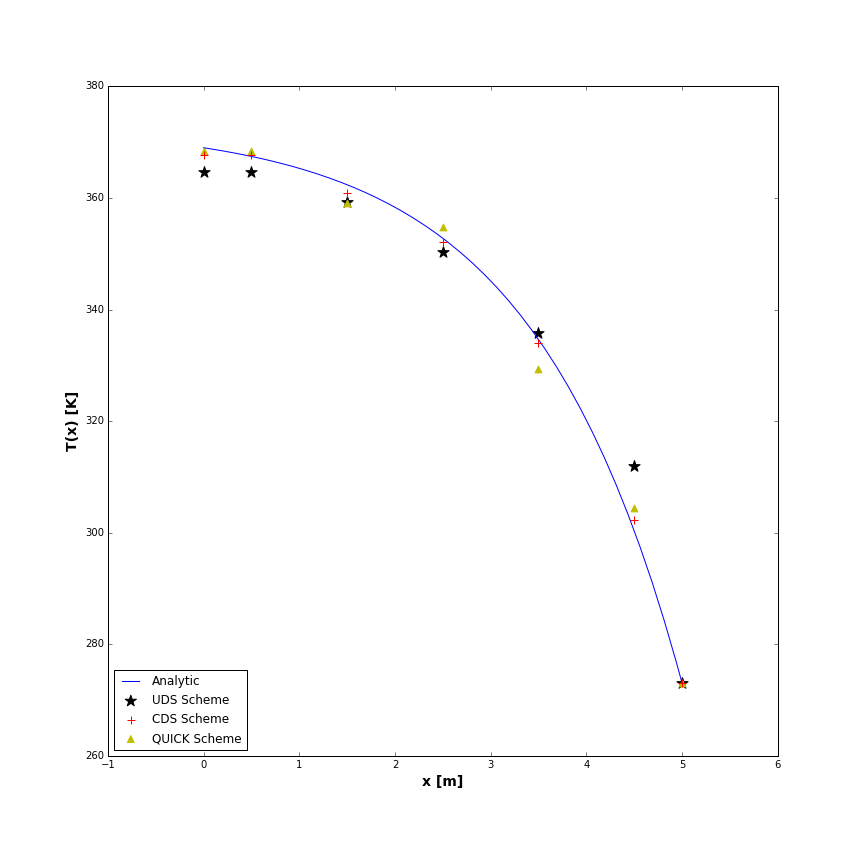


Figure : Problem 2, 5 control volumes, reversed flow for all schemes

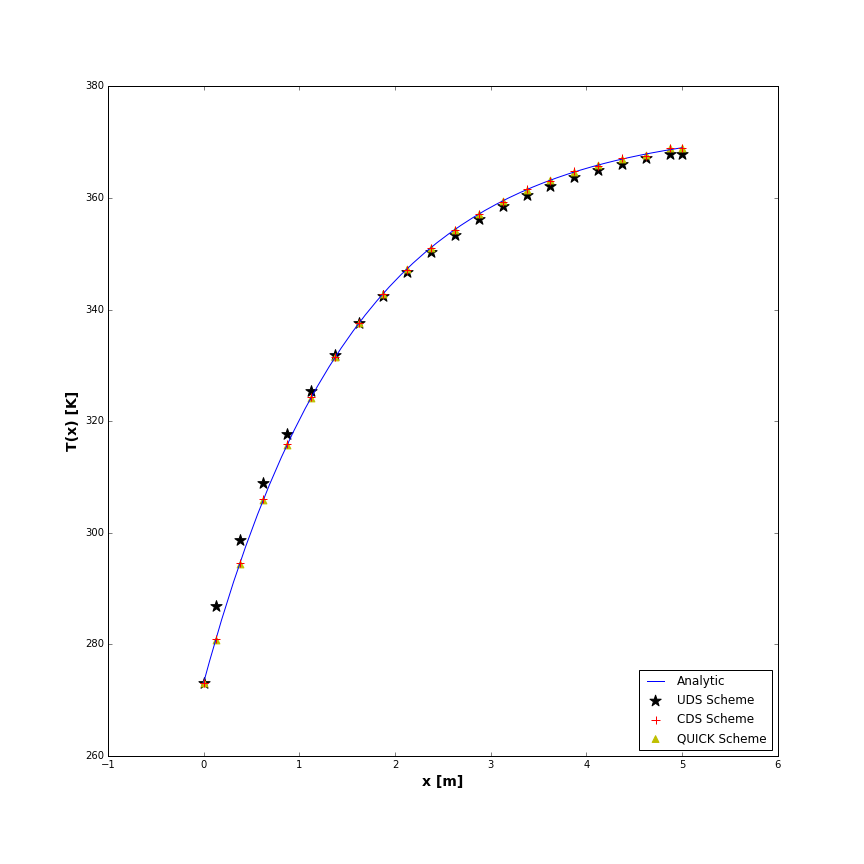


Figure : Problem 2, 20 control volumes, forward flow for all schemes

# Appendix 1

See the following attached pages for the updated subroutines.