NUMERICAL SIMULATION OF TWO-DIMENSIONAL
WOLFHARD-PARKER BURNER

by

Henrik G. Johansson

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE
IN
MECHANICAL ENGINEERING

APPROVED:

[Signatures]

Dr. Balakrishnan Ganeshan, Chair
Dr. Uri Vandenbomer
Dr. Richard J. Roby

July, 1996
Blacksburg, Virginia

Keywords: Combustion, Wolfhard-Parker burner, diffusion flame, flame sheet, soot
LD
SG55
V855
1996
J643
c.2
NUMERICAL SIMULATION OF TWO-DIMENSIONAL WOLFHARD-PARKER BURNER

by
Henrik G. Johansson
Committee Chairman: Dr. Balakrishnan Ganeshan
Department of Mechanical Engineering

(ABSTRACT)

A joint experimental and theoretical project has been initiated at Virginia Tech to study the effects of dual-mode combustion at high pressures for a two-dimensional Wolfhard-Parker burner. This thesis is the first stage of the theoretical part of the project, and contains a numerical study of laminar coflow diffusion flames stabilized on a confined Wolfhard-Parker burner.

A global finite difference method is used where the nonlinear equations written on a stream function-vorticity formulation are solved with a flame sheet approach. The pseudotransient, approximative factorization method is utilized to solve the coupled system of equations. Adaptive gridding, numerical evaluation of Jacobians and iterations within time step are implemented for computational efficiency.

Numerical results have been obtained for different fuels under different conditions. Comparison with measured data by Smyth et al. (1985) for a buoyancy dominated methane-air flame is made. The location of the flame front is accurately predicted. The temperature is overpredicted in the fuel rich zone since pyrolysis and radiation effects have not been accounted for in the numerical model. Good agreement is observed for major species and velocities. As expected, large velocity increase and horizontal inflow of nitrogen and combustion products associated with buoyancy occur in the lower region of the flame.
Acknowledgments

First of all, I thank Dr. Balakrishnan Ganeshan for allowing me to work with him on this project. I thank him for his continuos support, understanding and friendship. Any good achievement in this project would not have been possible without his internationally recognized knowledge of combustion and numerics. He has been an outstanding advisor all along the way, and I am truly grateful towards him. I thank Dr. Uri Vandsburger for initially introducing me into the field of combustion. His support, encouragement and advice has been invaluable to this project. I also thank Dr. Rick Roby for serving on my committee.

I thank all of the graduate students in the Reacting Flows Laboratory at Virginia Tech for their assistance, ideas and friendship. It has been very stimulating to work and share projects together. I also thank my fellow graduate student Rollin Roos for his deep interest in this project. His thoughts, advice and insight have enhanced my work.

I thank my friends here at Virginia Tech, who all together made Blacksburg a nice place to live in. I especially thank my special friend Mona Moret.

Finally I would like to thank my entire family who always was present in my mind, though far away in north Europe (thanks Bell for inventing and improving the telephone). I thank my parents Margareta and Willy for their love and support. I thank my brothers, Erik and Kristian, for all the good times we have shared growing up together and will continue to share.
Table of Contents

Abstract ............................................................................................................ ii
Acknowledgments .......................................................................................... iii
Table of Contents ............................................................................................ iv
List of Figures ................................................................................................ vi
List of Tables ................................................................................................... viii
1. Introduction ................................................................................................ 1
   1.1 Previous work on dual-mode combustion .............................................. 4
   1.2 Motivation .............................................................................................. 7
   1.3 Computational considerations in flame modeling ................................. 10
   1.4 Scope of thesis ...................................................................................... 11
2. Formulation .................................................................................................. 12
   2.1 Governing equations ............................................................................ 13
   2.2 Boundary conditions ............................................................................ 16
   2.3 Stream function-vorticity formulation ............................................... 18
   2.4 Implementation of boundary conditions for $\psi$ and $\omega$ .................. 19
3. Flame sheet model ...................................................................................... 25
   3.1 Derivation of flame sheet representation of the conservation equations .. 25
   3.2 Boundary conditions for flame sheet computations ........................... 31
   3.3 An analytic solution for diffusion flames stabilized on a Wolfhard-Parker burner ................................................................. 34
4. Method of solution for Wolfhard-Parker burner .......................................... 40
   4.1 Pseudotransient method ...................................................................... 40
   4.2 Approximate factorization .................................................................... 43
   4.3 Discretization ....................................................................................... 46
4.4 Boundary conditions on a pseudotransient formulation..........................51
4.5 Iterations within time step...........................................................................51
4.6 Freezing and numerical evaluation of Jacobians..........................................52
4.7 Adaptive gridding.........................................................................................53
5. Numerical code...............................................................................................56
6. Results and discussion......................................................................................61
   6.1 Computations with methane as fuel............................................................61
       6.1.1 Mesh and adaptive gridding...............................................................63
       6.1.2 Comparison with experiment.............................................................65
       6.1.3 Buoyancy effects and comparison with analytic solution.....................76
   6.2 Computation with propane as fuel..............................................................79
       6.2.1 Comparison with analytic solution.....................................................81
       6.2.2 Comparison with computation..........................................................82
7. Conclusions and recommendations for future work...........................................87
References............................................................................................................93
Appendix...............................................................................................................99
Vita......................................................................................................................103
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Illustration of dual-mode combustion</td>
<td>3</td>
</tr>
<tr>
<td>1.2</td>
<td>A schematic of counter-flow flame</td>
<td>4</td>
</tr>
<tr>
<td>1.3</td>
<td>Soot volume fraction</td>
<td>5</td>
</tr>
<tr>
<td>1.4</td>
<td>Soot particle number density</td>
<td>6</td>
</tr>
<tr>
<td>1.5</td>
<td>Soot particle diameter</td>
<td>6</td>
</tr>
<tr>
<td>1.6</td>
<td>Schematic diagram of Wolfhard-Parker burner. The walls that enclose the burner are not shown</td>
<td>9</td>
</tr>
<tr>
<td>2.1</td>
<td>Schematic sketch of the computational domain used to simulate the Wolfhard-Parker burner</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>Schematic of smoothing of vertical velocity at the inflow</td>
<td>20</td>
</tr>
<tr>
<td>2.3</td>
<td>Illustration of discretization about a grid point ((j,k)) on the wall</td>
<td>22</td>
</tr>
<tr>
<td>3.1</td>
<td>Mass fractions of various species and temperature as functions of the mixture fraction</td>
<td>31</td>
</tr>
<tr>
<td>3.2</td>
<td>Flame shapes for the Wolfhard-Parker burner ((c = 0.2))</td>
<td>38</td>
</tr>
<tr>
<td>4.1</td>
<td>Illustration of splitting technique</td>
<td>43</td>
</tr>
<tr>
<td>4.2</td>
<td>Illustration of adaptive gridding for arbitrary variable profile</td>
<td>54</td>
</tr>
<tr>
<td>5.1</td>
<td>Definition of class (ListNode)</td>
<td>57</td>
</tr>
<tr>
<td>5.2</td>
<td>Elements of array (pos)</td>
<td>57</td>
</tr>
<tr>
<td>5.3</td>
<td>Definition of class (List). Only a small portion of the member functions is shown</td>
<td>58</td>
</tr>
<tr>
<td>5.4</td>
<td>Schematic illustration of a simple (4 \times 3) mesh</td>
<td>59</td>
</tr>
<tr>
<td>6.1</td>
<td>Mesh used in the methane computation. Solid line indicates the location of the computed flame front</td>
<td>64</td>
</tr>
<tr>
<td>6.2</td>
<td>Temperatures at a height of 0.9 cm above the inflow</td>
<td>66</td>
</tr>
<tr>
<td>6.3</td>
<td>Mole fractions of methane and oxygen at a height of 0.9 cm above the inflow</td>
<td>68</td>
</tr>
<tr>
<td>6.4</td>
<td>Mole fractions of carbon dioxide and water at a height of 0.9 cm above the inflow</td>
<td>68</td>
</tr>
<tr>
<td>6.5</td>
<td>Vertical velocities at a height of 0.9 cm above the inflow</td>
<td>69</td>
</tr>
</tbody>
</table>
6.6 Horizontal velocities at a height of 0.9 cm above the inflow.................................70
6.7 Mixture fraction field.................................................................72
6.8 Stream function field...............................................................72
6.9 Vorticity field...............................................................73
6.10 Temperature field..............................................................73
6.11 Vertical velocity field...........................................................74
6.12 Horizontal velocity field.......................................................74
6.13 Streamlines in the Wolfhard-Parker burner for methane as fuel.........................75
6.14 Comparison of vertical velocity at the centerline for buoyant and non
    buoyant flames...................................................................77
6.15 Comparison of flame shapes for methane as fuel.............................................78
6.16 Comparison of numerical and analytical flame shapes for propane as fuel..........81
6.17 Mixture fraction field.............................................................83
6.18 Stream function field.............................................................83
6.19 Vorticity field.....................................................................84
6.20 Temperature field.................................................................84
6.21 Vertical velocity field.............................................................85
6.22 Horizontal velocity field.........................................................85
6.23 Streamlines in the Wolfhard-Parker burner for propane as fuel.......................86
7.1 Illustration of solution procedure for Wolfhard-Parker burner.........................90
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Physical and chemical parameters used in the methane computations.</td>
<td>62</td>
</tr>
<tr>
<td>2. Location of peak experimental and computational temperatures at three different axial locations.</td>
<td>67</td>
</tr>
<tr>
<td>3. Computed vertical velocity at the centerline and computed peak horizontal velocity as a function of height above the burner.</td>
<td>71</td>
</tr>
<tr>
<td>4. Physical and chemical parameters used in the propane computation.</td>
<td>80</td>
</tr>
<tr>
<td>5. Computed vertical velocity at the centerline and computed maximum inward horizontal velocity as a function of height above the burner.</td>
<td>82</td>
</tr>
</tbody>
</table>
1. Introduction

Soot emissions from power plants, automobiles, airborne propulsion devices, marine and land based power generators, constitute an undesired pollutant for various reasons. Soot can cause component damage due to heat transfer effects. For example, the life time of the blades in gas turbines is severely affected by the presence of soot in the gas stream. Furthermore, soot can result in a reduction of efficiency for combustion devices. Soot is also a known mutagen (Lahaye and Prado, 1983) and can absorb carcinogenic materials.

While reduction of pollutant emission from automobiles is based on 3-way catalysts, this method has not yet been successful in the presence of soot. The post-combustion treatment of pollutants is, however, not a viable possibility in propulsion devices based on the thrust developed by the stream of a hot gas. Therefore, the control of soot emissions level (as well as other pollutants) has to be done in the combustion zone itself. This implies that the processes of soot formation and destruction in liquid fuel based combustors must be fully understood. The issue of soot emissions will gain further urgency as high energy density fuels are considered; materials which generally have a strong sooting tendency.

Soot particles are carbonaceous particles formed from gas phase processes. The history of the fuel components that eventually form soot is a most complex (Glassman, 1987). Many sequential and overlapping steps take place in the overall soot formation system (Haynes and Wagner, 1981). The initial step, called nucleation, occurs when the fuel molecules dissociate into smaller fragments upon heating, but subsequently polymerize into much larger hydrocarbons. The next step, called the growth and agglomeration stage, is where the bulk of the soot is formed. Gas phase hydrocarbons condense onto soot precursors, forming fairly spherical particles. As the rate of surface growth slows, these particles tend to agglomerate into long chainlike structures. In the final oxidation step these soot particles may be burned. Whether or not soot is released from a flame depends
on the rates of all three of these steps. Therefore, it is important to study the history of the soot particles within the flame in order to understand the overall sooting process.

In premixed flames, soot formation and oxidative attack on soot and soot precursors occur simultaneously. Increases in flame temperature cause a decrease in soot formation, since the oxidative reactions become faster than pyrolytic reactions at higher temperatures, and oxidation dominates. As the fuel pyrolyzes, it eventually forms C_2H_2 (acetylene), which grows by adding H and other C_2H_2 groups until a benzene ring is formed (Glassman, 1988).

In diffusion flames, there is no oxidation of the soot simultaneous with its formation, and any increase in flame temperature causes a corresponding increase in the formation rate of soot. The initial structure of the fuel also becomes very important in determining the rate at which it can form aromatic rings (Miller et al., 1992, Sidebotham et al., 1992). As in premixed flames, the ring forming process is thought to begin with acetylene, which grows by the addition of hydrogen and acetylene to form vinyl radicals, butadienyl radicals, vinyl acetylene and finally phenylacetylene after a series of reactions with acetylene and hydrogen. In the soot inception region, adequate concentrations of these species exist for this mechanism to be valid. Another path to benzene in diffusion flames which is far more dependent on the structure of the original fuel molecules is also possible (Miller et al., 1992).

Much work has been done trying to determine the precise mechanism by which fuel molecules are transformed into soot particles. There is still much debate over the particular steps to forming the soot nucleus, a particle about 5 nm in diameter containing both carbon and hydrogen. The main difficulty seems to be the limited amount of time available for the formation of these particles (about 1 ms).

Most previous studies of soot formation have focused on either laminar premixed flames or laminar diffusion flames and have provided the basis of the current knowledge on the fundamental processes of soot formation. Combustion in liquid-fueled systems, diesels and gas turbines, does not, however, occur as a premixed or pure diffusion flame.
The combustion is rather a combination of the two, termed here dual-mode combustion, in contrast with a pure diffusion flame termed as single-mode. The source of these characteristics is the distillation curve of the fuel, which is not neat. The dual-mode combustion process is illustrated in Figure 1.1. Upon injection into the compressed heated air, the lighter distillates in the fuel vaporize and mix rapidly with the air. This lean mixture burns as a premixed flame. The balance of the fuel burns in one of the recognized spray models, all of which are diffusion limited burning. This burning, however, occurs in an environment of vitiated air containing products of the previous combustion. In order to correctly predict soot formation from liquid-fueled combustion devices, the effect of a dual-mode combustion environment needs to be clarified.

Figure 1.1. Illustration of dual-mode combustion.
1.1 Previous work on dual-mode combustion

The effects of dual-mode combustion on soot formation have been studied for a one-dimensional burner by Afiel et al. (1995). The burner configuration in their investigation was of cylindrical counterflow type, first used by Tsuji et al. (1966). A schematic of the configuration may be seen in Figure 1.2. Fuel was injected through a porous ceramic cylinder while the oxidizer flowed in the opposite direction and over the cylinder. A diffusion flame established itself around the cylinder as shown.

![Diagram of counter-flow flame](image)

**Figure 1.2.** A schematic of counter-flow flame.

The advantages with this configuration are the one-dimensional measurement zone along the stagnation streamline, simplifying the data acquisition, and no uncontrolled entrainment of oxidizer into the flame. The main disadvantage is the narrow reaction zone (between the flame front and the stagnation line) of approximately 5 mm requiring good spatial resolution and high accuracy of experimental techniques to achieve reliable results.
Results from Aftel et al. for ethene, C$_2$H$_4$, as diffusion flame fuel are shown in Figure 1.3-1.5, where different soot characteristics are plotted versus the particle time from the flame front when the soot particles are being convected downstreams for single-mode conditions and two different dual-mode conditions: 0.3 and 0.5 which correspond to the equivalence ratio of the premixed H$_2$/O$_2$ flame. These results clearly show a delay in the onset of soot on the fuel side of the diffusion flame in the dual-mode flames compared with pure diffusion flames. Also, the soot in the dual-mode flames is characterized by larger numbers of smaller particles. Supported by numerical analysis, the observations made are explained mostly in terms of H$_2$O, which is a product of combustion for both the premixed and diffusion controlled burning. The dissociation at the diffusion flame front yields OH which attacks both soot precursors as well as neutralizes the growth potential of young soot particles. The concentration of growth species like acetylene and propargyl were also seen to be reduced.

Figure 1.3. Soot volume fraction.
Figure 1.4. Soot particle number density.

Figure 1.5. Soot particle diameter.
Studies of soot formation under single-mode conditions have been undertaken at high pressure (Flower and Bowman, 1984). These studies showed an increase in soot production. Though it is anticipated that the increased pressure will also increase oxidation by OH, the exact balance is not clear.

A joint experimental and theoretical project has been initiated at Virginia Tech to parametrically study the effect of dual-mode combustion on sooting characteristics of higher hydrocarbons at high pressures. The objective is to develop a predictive model for soot formation in dual-mode combustion including gas phase chemistry and soot chemistry. This thesis is the first step in the theoretical part of the project, and contains a numerical study of coflow laminar diffusion flames at atmospheric pressure.

1.2 Motivation

Turbulent reacting flows contain a wide range of time and length scales which cannot be resolved completely (Libby and Williams, 1993). For this reason, most of the computational studies involving detailed chemistry have focused on essentially one-dimensional configurations, i.e., freely propagating or burner stabilized premixed flames and counterflow premixed or diffusion flames (Margolis, 1978, Smooke, 1982, Smooke et al., 1985, Dixon-Lewis et al., 1982). Although these one-dimensional studies have provided invaluable insight into the physics of reacting flows, modeling practical combustion systems requires knowledge of the fundamental processes involved in two and three-dimensional flows. With the advent of large memory parallel and vector computers, and improving numerical techniques, multi-dimensional simulation of laminar flows with realistic chemistry models have become more realistic in recent years. Still, two-dimensional models typically requires one hundred CPU hours on the most advanced supercomputers, and three-dimensional are not yet computationally feasible (Smooke, 1991).
A two-dimensional laminar coflow axisymmetric diffusion flame in which a cylindrical fuel stream is surrounded by a coflowing oxidizer jet is the configuration of many existing combustion devices. Infinite rate chemistry computations for this configuration have been performed by Mitchell et al. (1980), Boedeker and Dobbs (1986) and Villaseñor (1994). Miller and Kee (1977) have performed finite rate chemistry computations for the same configuration for a hydrogen-air flame and Smooke et al. (1989) for methane-air flame. The axial symmetry of the concentric configuration makes experimental analyses challenging, particularly when optical diagnostic tools or thermocouples are used, thus, it is difficult to verify numerical results with experiments (Glassman, 1987).

A two-dimensional Wolfhard-Parker burner, schematically shown in Figure 1.1, provides a more convenient configuration for experimental study. Soot formation experiments have been performed for this burner type by Kent et al. (1981) and Smyth et al. (1985), and are explained in detailed in chapter 6. The minimal variations along the slot enable easy implementation of optical and intrusive techniques to perform a parametric study. Furthermore, for the same reasons the experiments can be modeled with a set of two-dimensional conservation equations. Infinite rate chemistry computations for the Wolfhard-Parker burner have been performed by Liu et al. (1993) for propane as fuel.
Figure 1.6. Schematic diagram of Wolfhard-Parker burner. The walls that enclose the burner are not shown.

In order to obtain steady laminar conditions for the experiments a stabilizing screen must be used. In this thesis its effects have been neglected, but it will in future studies be modeled. The original Wolfhard-Parker has been modified by additional oxidizer slots for stability reasons and to eliminate end effects, similar to the design of Kent and Wagner (1982).
1.3 Computational considerations in flame modeling

There are several important computational issues that need to be considered in the modeling of flames. The issues listed below will be addressed in the appropriate sections of this thesis:

- The total number of differential equations which need to be considered is very large. For example, for a methane-air system it is not uncommon to find a chemical kinetic mechanism with 40 distinct chemical species. This number increases to approximately 300 when the fuel under consideration is propane.

- The conservation equations are highly non-linear through the exponential dependence of the species source terms on the temperature. The convective terms are also non-linear, but this non-linearity is not as severe as for the species source terms.

- In a diffusion flame most of the chemical activity occurs in a narrow region located near the peak temperature. Unlike the premixed flames in which the burning rate is controlled by chemical reactions, the distinctive characteristic of a diffusion flame is that the burning rate is controlled by the rate at which the fuel and the oxidizer are brought together in stoichiometric proportions. Hence, for diffusion flames there exists a strong coupling (interaction) between the thermochemistry and the fluid dynamics, which makes numerical modeling more challenging for these flames (Smooke et al., 1989).

- There is a large difference between the scale of the burner and the flame thickness. The Wolfhard-Parker burner typically has dimensions of order 0.1 m, the flame thickness is normally on the order of $10^{-4}$ m. Such wide range of length scales imposes severe grid requirements in order to achieve high accuracy. Hence, to properly resolve
the reaction zone while maintaining computational efficiency, a highly non-uniform mesh is needed.

1.4 Scope of thesis

The goals of this study are:

1. Obtain numerical results for the Wolfhard-Parker burner with a simplified non radiative combustion model. The objective for the model to be developed is to be able to numerically predict the mass fractions of the species under investigation, the temperature and the flow field everywhere in the burner.

2. Develop an efficient computer code that can easily be modified for finite rate chemistry computations.

This thesis is organized in the following way. In chapter two the theoretical problem is formulated using a stream function-vorticity formulation. The combustion model used is derived and discussed in chapter three together with an analytic solution of diffusion flames stabilized on a Wolfhard-Parker burner. The solution method is developed in chapter four, and the numerical code is discussed in chapter five. Numerical and analytical results for methane-air and propane-air flames are presented and discussed in chapter six. Conclusions and recommendations for future work are included in chapter seven.
2. Formulation

The Wolfhard-Parker burner is a confined, rectangular diffusion flame, where fuel discharges through the central slot into a coflowing oxidizer stream. The slots through which the fuel and the oxidizer flow are rectangular, with $x_1$ and $x_2$ denoting the distance from the centerline to the central and outer slot walls, respectively. The distance from the centerline to the wall is denoted $x_w$. The fuel and the oxidizers first meet in stoichiometric proportions at the outlet of the central slot wall and a flame is stabilized just above this zone.

![Schematic sketch of the computational domain used to simulate the Wolfhard-Parker burner.](image)

Figure 2.1. Schematic sketch of the computational domain used to simulate the Wolfhard-Parker burner.
The computational domain (shown in Figure 2.1) is bounded by the centerline of the burner and the wall in the horizontal direction, and by the inflow and outflow boundaries in the vertical direction.

The formulation of the problem assumes the following

a) The flow is laminar, steady and two-dimensional.
b) Uniform inflow conditions for each of the slots.
c) Radiative heat transfer is negligible.
d) The diffusion of heat due to concentration gradients (Dufour effect) is negligible.
e) Mass diffusion due to temperature gradients (Soret effect) is negligible.
f) Monatomic gas mixture, i.e. bulk viscosity is zero.
g) Viscous dissipation is negligibly small.

2.1 Governing equations

The flame structure with approximations (a)-(g) can be modeled with a set of two-dimensional equations. With \( x \) and \( y \) denoting the horizontal and vertical coordinates, respectively, the governing equations can be written as

\[
\frac{\partial}{\partial x} (\rho u) + \frac{\partial}{\partial y} (\rho v) = 0, \tag{2.1}
\]

\( x \)-momentum:

\[
\rho \left( \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial y} \left( \mu \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial u}{\partial x} \right) - \frac{2}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) - \frac{2}{3} \frac{\partial}{\partial x} \left( \mu \frac{\partial v}{\partial y} \right), \tag{2.2}
\]
\( \rho \left( u \frac{\partial u}{\partial x} + v \frac{\partial v}{\partial y} \right) = -\frac{\partial p}{\partial y} + \frac{\partial}{\partial x} (\mu \frac{\partial u}{\partial y}) + \frac{\partial}{\partial y} (\mu \frac{\partial v}{\partial x}) - \frac{2}{3} \frac{\partial}{\partial y} (\mu \frac{\partial v}{\partial y}) - \frac{2}{3} \frac{\partial}{\partial x} (\mu \frac{\partial u}{\partial x}) - g \rho , \)  

\( (2.3) \)

**Energy:**

\[ \rho c_p \left( u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} \right) = \frac{\partial}{\partial x} (\lambda \frac{\partial T}{\partial x}) + \frac{\partial}{\partial y} (\lambda \frac{\partial T}{\partial y}) - \left\{ \frac{\partial}{\partial x} (\rho T \sum_{i=1}^{N} c_{p_i} Y_i V_i) + \frac{\partial}{\partial y} (\rho T \sum_{i=1}^{N} c_{p_i} Y_i V_i) \right\} - \sum_{i=1}^{N} \dot{\omega}_i \Delta h^*_{i} , \]

\( (2.4) \)

**Species:**

\[ \rho \left( u \frac{\partial Y_i}{\partial x} + v \frac{\partial Y_i}{\partial y} \right) + \frac{\partial}{\partial x} (\rho Y_i V_i) + \frac{\partial}{\partial y} (\rho Y_i V_i) - \dot{\omega}_i = 0 , \quad i = 1, 2, ..., N. \]

\( (2.5) \)

The system is closed by the equation of state for ideal gas mixtures,

\[ \rho R T = \rho \bar{W} . \]

\( (2.6) \)

In Eqs. (2.1)-(2.6), \( T \) is the temperature, \( Y_i \) is the mass fraction of species \( i \), \( p \) is the pressure, \( u \) and \( v \) are the velocities of the fluid mixture in the horizontal and vertical directions, respectively, \( \rho \) is the mixture density, \( \bar{W} \) is the average molecular weight of the mixture, \( R \) is the universal gas constant, \( \lambda \) is the thermal conductivity, \( c_p \) and \( c_{p_i} \) are the specific heat at constant pressure for the mixture and species \( i \), respectively, \( N \) is the total number of species, \( \dot{\omega}_i \) is the rate of production for species \( i \), \( \Delta h^*_{i} \) is the heat of formation for species \( i \) at a reference temperature per unit mass, \( g \) is the acceleration due to gravity and \( \mu \) is the viscosity of the mixture.

An order of magnitude estimate of the terms in Eq. (2.3) indicates that all pressure variations in the flame are much smaller than the atmospheric pressure, \( p_{\text{atm}} \). Equation (2.6) therefore implies that the density for a given mixture composition is a function of temperature only.

With \( X_i \) denoting the mole fraction of species \( i \), the \( x \) and \( y \) components of the mass diffusion velocity vector, \( V_i \), can be determined from the relationships,
\[ V_i = -\frac{1}{X_i} D_i \frac{\partial X_i}{\partial x}, \quad i = 1, 2, ..., N, \quad (2.7) \]

\[ V_i = -\frac{1}{X_i} D_i \frac{\partial X_i}{\partial y}, \quad i = 1, 2, ..., N, \quad (2.8) \]

where \( D_i \), the "pseudo-diffusion" coefficient of species \( i \), is related to the binary diffusion coefficients of species pair \( i, j \), \( D_{ij} \) by (Curtiss and Hirschfelder, 1949),

\[ D_i = \frac{(1-Y_i)}{\sum_{j=1}^{N} X_i / D_{ij}}. \quad (2.9) \]

The rate of production of species \( i \) is given by the phenomenological expression of chemical kinetics

\[ \dot{w}_i = W_i \sum_{j=1}^{m} (v_{ji}^p - v_{ji}^n) \left[ k_{f,j} \prod_{n=1}^{N} \left( \frac{\rho Y_n}{W_n} \right)^{\nu_n^p} \right] - k_{b,j} \prod_{n=1}^{N} \left( \frac{\rho Y_n}{W_n} \right)^{\nu_n^n} \]. \quad (2.10) \]

where \( m \) is the total number of simultaneous elementary reactions in the chemical-kinetic scheme

\[ \sum_{i=1}^{N} v_{iM_i} \rightarrow \sum_{i=1}^{N} v_{iM_i}, \quad k = 1, 2, ..., m, \quad (2.11) \]

with \( M_i \) representing the chemical symbol for species \( i \). In Eq. (2.10) \( k_{f,j} \) and \( k_{b,j} \) denote the specific rate constants for the forward and backward reactions, respectively. The forward specific rate constant is usually evaluated by the modified Arrhenius law,

\[ k_{f,j} = A_j T^{\alpha_j} \exp(-E_j / RT), \quad (2.12) \]

where \( A_j \) is the pre-exponential factor, \( \alpha_j \) is the temperature exponent and \( E_j \) is the activation energy for the elementary reaction \( j \). The backward specific rate constant, \( k_{b,j} \), can be obtained from

\[ k_{b,j} = \frac{k_{f,j}}{K_j^c}, \quad (2.13) \]
where $K_j^c$ is the concentration equilibrium constant at temperature $T$. The thermodynamic and transport properties for the governing equations can be obtained from the JANAF tables (JANAF, 1971).

2.2 Boundary conditions

The conservation equations are elliptic, and hence to obtain a well-posed problem one needs to specify boundary conditions at all boundaries of the computational domain. Symmetry boundary conditions are imposed at the centerline, since the burner is symmetric. At the wall no-slip and adiabatic wall boundary conditions are imposed. For the outflow boundary the gradients in the vertical direction are set to zero. If subscript $F$ refers to the fuel stream, and subscripts $OX_1$ and $OX_2$ refer to the oxidizer streams, respectively, the boundary conditions can be written as

Centerline ($x = 0$):

$$
\frac{\partial \rho}{\partial x} = u = \frac{\partial v}{\partial x} = \frac{\partial Y_i}{\partial x} = \frac{\partial T}{\partial x} = 0, \quad i = 1, 2, \ldots, N,
$$

(2.14)

Outflow ($y \to \infty$):

$$
u \frac{\partial v}{\partial y} = \frac{\partial Y_i}{\partial y} = \frac{\partial T}{\partial y} = 0, \quad i = 1, 2, \ldots, N,
$$

(2.15)

Inflow ($y = 0$):

$0 < x < x_i$:

$$ho = \rho_F, \quad (2.16a)
$$

$$u = 0, \quad (2.16b)
$$

$$v = v_F, \quad (2.16c)
$$

$$Y_i = Y_i^F, \quad i = 1, 2, \ldots, N, \quad (2.16d)$$
\[ T = T_F, \] (2.16e)

\[ x_1 < x < x_2: \]

\[ \rho = \rho_{ox}, \] (2.17a)

\[ u = 0, \] (2.17b)

\[ v = v_{ox}, \] (2.17c)

\[ Y_i = Y_{i,ox}, \quad i = 1, 2, \ldots, N, \] (2.17d)

\[ T = T_{ox}, \] (2.17e)

\[ x_2 < x < x_w: \]

\[ \rho = \rho_{ox_2}, \] (2.18a)

\[ u = 0, \] (2.18b)

\[ v = v_{ox_2}, \] (2.18c)

\[ Y_i = Y_{i,ox_2}, \quad i = 1, 2, \ldots, N, \] (2.18d)

\[ T = T_{ox_2}, \] (2.18e)

Wall \((x = x_w):\)

\[ u = 0, \] (2.19a)

\[ v = 0, \] (2.19b)

\[ \frac{\partial Y_i}{\partial x} = 0, \quad i = 1, 2, \ldots, N, \] (2.19c)

\[ \frac{\partial T}{\partial x} = 0. \] (2.19d)

Boundary condition (2.19c) follows from the fact that species cannot diffuse into the solid wall.
2.3 Stream function-vorticity formulation

The conservation equations for the Wolfhard-Parker burner, can be rewritten in a stream function-vorticity formulation, which eliminates the pressure and thus reduces the number of dependent variables by one (White, 1986). The vorticity is defined as

\[ \omega = \nabla \times \mathbf{v}, \]

(2.20)

where \( \mathbf{v} \) is the velocity vector. In a two-dimensional coordinate system, only the z-component of the vorticity vector \( \omega \), denoted by \( \omega \),

\[ \omega = (0,0,-\omega), \]

(2.21)

is non-zero,

\[ \omega = \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}. \]

(2.22)

With this definition, \( \omega \) is a measure of the clockwise rotation of the flow in the burner. The stream function \( \psi \) is defined in such a way that the \textit{steady-state} continuity equation is always satisfied, i.e.,

\[ \rho u = -\frac{\partial \psi}{\partial y}, \]

(2.23)

\[ \rho v = \frac{\partial \psi}{\partial x}. \]

(2.24)

With the definitions in Eqs. (2.22)-(2.24), the governing set of equations take the form

\[ \text{Stream function:} \]

\[ -\frac{\partial}{\partial x} \left( \frac{1}{\rho} \frac{\partial \psi}{\partial x} \right) - \frac{\partial}{\partial y} \left( \frac{1}{\rho} \frac{\partial \psi}{\partial y} \right) - \omega = 0, \]

(2.25)
Vorticity:

\[ -\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} \omega \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} \omega \right) - \frac{\partial}{\partial x} \left( \mu \frac{\partial \omega}{\partial x} \right) - \frac{\partial}{\partial y} \left( \mu \frac{\partial \omega}{\partial y} \right) = \]

\[ \frac{\partial p}{\partial x} \frac{\partial}{\partial y} \left( \frac{u^2 + v^2}{2} \right) - \frac{\partial p}{\partial y} \frac{\partial}{\partial x} \left( \frac{u^2 + v^2}{2} \right) + g \frac{\partial p}{\partial x}, \quad (2.26) \]

Energy:

\[ c_p \left\{ -\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} T \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} T \right) \right\} = \frac{\partial}{\partial x} \left( \frac{\lambda}{\partial x} T \right) + \frac{\partial}{\partial y} \left( \frac{\lambda}{\partial y} T \right) \]

\[ -\left\{ \frac{\partial}{\partial x} \left( \rho T \sum_{i=1}^{N} c_{pi} Y_i V_i \right) + \frac{\partial}{\partial y} \left( \rho T \sum_{i=1}^{N} c_{pi} Y_i V_i \right) \right\} - \sum_{i=1}^{N} \dot{w}_i \Delta h^*_i, \quad (2.27) \]

Species:

\[ -\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} Y_i \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} Y_i \right) + \frac{\partial}{\partial x} \left( \rho Y_i V_i \right) + \frac{\partial}{\partial y} \left( \rho Y_i V_i \right) - \dot{w}_i = 0, \quad i = 1, 2, ..., N. \quad (2.28) \]

2.4 Implementation of boundary conditions for \( \psi \) and \( \omega \)

The boundary conditions for flow problems are usually formulated in terms of the velocity components. The stream function-vorticity formulation needs additional boundary conditions for \( \psi \) and \( \omega \). Since the solution of both variables are described by second-order differential equations, boundary conditions on all boundaries are necessary. It is well known that these boundary conditions may cause numerical instabilities (Dennis et al., 1979). Therefore, some attention is drawn to this subject.

At the inflow boundary, uniform irrotational flow cannot be assumed when the fuel and oxidizers have different inflow velocities, since the vorticity is discontinuous at the
interface between such boundaries. Moreover, with a no-slip boundary condition the vorticity also becomes discontinuous at the wall. For any numerical algorithm to converge in the presence of discontinuities, smoothing functions have to be used. A fourth order polynomial is therefore used to eliminate the discontinuities introduced by the vorticity. Figure 2.2 illustrates how the incoming velocity profile is smoothed. The width of the smoothing is determined from user specified input.

![Figure 2.2. Schematic of smoothing of vertical velocity at the inflow.](image)

The stream function can now be determined analytically by integration of its definition in Eq. (2.24).

Boundary conditions for the outflow boundary are difficult to specify since the flow field is not known beforehand (Norton et al., 1993). One possibility is to use

$$\frac{\partial \psi}{\partial y} = 0,$$

which according to the definition in Eq. (2.23) corresponds to \( u = 0 \). A more stringent condition is

$$\frac{\partial^2 \psi}{\partial y^2} = 0.$$

(2.30)
This boundary condition which is known as Roaches' condition (Roache, 1972) is consistent with the assumption that the derivative of the horizontal mass-flux vector is equal to zero, i.e.,

$$\frac{\partial (\rho u)}{\partial y} = 0.$$  \hspace{1cm} (2.31)

This condition has been used successfully in incompressible flow computations, but accuracy might be compromised if the density varies in the vertical direction close to the outflow. Therefore, for the present study, Eq. (2.29) is applied as the boundary condition at the outflow boundary. This is a good assumption, especially if the outflow boundary is positioned far away from the regions of high activity. To ensure this, the height of the computational domain has to be much larger than the flame height for overventilated flames, or the location for which the flame attaches the wall in underventilated cases.

A suitable vorticity boundary condition at the outflow is (Baker, 1983)

$$\frac{\partial \omega}{\partial y} = 0.$$  \hspace{1cm} (2.32)

At the wall, an analytic value $\psi_w$ of the stream function can be used. The vorticity boundary condition at a solid wall has been discussed extensively in the literature. For example, see Fletcher (1991). By utilizing the fact that the stream function is constant at the wall, Eq. (2.25) reduces to

$$\omega = -\frac{\partial}{\partial x} \left( \frac{1}{\rho} \frac{\partial \psi}{\partial x} \right).$$  \hspace{1cm} (2.33)

Equation (2.1) implies that $\rho$ is a function of $y$ only at the wall, Eq. (2.33) can then be written as

$$\omega = -\frac{1}{\rho} \frac{\partial^2 \psi}{\partial x^2}. \hspace{1cm} (2.34)$$

A second order Taylor series expansion of the stream function about the grid point $(j,k)$ on the wall, illustrated in Figure 2.3, yields
\begin{align}
\psi_{j-1,k} = \psi_{j,k} - \Delta x \left[ \frac{\partial \psi}{\partial x} \right]_{j,k} + \frac{\Delta x^2}{2} \left[ \frac{\partial^2 \psi}{\partial x^2} \right]_{j,k} + O(\Delta x^3). \tag{2.35}
\end{align}

Figure 2.3. Illustration of discretization about a grid point \((j,k)\) on the wall.

The discretized form of Eq. (2.34) is

\begin{align}
\omega_{j,k} = -\frac{1}{\rho_{j,k}} \left[ \frac{\partial^2 \psi}{\partial x^2} \right]_{j,k}. \tag{2.36}
\end{align}

By repeated use of the definition of the stream function in Eq. (2.24) and the no slip boundary condition gives,

\begin{align}
\left[ \frac{\partial \psi}{\partial x} \right]_{j,k} = (\rho v)_{j,k} = 0. \tag{2.37}
\end{align}

Equation (2.35) can then be rearranged to give the vorticity boundary condition at the wall as,

\begin{align}
\omega_{j,k} = -\frac{2}{\rho_{j,k} \Delta x^2} (\psi_{j-1,k} - \psi_{j,k}) + O(\Delta x). \tag{2.38}
\end{align}

It is clear from Eq. (2.38) that the implementation of the vorticity boundary condition is consistent with the overall first order accuracy of the numerical scheme. A higher order vorticity boundary condition can easily be derived, but cannot be implemented due to the structure of the adopted solver, which will be discussed later in this thesis.
An alternative boundary condition for $\omega$ is available in a pseudotransient formulation (Israeli, 1972), which has not been implemented in the code yet, but will be investigated in future studies.

The boundary conditions for stream function-vorticity formulation of a two-dimensional Wolfhard-Parker burner with uniform inflow conditions can be summarized as

Centerline ($x = 0$):

$$\psi = \omega = \frac{\partial Y_i}{\partial x} = \frac{\partial T}{\partial x} = 0, \quad i = 1, 2, \ldots, N,$$

(2.39)

Outflow ($y \to \infty$):

$$\frac{\partial \psi}{\partial y} = \frac{\partial \omega}{\partial y} = \frac{\partial Y_i}{\partial y} = \frac{\partial T}{\partial y} = 0, \quad i = 1, 2, \ldots, N,$$

(2.40)

Inflow ($y = 0$):

$0 < x < x_1$:

$$\psi = x \rho_F v_F,$$

(2.41a)

$$\omega = 0,$$

(2.41b)

$$Y_i = Y_r, \quad i = 1, 2, \ldots, N,$$

(2.41c)

$$T = T_F,$$

(2.41d)

$x_1 < x < x_2$:

$$\psi = x_1 \rho_F v_F + (x - x_1) \rho_{ox_i} v_{ox_i},$$

(2.42a)

$$\omega = 0,$$

(2.42b)

$$Y_i = Y_{ox_i}, \quad i = 1, 2, \ldots, N,$$

(2.42c)

$$T = T_{ox_i},$$

(2.42d)
\[ x_2 < x < x_w: \]
\[ \psi = x_1 \rho_F v_F + (x_2 - x_1) \rho_{o\kappa_1} v_{o\kappa_1} + (x - x_2) \rho_{o\kappa_2} v_{o\kappa_2}, \quad (2.43a) \]
\[ \omega = 0, \quad (2.43b) \]
\[ Y_i = Y_{i_o\kappa_2}, \quad i = 1, 2, \ldots, N, \quad (2.43c) \]
\[ T = T_{o\kappa_2}, \quad (2.43d) \]

Wall (\( x = x_w \)):
\[ \psi = x_1 \rho_F v_F + (x_2 - x_1) \rho_{o\kappa_1} v_{o\kappa_1} + (x_w - x_2) \rho_{o\kappa_2} v_{o\kappa_2}, \quad (2.44a) \]
\[ \omega = -\frac{2}{\rho} \Delta x^2 (\psi_{w-1} - \psi_w), \quad (2.44b) \]
\[ \frac{\partial Y_i}{\partial x} = 0, \quad i = 1, 2, \ldots, N, \quad (2.44c) \]
\[ \frac{\partial T}{\partial x} = 0, \quad (2.44d) \]

where \( \psi_{w-1} \) in Eq. (2.44b) is the value of the stream function at a grid point next to the wall.
3. Flame sheet model

Equations (2.25)-(2.28) are highly coupled and non-linear and their solution requires a good starting estimate. When a sufficiently good initial solution for the discretized equations is known, then, a converged solution can be obtained in fewer iterations than if no initial solution is available. Moreover, without a good initial solution the numerical technique chosen might never converge. A technique to obtain such a starting solution is to utilize the flame sheet approximation (Williams, 1985). This approximation is based on the assumption of infinitely fast kinetics, where the fuel and the oxidizer are separated by an exothermic reaction zone (flame sheet). In this zone, the fuel and the oxidizer are in stoichiometric proportions and the temperature is at a maximum. The infinitely fast kinetics assumption implies that the fuel and the oxidizer are being completely consumed in the reaction zone, allowing no leakage of reactants. In reality, the oxidation of the fuel to form intermediates and products proceeds through a set of elementary chemical reactions. Still, the flame sheet approximation has proven to be very valuable in generating starting solutions, and can provide initial solution profiles for the stream function, the vorticity, the temperature, and the major species, i.e., fuel, O₂, N₂, CO₂ and H₂O (Burke and Schumann, 1928, Mitchell, 1975, Mitchell et al., 1980, Smooke et al., 1984). The stoichiometric fuel-oxygen interface and, hence, the points of heat release, can thus be accurately predicted.

3.1 Derivation of flame sheet representation of the conservation equations

The objective is to express the energy equation, Eq. (2.27), and the chemical species equations, Eq. (2.28), in a similar form. This enables the chemical source term for species \( i \), \( \dot{w}_i \), to be removed from the equations by considering suitable linear combinations of dependent variables (coupling functions). This approach, often referred to as the Schvab-
Zeldowich formulation, has the advantage that the number of partial differential equations that must be solved are reduced from $N + 2$ to only three.

To enable a coupling, as discussed above, the following assumption are made

i. A single-step irreversible reaction for the fuel ($F$), the oxidizer ($OX$) and the products ($P_j$) on the form

$$v_F F + v_{OX} OX \rightarrow \sum_{j=1}^{N_p} v_{P_j} P_j,$$

where $v_F$, $v_{OX}$, and $v_{P_j}$ are the stoichiometric coefficients of the fuel, the oxidizer and the $j$-th product, respectively, and $N_p$ is the number of products.

ii. Nitrogen is an inert that does not participate in chemical reactions.

iii. Negligible energy transfer due to mass transfer, i.e.,

$$\frac{\partial}{\partial x} \left( \rho T \sum_{i=1}^{N} c_{p_i} Y_i V_i \right) + \frac{\partial}{\partial y} \left( \rho T \sum_{i=1}^{N} c_{p_i} Y_i V_i \right) = 0,$$

iv. $c_p$ and $c_{p_i}$ are assumed to be constant independent of temperature.

v. Fick's law can be used as defined in Eqs. (2.7) and (2.8).

With these approximations the governing equations for the stream function-vorticity formulation can be written as

Stream function:

$$-\frac{\partial}{\partial x} \left( \frac{1}{\rho} \frac{\partial \psi}{\partial x} \right) - \frac{\partial}{\partial y} \left( \frac{1}{\rho} \frac{\partial \psi}{\partial y} \right) - \omega = 0,$$

Vorticity:

$$-\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} \omega \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} \omega \right) - \frac{\partial}{\partial x} \left( \mu \frac{\partial \omega}{\partial x} \right) - \frac{\partial}{\partial y} \left( \mu \frac{\partial \omega}{\partial y} \right) =$$

$$\frac{\partial \rho}{\partial x} \frac{\partial}{\partial y} \left( \frac{\mu^2 + v^2}{2} \right) - \frac{\partial \rho}{\partial y} \frac{\partial}{\partial x} \left( \frac{\mu^2 + v^2}{2} \right) + g \frac{\partial \rho}{\partial x},$$

(3.3)
Energy:

\[-\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} T \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} T \right) - \frac{\partial}{\partial x} \left( \frac{\lambda}{c_p} \frac{\partial T}{\partial x} \right) - \frac{\partial}{\partial y} \left( \frac{\lambda}{c_p} \frac{\partial T}{\partial y} \right) =
\]
\[+ \frac{1}{c_p} (W_F \nu F \Delta h^0_{fr} + W_{ox} \nu_{ox} \Delta h^0_{ox} - \sum_{j=1}^{N_p} W_{p_j} \nu_{p_j} \Delta h^0_{fr_j}) \dot{w}, \quad (3.4)\]

Fuel:

\[-\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} Y_F \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} Y_F \right) - \frac{\partial}{\partial x} \left( \rho D_F \frac{\partial Y_F}{\partial x} \right) - \frac{\partial}{\partial y} \left( \rho D_F \frac{\partial Y_F}{\partial y} \right) + W_F \nu F \dot{w} = 0, \quad (3.5)\]

Oxidizer:

\[-\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} Y_{ox} \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} Y_{ox} \right) - \frac{\partial}{\partial x} \left( \rho D_{ox} \frac{\partial Y_{ox}}{\partial x} \right) - \frac{\partial}{\partial y} \left( \rho D_{ox} \frac{\partial Y_{ox}}{\partial y} \right) + W_{ox} \nu_{ox} \dot{w} = 0, \quad (3.6)\]

Products:

\[-\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} Y_{p_j} \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} Y_{p_j} \right) - \frac{\partial}{\partial x} \left( \rho D_{p_j} \frac{\partial Y_{p_j}}{\partial x} \right) - \frac{\partial}{\partial y} \left( \rho D_{p_j} \frac{\partial Y_{p_j}}{\partial y} \right)
\]
\[-W_{p_j} \nu_{p_j} \dot{w} = 0, \quad j = 1, 2, ..., N_p, \quad (3.7)\]

Nitrogen:

\[-\frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} Y_{N_2} \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} Y_{N_2} \right) - \frac{\partial}{\partial x} \left( \rho D_{N_2} \frac{\partial Y_{N_2}}{\partial x} \right) - \frac{\partial}{\partial y} \left( \rho D_{N_2} \frac{\partial Y_{N_2}}{\partial y} \right) = 0, \quad (3.8)\]

where

\[\dot{w} = \frac{\dot{w}_F}{W_F \nu_F} = \frac{\dot{w}_{ox}}{W_{ox} \nu_{ox}} = \frac{\dot{w}_{p_j}}{W_{p_j} \nu_{p_j}}, \quad (3.9)\]

is the rate of progress of the reaction.

If

\[q = \Delta h^0_{fr} + \frac{W_{ox} \nu_{ox}}{W_F \nu F} \Delta h^0_{ox} - \sum_{j=1}^{N_p} \frac{W_{p_j} \nu_{p_j}}{W_F \nu_F} \Delta h^0_{fr_j}, \quad (3.10)\]

denotes the heat release per unit mass of fuel, Eq. (3.4) can be expressed as
\[- \frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} T \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} T \right) - \frac{\partial}{\partial x} \left( \frac{\lambda}{c_p} \frac{\partial T}{\partial x} \right) - \frac{\partial}{\partial y} \left( \frac{\lambda}{c_p} \frac{\partial T}{\partial y} \right) - \frac{q}{c_p} W_F v_F \dot{w} = 0. \tag{3.11}\]

By assuming the Lewis numbers,
\[Le_i = \frac{\lambda}{\rho D_i c_p}, \quad i = F, OX, P_j, N_2, \tag{3.12}\]
are all equal to unity, the individual "pseudo-diffusion" coefficients are equal to a unique diffusion coefficient, \(D\),
\[D = \frac{\lambda}{\rho c_p}. \tag{3.13}\]

Equation (3.11) and Eqs. (3.5)-(3.8) now have the same form and can be expressed as
\[L(\alpha) = \dot{w}, \tag{3.14}\]
where \(L(\ )\) is a linear operator defined as
\[L(\alpha_i) = - \frac{\partial}{\partial x} \left( \frac{\partial \psi}{\partial y} \alpha_i \right) + \frac{\partial}{\partial y} \left( \frac{\partial \psi}{\partial x} \alpha_i \right) - \frac{\partial}{\partial x} \left( \rho D \frac{\partial \alpha_i}{\partial x} \right) - \frac{\partial}{\partial y} \left( \rho D \frac{\partial \alpha_i}{\partial y} \right), \tag{3.15}\]
i = T, F, OX, P_j, and where
\[\alpha_T = \frac{c_p}{q} \frac{T}{W_F v_F}, \tag{3.16}\]
\[\alpha_F = - \frac{Y_F}{W_F v_F}, \tag{3.17}\]
\[\alpha_{OX} = - \frac{Y_{OX}}{W_{OX} v_{OX}}, \tag{3.18}\]
\[\alpha_{P_j} = \frac{Y_{P_j}}{W_{P_j} v_{P_j}}. \tag{3.19}\]

The nonlinear term \(\dot{w}\) can thus be eliminated from all except one of the relations in Eq. (3.14). If, for example, one solves the equation for \(\alpha_F\),
\[L(\alpha_F) = \dot{w}, \tag{3.20}\]
then the other variables can be determined by linear equations for the coupling function \(\beta\).
\[ L(\beta_i) = 0, \quad i = T, OX, P_i, \]  

(3.21)

where

\[ \beta_T = \alpha_T - \alpha_F, \]  

(3.22)

\[ \beta_{OX} = \alpha_{OX} - \alpha_F, \]  

(3.23)

\[ \beta_{P_i} = \alpha_{P_i} - \alpha_F. \]  

(3.24)

Another suitable coupling function is the mixture fraction \( Z \) defined as

\[ Z = \frac{\beta_{OX} - \beta_{OX_w}}{\beta_{P_w} - \beta_{OX_w}}, \]  

(3.25)

where subscripts \( w \) and \( 0 \) correspond to the inflow conditions on the oxidizer side and the fuel side, respectively. Since the definition of \( Z \) is a linear function of \( \beta_i \), the conservation equation for \( Z \) satisfies the relation,

\[ -\frac{\partial}{\partial x} (\frac{\partial \psi}{\partial y} Z) + \frac{\partial}{\partial y} (\frac{\partial \psi}{\partial x} Z) - \frac{\partial}{\partial x} (\rho D \frac{\partial Z}{\partial x}) - \frac{\partial}{\partial y} (\rho D \frac{\partial Z}{\partial y}) = 0. \]  

(3.26)

The chemical source term does not appear in Eq. (3.26) because \( Z \) is a conserved scalar that can neither be created nor destroyed by chemical reactions. The equation for the mixture fraction, Eq. (3.26), can be solved with the stream function equation, Eq. (3.2), and the vorticity equation, Eq. (3.3), to generate flame sheet solutions.

The mass fractions and the temperature can be recovered from the mixture fraction. If no oxidizer is assumed in the incoming fuel stream and no fuel in the incoming oxidizer streams, then Eq. (3.25) can be written as

\[ Z = \frac{\alpha_{OX} - \alpha_F - \alpha_{OX_w}}{-\alpha_{P_w} - \alpha_{OX_w}}. \]  

(3.27)

The value of \( Z \) at the flame front, \( Z_f \), is the stoichiometric mixture fraction and can be determined by imposing the restriction that fuel and oxidizer are completely consumed at the flame front, i.e. \( Y_F = Y_{OX} = 0 \). Allowing \( \alpha_{OX} = \alpha_F = 0 \) in Eq. (3.27) gives

\[ Z_f = Y_{OX_w} / (Y_{OX_w} + \frac{W_{OX} V_{OX}}{W_F V_F} Y_{P_w}). \]  

(3.28)
For methane-air combustion the value is $Z_f = 0.05482$. If the oxygen content is different in the two oxidizer slots, as for the dual mode case, a unique value of $Z_f$ cannot be determined. In the following solution, the oxygen content for the two oxidizer slots is assumed to be the same, otherwise the method described above does not apply.

On the fuel side, $Z > Z_f$:

$$T = T_0 Z + \left[ T_w + Y_{ox} \frac{q}{c_p} \frac{W_F V_F}{W_{ox} V_{ox}} \right] (1 - Z), \quad (3.29)$$

$$Y_F = Y_{F_0} Z + Y_{ox} \frac{W_F V_F}{W_{ox} V_{ox}} (Z - 1), \quad (3.30)$$

$$Y_{ox} = 0, \quad (3.31)$$

$$Y_{p_j} = Y_{ox} \frac{W_{p_j} V_{p_j}}{W_{ox} V_{ox}} (1 - Z), \quad (3.32)$$

$$Y_{N_2} = 1 - Y_F - Y_{ox} - \sum_{j=1}^{N_p} Y_{p_j}. \quad (3.33)$$

On the oxidizer side, $Z < Z_f$:

$$T = T_w (1 - Z) + \left[ T_0 + Y_{F_0} \frac{q}{c_p} \right] Z, \quad (3.34)$$

$$Y_F = 0, \quad (3.35)$$

$$Y_{ox} = Y_{ox} (1 - Z) - Y_{F_0} \frac{W_{ox} V_{ox}}{W_F V_F} Z, \quad (3.36)$$

$$Y_{p_j} = Y_F \frac{W_{p_j} V_{p_j}}{W_F V_F} Z, \quad (3.37)$$

$$Y_{N_2} = 1 - Y_F - Y_{ox} - \sum_{j=1}^{N_p} Y_{p_j}. \quad (3.38)$$
Eqs. (3.33) and (3.38) are used to ensure conservation of mass. The mass fractions and the temperature vary linearly with the mixture fraction. Figure 3.1 illustrates the relationships, and as suggested by Eqs. (3.29)-(3.38) consists entirely of straight lines.

![Diagram](image)

Figure 3.1. Mass fraction of various species and temperature as functions of the mixture fraction.

### 3.2 Boundary conditions for the flame sheet model computations

The boundary conditions for $\psi$ and $\omega$ are the same as in chapter 2. The boundary conditions for the mixture fraction $Z$ are derived in such a way that they are consistent with the boundary conditions for $T$ and $Y_i$. At the wall, one has the choice of imposing Dirichlet ($Z = 0$) or Neuman ($\partial Z/\partial x = 0$) boundary conditions. Dirichlet boundary conditions correspond to a fixed wall temperature and fixed oxidizer concentration, and Neuman boundary conditions correspond to adiabatic wall and no mass diffusion into the wall. If Dirichlet boundary conditions are implemented, underventilated flames cannot be
modeled, since the flame will never attach the wall. Neuman boundary conditions are therefore preferable. The boundary conditions for the flame sheet can be written as

Centerline \((x = 0)\):

\[
\psi = \omega = \frac{\partial Z}{\partial x} = 0, \quad (3.39)
\]

Outflow \((y \to \infty)\):

\[
\frac{\partial \psi}{\partial y} = \frac{\partial \omega}{\partial y} = \frac{\partial Z}{\partial y} = 0, \quad (3.40)
\]

Inflow \((y = 0)\):

\(0 < x < x_1\):

\[
\psi = x \rho_F v_F, \quad (3.41a)
\]

\[
\omega = 0, \quad (3.41b)
\]

\[
Z = 1, \quad (3.41c)
\]

\(x_1 < x < x_2\):

\[
\psi = x_1 \rho_F v_F + (x - x_1) \rho_{\text{ox}_1} v_{\text{ox}_1}, \quad (3.42a)
\]

\[
\omega = 0, \quad (3.42b)
\]

\[
Z = 0, \quad (3.42c)
\]

\(x_2 < x < x_w\):

\[
\psi = x_1 \rho_F v_F + (x_2 - x_1) \rho_{\text{ox}_1} v_{\text{ox}_1} + (x - x_2) \rho_{\text{ox}_2} v_{\text{ox}_2}, \quad (3.43a)
\]

\[
\omega = 0, \quad (3.43b)
\]

\[
Z = 0, \quad (3.43c)
\]

Wall \((x = x_w)\):

\[
\psi = x_1 \rho_F v_F + (x_2 - x_1) \rho_{\text{ox}_1} v_{\text{ox}_1} + (x - x_2) \rho_{\text{ox}_2} v_{\text{ox}_2}, \quad (3.44a)
\]

\[
\omega = -\frac{2}{\rho \Delta x^2} (\psi_{w-1} - \psi_w), \quad (3.44b)
\]
\[
\frac{\partial Z}{\partial x} = 0. 
\] (3.44c)

The temperature field obtained from the solution is used to update the temperature dependent properties. The density is evaluated through Eq. (2.29). By using the definition of the Prandtl number

\[
Pr = \frac{\mu c_p}{\lambda},
\] (3.45)

and the assumption that all Lewis numbers are equal to unity, one can express the mass-diffusion flux as

\[
\rho D = \frac{\lambda}{c_p} = \frac{\mu}{(Pr)_{ref}},
\] (3.46)

where \((Pr)_{ref}\) is a reference Prandtl number. The viscosity can, for example, be obtained from the power law

\[
\mu = \mu_{ref} \left(\frac{T}{T_{ref}}\right)^r,
\] (3.47)

where \(r = 0.7\), \(T_{ref} = 298\) K and \(\mu_{ref} = 18.5 \times 10^{-6}\) Ns/m² is a reference value for air (Kanury, 1982). The value of the temperature exponent \(r\) is obtained from a one-dimensional finite rate chemistry computation for methane as fuel performed by Smooke et al. (1989). The scaled heat release parameter \(q/c_p\) can be determined from an estimate of the peak temperature.

In the flame sheet model the peak temperature is the adiabatic stoichiometric flame temperature, since infinitely fast kinetics are assumed. The purpose of the flame sheet model though is to produce a good starting estimate for the finite rate chemistry solution, and therefore a value corresponding to finite reaction rates is used. The peak temperature for most hydrocarbon fuels is approximately 90% of the adiabatic stoichiometric value (Boedeker and Dobbs, 1986).
3.3 An analytic solution for diffusion flames stabilized on a Wolfhard-Parker burner

To help verify numerical results and to generate a good initial estimate for the flame sheet solutions, thus reducing the computational time, an analytic solution of the flame front location is derived in this section.

A technique to determine the flame shape is to follow the Burke and Schumann (1928) approach for an axisymmetric diffusion flame. In addition to some of the assumptions made earlier in this chapter, the following approximations need to be made:

1. Vertical diffusion is negligible in comparison with horizontal diffusion, i.e.

   \[
   \frac{\partial y_v}{\partial y} \ll \frac{\partial y_v}{\partial x}. \tag{3.48}
   \]

   This is a good assumption when the flame height is much larger than the width of the fuel slot, as for the Wolfhard-Parker burner.

2. Mixing is caused by diffusion only, hence the horizontal velocity component is equal to zero,

   \[ u = 0. \tag{3.49} \]

   The heat release at the flame front causes the flow to expand, and gives \( u \) a non-zero value, making this assumption questionable.

3. The total mass flux in the vertical direction is constant everywhere in the burner,

   \[ \rho v = \text{constant}. \tag{3.50} \]

   This assumption holds for uniform inflow conditions and if \( \rho v \) is the same for both the fuel and the oxidizer streams, otherwise less accurate results can be expected.

4. The product \( \rho D \) is constant throughout the burner,

   \[ \rho D = \text{constant}. \tag{3.51} \]

   This assumption constitutes a reasonably accurate approximation, especially if \( \rho D \) does not differ too much between the fuel and oxidizer streams.
Roper (1977) applied the Burke-Schumann approach to slot burners and found an error in the prediction of the flame height by a factor of about six compared to experiments when buoyancy effects could be neglected for flame heights below 6 cm. The results for the axisymmetric case are in excellent agreement with experiments, in spite of some of the assumptions made above. An expansion of the flow is associated with a change in the flame width, and hence a change in the diffusion. In the axisymmetric case this change is counter-balanced by an opposing change in the time required for diffusion, and results in a balancing of these two effects. For a slot burner this cancellation does not occur, explaining the discrepancy. Nevertheless, an analytic solution is still helpful for predicting the flame shape, especially if the flame height is large and the flame momentum controlled.

With the assumptions listed above a differential equation can be obtained for the coupling function for fuel and oxidizer in Eq. (3.23) of the form

$$\frac{v}{D} \frac{\partial \beta}{\partial y} - \frac{\partial^2 \beta}{\partial x^2} = 0. \quad (3.52)$$

The boundary conditions for $\beta$ are the same as for the mixture fraction $Z$ in Eq. (3.39) and Eqs. (3.41)-(3.44). Note that only one boundary condition in the vertical direction is needed since vertical diffusion is neglected. Therefore a boundary condition at the outflow is not necessary.

Introducing dimensionless coordinates

$$\xi = \frac{x}{x_w}, \quad (3.53)$$

$$\eta = \frac{yD}{v x_w^2}, \quad (3.54)$$

the parameters,

$$c = \frac{x_l}{x_w}, \quad (3.55)$$

$$v = \frac{Y_{OX,w} W_F V_F}{Y_{F,0} W_{OX} V_{OX}}, \quad (3.56)$$

and the variable,
\[ \gamma = \frac{BW_F Y_F}{Y_{F,0}}. \]  

Equation (3.52) and the boundary conditions become

\[ \frac{\partial \gamma}{\partial \eta} - \frac{\partial^2 \gamma}{\partial \xi^2} = 0, \]  

Centerline ($\xi = 0$):

\[ \frac{\partial \gamma}{\partial x} = 0, \]  

Inflow ($\eta = 0$):

\[ 0 \leq \xi \leq c: \]

\[ \gamma = 1, \]  

\[ c \leq \xi \leq 1: \]

\[ \gamma = -\nu, \]  

Wall ($\xi = 1$):

\[ \frac{\partial \gamma}{\partial x} = 0. \]

Since the governing equation is linear an analytic solution can be derived using the method of separation of variables. Assume a solution of the form

\[ \gamma(\xi, \eta) = X(\xi)Y(\eta), \]  

which substituted in Eq. (3.58) gives

\[ \frac{\partial Y}{\partial \eta} + \phi^2 Y = 0, \]  

\[ \frac{\partial^2 X}{\partial \xi^2} + \phi^2 X = 0. \]

The solution of Eqs. (3.64) and (3.65) when $\phi \neq 0$ is

\[ \gamma = \left[ C_1 \sin(\phi \xi) + C_2 \cos(\phi \xi) \right] e^{-\nu \eta}, \]  

36
where $C_1$, $C_2$ and $\phi$ are constants. Boundary condition (3.59) yields

$$C_1 = 0,$$

(3.67)

whereas (3.62) yields

$$\phi_n = n\pi, \quad n = 1, 2, \ldots$$

(3.68)

Equation (3.66) thus reduces to,

$$\gamma = \sum_{n=1}^{\infty} B_n \cos(\phi_n \xi) e^{-\beta \eta},$$

(3.69)

where $B_n$ is a constant. Boundary conditions (3.60) and (3.61) give

$$f(\xi) = \sum_{n=1}^{\infty} B_n \cos(\phi_n \xi),$$

(3.70)

where $f(\xi)$ is a function that corresponds to the inflow boundary conditions. The constant $B_n$ can be determined by applying orthogonality condition to Eq. (3.70) which gives

$$B_n = \frac{2}{\phi_n} (1 + \nu) \sin(\phi_n \xi).$$

(3.71)

The special case of $\phi = 0$ as a solution to Eqs. (3.64) and (3.65) generates a constant. This constant is the equilibrium value of $\gamma$ for large $\eta$ ($\eta \to \infty$). An equilibrium analysis based on volumetric flow rates gives

$$\gamma = (1 + \nu) c - \nu.$$

(3.72)

Adding the two solutions together gives the final solution

$$\gamma = (1 + \nu) c - \nu + 2(1 + \nu) \sum_{n=1}^{\infty} \frac{1}{\phi_n} \sin(\phi_n \xi) \cos(\phi_n \xi) e^{-\beta \eta}.$$ 

(3.73)

Since $\beta = 0$ denotes the location of the flame front, from Eq. (3.57) it follows that

$$\gamma = 0.$$

(3.74)

By using Eq. (3.74) in Eq. (3.73) the location of the flame front can be determined. Typical flame shapes are shown in Figure 3.3 for two different values of the parameter $\nu$. 

37
The physical flame height can be recovered from $\eta$ by using Eq. (3.54), and an average value of $\rho v$. The diffusion coefficient, $D$, can be obtained by utilizing Eq. (3.46), where the viscosity is evaluated at an average temperature.

For a rectangular burner like the Wolfhard-Parker burner, the flame height is large enough to cause the exponential term in Eq. (3.73) to decrease rapidly as $n$ increases at these values of $\eta$. An estimate of the dimensionless flame height can therefore be obtained by considering only the first term in the summation ($n = 1$). An underventilated flame attaches to the wall, thus $\xi = 1$ gives the dimensionless flame height.
\[ \eta = -\frac{1}{\phi_1^2} \ln \left( \frac{v - (1+v)c \phi_1}{2(1+v)\sin(\phi_1)c \cos(\phi_1)} \right). \] (3.75)

An overventilated flame closes at the centerline, thus \( \xi = 1 \) gives the dimensionless flame height

\[ \eta = -\frac{1}{\phi_1^2} \ln \left( \frac{v - (1+v)c \phi_1}{2(1+v)\sin(\phi_1)c} \right). \] (3.76)
4. Method of solution for Wolfhard-Parker burner

Non-linear equations can be solved only if an iterative method is used. If only the steady state-solution is of interest implicit schemes are more attractive than explicit schemes, since the former can be constructed without a time step restriction in the linear stability sense. In practice there is a time step limitation, but it is much less restrictive than for explicit schemes. Thus, an implicit scheme is preferable in the modeling of steady state flames.

For flow problems that produce non-linear systems of equations that are not diagonally dominant, the two most popular solution methods are Newton's method and the pseudotransient, approximative factorization method (Fletcher, 1991). Newton's method has been used to solve steady stream function-vorticity equations successfully (Fornberg, 1985, Rizzetta, 1985). Inherent with Newton's method is a quadratic rate of convergence and the ability to force the solution to within a desired accuracy within a time frame not possible with the pseudotransient, approximative factorization method. The main drawbacks with Newton's method are the small radius of convergence and the high computational cost to solve the resulting system after discretization.

The radius of convergence can be increased if a modified Newton's method is used. Smooke et al. (1989) have applied this method to solve axisymmetric diffusion flames. The computational cost for the modified Newton's method is similar to Newton's method, and to make the modeling more feasible, the pseudotransient, approximative factorization method is chosen as the solution method.

4.1 Pseudotransient method

The flame sheet model facilitates achieving converged solutions for detailed chemistry problems. Nevertheless, most numerical techniques that have been used to solve steady
flame problems, utilize time iteration to ensure convergence (Dixon-Lewis, 1967, Warnatz, 1978, Westbrook and Dryer, 1979, Heimerl and Coffee, 1980, Smooke et al., 1989, Smooke and Giovangigli, 1992). The problem considered here is of a steady nature, therefore one could directly solve the governing equations. An alternative way to solve the steady problem is to construct an equivalent unsteady problem and to march the transient solution in time until steady-state is reached. Time then plays the role of an iteration parameter. This is a computationally less efficient but a more robust method. Let \( U \) denote the dependent variable vector, \( U = (\psi, \omega, Z)^T \), then Eqs. (3.2)-(3.3) and Eq. (3.26) can be written in vector- and residual form as

\[
F(U) = 0, \tag{4.1}
\]

where \( 0 \) is the null vector. An unsteady formulation of Eq. (4.1) can be expressed as,

\[
\frac{\partial U}{\partial t} + F(U) = 0. \tag{4.2}
\]

Note that Eq. (4.2) cannot be used to solve the truly transient problem. With the time derivative in Eq. (4.2) discretized with a backward difference formula around time level \( n + 1 \),

\[
\frac{(U^{n+1} - U^n)}{\Delta t} + F(U^{n+1}) = 0, \tag{4.3}
\]

where

\[
\Delta t = t^{n+1} - t^n. \tag{4.4}
\]

At each time level a system of non-linear equations have to be solved that looks very similar to Eq. (4.1). By comparing Eqs. (4.1) and (4.3) it is clear that the steady state Jacobian is more diagonally dominant for the unsteady case through the quantity \( 1/\Delta t \). From a physical point of view, as the time step approaches zero, the solution for \( t^{n+1} \) should approach the solution \( t^n \), thus convergence should be reliable from one time step to another. Equation (4.3) is an effective way to under-relax Eq. (4.1), and produces a better conditioned system.
However, since the problem considered is of a steady nature, the transient solution is of no interest. The sequence of time step should therefore be chosen to minimize the number of time steps to convergence. The objective is to choose a time step that is large enough to make progress toward the steady solution, but not so large that the transient method also has convergence difficulties.

The conservation equations for the Wolfhard-Parker burner, can be written in vector form as

\[
\frac{\partial U}{\partial t} \frac{\partial H}{\partial x} + \frac{\partial G}{\partial y} - \frac{\partial}{\partial x} (Q \frac{\partial U}{\partial x}) - \frac{\partial}{\partial y} (Q \frac{\partial U}{\partial y}) + K = M,
\]

where

\[
H = (0, -\rho u \omega, -\rho u Z)^T,
\]

\[
G = (0, \rho v \omega, \rho v Z)^T,
\]

are convective vectors,

\[
Q = \begin{bmatrix}
1/ \rho & 0 & 0 \\
0 & \mu & 0 \\
0 & 0 & \rho D
\end{bmatrix},
\]

is the diffusion matrix,

\[
K = (-\omega, 0, 0)^T,
\]

is the source vector, and

\[
M = (0, \frac{\partial \rho}{\partial x} \frac{\partial}{\partial y} \left( \frac{u^2 + v^2}{2} \right) - \frac{\partial \rho}{\partial y} \frac{\partial}{\partial x} \left( \frac{u^2 + v^2}{2} \right) + \rho \frac{\partial}{\partial x}, 0)^T,
\]

is the dynamic pressure and gravity vector. Note that in Eqs. (4.6) and (4.7) the velocities are used instead of the gradients of the stream function for reasons to be explained later.
4.2 Approximate factorization

A major advantage of the unsteady formulation is that splitting techniques can be used (Peaceman and Rachford, 1955, Douglas and Gunn, 1964, Gourlay, 1977, Fletcher, 1991). When implicit schemes in two dimensions are used, special procedures are necessary if low memory requirements are desired. Standard finite difference discretization produces a banded matrix, the inversion of which is computationally very expensive. The special procedures are often built around some means of splitting the equations on a convenient coordinate basis, thus breaking the numerical process down into one of sequentially solving simpler problems. Figure 4.1 illustrates the splitting technique. The advancement of the solution by one time step takes place as a sequence of two half-steps. At each half-step only the terms associated with a specific spatial direction are treated implicitly. Therefore, at each half-step only three implicit terms appear and these can be grouped adjacent to the main diagonal. The problem can then be solved efficiently by the block Thomas algorithm (Isaacson and Keller, 1966).

![Sweep in y-direction](direction)

Sweep in y-direction for constant $j$

$k+1$

$k$

$k-1$

$j-1$ $j$ $j+1$

Sweep in x-direction for constant $k$

Figure 4.1. Illustration of splitting technique.
For the present study the approximate factorization method is chosen as the splitting technique. There are several advantages with the pseudotransient formulation combined with approximative factorization (Fletcher, 1991):

- Each iteration is economical since the block Thomas algorithm can be used.
- Very large radius of convergence.
- Small memory requirements (splitting uncouples grid lines).
- Detects if problem is not steady.
- Approximate solution can be exploited.

The main disadvantage is the low convergence rate, since a large number of iterations must be performed.

The non linear terms in Eq. (4.5) can be linearized around time level \( n + 1 \) as

\[
H^{n+1} = H^n + \left( \frac{\partial H}{\partial t} \right)^n \Delta t + O(\Delta t^2) = H^n + A\Delta U^{n+1} + O(\|\Delta U\|^2),
\]  
\[
G^{n+1} = G^n + \left( \frac{\partial G}{\partial t} \right)^n \Delta t + O(\Delta t^2) = G^n + B\Delta U^{n+1} + O(\|\Delta U\|^2),
\]  
\[
K^{n+1} = K^n + \left( \frac{\partial K}{\partial t} \right)^n \Delta t + O(\Delta t^2) = K^n + D\Delta U^{n+1} + O(\|\Delta U\|^2),
\]

\[
U^{n+1} = U^n + \Delta U^{n+1},
\]

where

\[
A = \left( \frac{\partial H}{\partial U} \right)^n,
\]  
\[
B = \left( \frac{\partial G}{\partial U} \right)^n,
\]  
\[
D = \left( \frac{\partial K}{\partial U} \right)^n,
\]
denote the Jacobian matrices. Substituting Eqs. (4.11)-(4.17) back in Eq. (4.5) yields a semi discrete form,

$$\frac{\Delta U^{n+1}}{\Delta t} - \frac{\partial}{\partial x}(A\Delta U^{n+1}) + \frac{\partial}{\partial y}(B\Delta U^{n+1}) - \frac{\partial}{\partial x}\left(Q \frac{\partial \Delta U^{n+1}}{\partial x}\right) - \frac{\partial}{\partial y}\left(Q \frac{\partial \Delta U^{n+1}}{\partial y}\right)$$

$$+ D\Delta U^{n+1} = \text{RHS}^n,$$  \hspace{1cm} (4.18)

where

$$\text{RHS}^n = M^n + \frac{\partial H^n}{\partial x} - \frac{\partial G^n}{\partial y} + \frac{\partial}{\partial x}\left(Q \frac{\partial U^n}{\partial x}\right) + \frac{\partial}{\partial y}\left(Q \frac{\partial U^n}{\partial y}\right) - K^n.$$  \hspace{1cm} (4.19)

With $I$ denoting the identity matrix, Eq. (4.18) can also be expressed as

$$\left[\frac{I}{\Delta t} - \frac{\partial}{\partial x} A + \frac{\partial}{\partial y} B - \frac{\partial}{\partial x}\left(Q \frac{\partial}{\partial x}\right) - \frac{\partial}{\partial y}\left(Q \frac{\partial}{\partial y}\right) + D\right] \Delta U^{n+1} = \text{RHS}^n.$$  \hspace{1cm} (4.20)

The matrix on the left-hand side of Eq. (4.20) can be written in approximate factorized form as

$$\left[I + \Delta t\left\{-\frac{\partial}{\partial x} A + \frac{\partial}{\partial x}\left(Q \frac{\partial}{\partial x}\right) + D\right\}\right] \Delta U^{n+1} = \Delta t \text{RHS}^n.$$  \hspace{1cm} (4.21)

The factorization in Eq. (4.21) is accurate to the order $\Delta t^2$, and is not unique, since one has the choice of where to involve $D$. Other possible factorizations are

$$\left[I + \Delta t\left\{-\frac{\partial}{\partial x} A - \frac{\partial}{\partial x}\left(Q \frac{\partial}{\partial x}\right)\right\}\right] \Delta U^{n+1} = \Delta t \text{RHS}^n,$$  \hspace{1cm} (4.22)

$$\left[I + \Delta t\left\{-\frac{\partial}{\partial x} A - \frac{\partial}{\partial x}\left(Q \frac{\partial}{\partial x}\right) + \frac{D}{2}\right\}\right] \Delta U^{n+1} = \Delta t \text{RHS}^n.$$  \hspace{1cm} (4.23)

Equations (4.21) and (4.22) are preferable to Eq. (4.23) since $D$ is involved in only one of the factors. Equation (4.21) can be solved in two steps

$$\left[I + \Delta t\left\{-\frac{\partial}{\partial x} A - \frac{\partial}{\partial x}\left(Q \frac{\partial}{\partial x}\right) + D\right\}\right] \Delta U^* = \Delta t \text{RHS}^n,$$  \hspace{1cm} (4.24)

45
\[
\left[ I + \Delta t \left( \frac{\partial}{\partial y} B - \frac{\partial}{\partial y} \left( \frac{Q}{\partial y} \right) \right) \right] \Delta U^{n+1} = \Delta U^*.
\] (4.25)

Equations (4.24) and (4.25) represent the linear system of equations that must be solved at each iteration for \( \Delta U^{n+1} \). The solution at the implicit time level \((n+1)\) is obtained utilizing Eq. (4.14). Equations (4.24)-(4.25) are implemented in the numerical code for solving the governing Eqs. (3.2)-(3.2) and (3.26) by dividing both equations by \( \Delta t \) to produce a better conditioned system and to reduce the number of operations.

4.3 Discretization

The objective is to obtain a discrete solution of the governing Eqs. (4.24) and (4.25) on the staggered mesh \( \mathcal{M}_x \), which are formed by the intersection of the lines of the mesh \( \mathcal{M}_x \)
\[
\mathcal{M}_x = \left\{ 0 = x_0 < x_1 < \ldots < x_j \ldots < x_{M_x - 1} = x_w \right\},
\] (4.26)
and the mesh \( \mathcal{M}_y \)
\[
\mathcal{M}_y = \left\{ 0 = y_0 < y_1 < \ldots < y_k \ldots < y_{M_y - 1} = y_w \right\},
\] (4.27)
where \( M_x \) and \( M_y \) are the number of grid points in the horizontal and vertical directions, respectively. The derivatives in the equations of interest are approximated by finite difference expressions. The problem of finding an analytical solution to the equations is then converted into one of finding an approximation to this solution at each point \((x_j, y_k)\) of the mesh \( \mathcal{M}_2 \). The ordering of \( \Delta U^{n+1} \) used is
\[
\Delta U^{n+1} = \left[ (\Delta \psi_{1,1}^{n+1}, \Delta \omega_{1,1}^{n+1}, \Delta Z_{1,1}^{n+1}), (\Delta \psi_{2,1}^{n+1}, \Delta \omega_{2,1}^{n+1}, \Delta Z_{2,1}^{n+1}), \ldots (\Delta \psi_{j,k}^{n+1}, \Delta \omega_{j,k}^{n+1}, \Delta Z_{j,k}^{n+1}), \ldots \right]^T.
\] (4.28)

The diffusive terms can be discretized with a second order accurate central-difference scheme.
\[
\frac{\partial}{\partial x} \left( Q \frac{\partial (\Delta U^{n+1})}{\partial x} \right)_{j,k} = 2 \frac{\Delta y_{k+1} + \Delta y_k}{\Delta x_{j+1}} \left( Q_{j+1/2,k} \frac{\delta_x^+ (\Delta U^{n+1})_{j,k}}{\Delta x_{j+1}} - Q_{j-1/2,k} \frac{\delta_x^- (\Delta U^{n+1})_{j,k}}{\Delta x_{j}} \right),
\]
\[
\frac{\partial}{\partial y} \left( Q \frac{\partial (\Delta U^{n+1})}{\partial y} \right)_{j,k} = 2 \frac{\Delta y_{k+1} + \Delta y_k}{\Delta y_{k+1}} \left( Q_{j,k+1/2} \frac{\delta_y^+ (\Delta U^{n+1})_{j,k}}{\Delta y_{k+1}} - Q_{j,k-1/2} \frac{\delta_y^- (\Delta U^{n+1})_{j,k}}{\Delta y_{k}} \right),
\]
where
\[
\Delta x_j = x_j - x_{j-1}, \quad j = 1, 2, ..., M_x - 1,
\]
\[
\Delta y_k = y_k - y_{k-1}, \quad k = 1, 2, ..., M_y - 1,
\]
are the mesh intervals, and
\[
\delta_x^+ \Delta U_{j,k} = \Delta U_{j+1,k} - \Delta U_{j,k},
\]
\[
\delta_y^- \Delta U_{j,k} = \Delta U_{j,k} - \Delta U_{j-1,k},
\]
\[
\delta_y^+ \Delta U_{j,k} = \Delta U_{j,k} - \Delta U_{j,k+1},
\]
\[
\delta_y^- \Delta U_{j,k} = \Delta U_{j,k} - \Delta U_{j,k-1},
\]
are operators introduced for convenience.

Special care has to be taken for the discretization of the convective terms. For example, a second order centered difference scheme can easily cause numerical oscillations. Smooke (1982) has reported convergence difficulties in the modeling of premixed flames with the centered scheme, as opposed to upwind schemes. An upwind scheme is a backward or forward difference scheme (depending on the signs of the velocities). Let \( vs = 1 \) if \( v_{j,k} \geq 0 \) and \( us = 1 \) if \( u_{j,k} > 0 \), otherwise zero, then the upwind difference expressions for the convective terms become
\[
\frac{\partial}{\partial x} (A \Delta U^{n+1})_{j,k} = us \frac{\delta_x^- (A \Delta U^{n+1})_{j,k}}{\Delta x_{j+1}} + (1 - us) \frac{\delta_x^+ (A \Delta U^{n+1})_{j,k}}{\Delta x_{j+1}},
\]
\[
\frac{\partial}{\partial y} (B \Delta U^{n+1})_{j,k} = vs \frac{\delta_y^- (B \Delta U^{n+1})_{j,k}}{\Delta y_{k+1}} + (1 - vs) \frac{\delta_y^+ (B \Delta U^{n+1})_{j,k}}{\Delta y_{k+1}}.
\]
The first order accuracy for the convective terms introduces numerical diffusivity, thus improving convergence characteristics of the numerical scheme. The numerical diffusion
is proportional to $\rho u \Delta x_j$ and $\rho v \Delta y_k$, and if comparable to the actual diffusion rates, $\lambda / c_p$ for heat, and $\rho D$ for mass, a broadening of the solution profiles can be expected.

Furthermore, since the numerical diffusion is proportional to the grid spacing, the effect of numerical diffusivity is exacerbated on coarser meshes. This effect is advantageous when the solution method utilizes mesh refinements, since it is easier to obtain converged solutions with large diffusivity. Once a solution on a coarse mesh has been obtained, this solution is used as an initial estimate on a finer mesh. When the mesh spacing is reduced, high gradients can be better resolved.

As an alternative to the first order upwind scheme a centered scheme plus a second order dissipative term can be used. The two schemes are equivalent but higher order dissipative terms can be easily added with the centered scheme without changing the structure of the code (Pulliam, 1986).

Equations (4.29) and (4.30) can be simplified by introducing definitions,

$$a_{j,k} = \frac{2Q_{j+1/2,k}}{(\Delta x_{j+1} + \Delta x_j)\Delta x_{j+1}}, \quad (4.34)$$

$$b_{j,k} = \frac{2Q_{j-1/2,k}}{(\Delta x_{j+1} + \Delta x_j)\Delta x_j}, \quad (4.35)$$

$$c_{j,k} = \frac{2Q_{j,k+1/2}}{(\Delta y_{k+1} + \Delta y_k)\Delta y_{k+1}}, \quad (4.36)$$

$$d_{j,k} = \frac{2Q_{j,k-1/2}}{(\Delta y_{k+1} + \Delta y_k)\Delta y_k}. \quad (4.37)$$

Sweep in the $x$-direction for every interior horizontal grid line when solving for $\Delta U^*$

$$\left[I + \Delta t \left( -us \frac{\delta^- A_{j,k}}{\Delta x_j} - (1-us) \frac{\delta^+ A_{j,k}}{\Delta x_{j+1}} - a_{j,k}\delta^+ \delta^- + b_{j,k} \delta^- + D_{j,k} \right) \right] \Delta U^*_{j,k} = \Delta t \text{RHS}^*_{j,k}. \quad (4.38)$$

By collecting terms it is possible to write for an interior grid point $(x_j, y_k)$

$$R_{j-1,k} \Delta U^*_{j-1,k} + S_{j,k} \Delta U^*_{j,k} + T_{j,k} \Delta U^*_{j+1,k} = \Delta t \text{RHS}^*_{j,k}. \quad (4.39)$$
where

\[
R_{j-1,k} = \Delta t \left\{ \frac{u s}{\Delta x_j} A_{j-1,k} - b_{j,k} \right\},
\]

(4.40)

\[
S_{j,k} = I + \Delta t \left\{ -u s \frac{A_{j,k}}{\Delta x_j} + (1 - u s) \frac{A_{j,k}}{\Delta x_{j+1}} + a_{j,k} + b_{j,k} + D_{j,k} \right\},
\]

(4.41)

\[
T_{j,k} = \Delta t \left\{ -(1 - u s) \frac{A_{j+1,k}}{\Delta x_{j+1}} - a_{j,k} \right\}.
\]

(4.42)

Equation (4.39) can be expressed in a matrix form as a block tridiagonal system:

\[
\begin{bmatrix}
S_{1,1} & T_{1,1} & & & & \\
R_{1,1} & S_{2,1} & T_{2,1} & & & \\
& \ddots & \ddots & \ddots & & \\
& & & R_{M,-3,1} & S_{M,-2,1} & 0 \\
& & & 0 & S_{t,2} & T_{t,2} \\
& & & & \cdots & \cdots \\
& & & & & \cdots \\
& & & & & \cdots \\
R_{j-1,k} & S_{j,k} & T_{j,k} & & & & \\
& \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
& & & & & & \cdots \\
& & & & & & \cdots \\
& & & & & & \cdots \\
& & & & & & \cdots \\
R_{M,-3,M,-2} & S_{M,-2,M,-2} & & & & \\
\end{bmatrix}
\begin{bmatrix}
\Delta U'_{j,k}
\end{bmatrix} = \Delta t \begin{bmatrix}
\text{RHS}_{j,k}
\end{bmatrix},
\]

(4.43)

where \(0\) is the 3×3 null matrix. \(R_{j-1,k}, S_{j,k}\) and \(T_{j,k}\) are 3×3 sub-matrices.

Similarly, when solving for \(\Delta U'^{n+1}\), an interior grid point \((x_j, y_k)\) can be expressed as,

\[
\left[ I + \Delta t \left\{ \frac{\delta^- B_{j,k}}{\Delta y_k} + (1 - v s) \frac{\delta^+ B_{j,k}}{\Delta y_{k+1}} - c_{j,k} \delta^+ \gamma + d_{j,k} \delta^- \gamma \right\} \right] \Delta U'^{n+1}_{j,k} = \Delta U'^{*}_{j,k}.
\]

(4.44)

Sweeping in the y-direction yields another block tridiagonal system,
\[
\begin{bmatrix}
SS_{l,1} & TT_{l,1} \\
RR_{l,1} & SS_{l,2} & TT_{l,2} \\
\vvec{ } \ & \vvec{ } \ & \vvec{ } \\
\vvec{ } & \vvec{ } \ & \vvec{ } \\
RR_{M_j-3} & SS_{M_j-2} & 0 \\
0 & SS_{l,1} & TT_{l,1} \\
\vvec{ } & \vvec{ } \ & \vvec{ } \\
\vvec{ } & \vvec{ } \ & \vvec{ } \\
RR_{j,k-1} & SS_{j,k} & TT_{j,k} \\
\vvec{ } & \vvec{ } \ & \vvec{ } \\
\vvec{ } & \vvec{ } \ & \vvec{ } \\
\vvec{ } & \vvec{ } \ & \vvec{ } \\
RR_{M_j-2M_j-3} & SS_{M_j-2M_j-2} & \vvec{ }
\end{bmatrix} \begin{bmatrix}
U_{j,k}
\end{bmatrix} = \Delta U_{j,k}, \quad (4.45)
\]

where,

\[
RR_{j,k-1} = \Delta t \left\{-d_{j,k} - vs \frac{B_{j,k-1}}{\Delta y_k}\right\}, \quad (4.46)
\]

\[
SS_{j,k} = I + \Delta t \left\{c_{j,k} + d_{j,k} + vs \frac{B_{j,k}}{\Delta y_k} - (1 - vs) \frac{B_{j,k}}{\Delta y_{k+1}}\right\}, \quad (4.47)
\]

\[
TT_{j,k} = \Delta t \left\{-c_{j,k} + (1 - vs) \frac{B_{j,k+1}}{\Delta y_{k+1}}\right\}, \quad (4.48)
\]

are 3×3 sub-matrices.

The computational cost for the block Thomas algorithm is \(O(EN^3)\), where \(E\) denotes the number of interior grid points, which makes it fairly efficient for large systems where \(N \geq 1\). To achieve good computational efficiency a solver is constructed that takes advantage of the sparsity pattern of the matrix to be inverted. For the flame sheet the off diagonal matrices \(R_{j-1,k}\) and \(RR_{j,1,k}\) are diagonal, hence block Thomas can be modified to handle one vector and two matrices for each sweep. Having a vector instead of a matrix hinders implementation of a second order accurate vorticity boundary condition at the wall, since \(R_{j-1,k}\) would then have off diagonal elements.

In Eqs. (4.6) and (4.7) the velocities are used in the convective terms at the explicit time level \(n\), instead of the gradients of the stream function which can be evaluated at the implicit time level \(n+1\). This is done to eliminate mixed derivatives such as,
\[
\frac{\partial}{\partial x} (0, \frac{\partial \psi}{\partial y}, \frac{\partial \psi}{\partial y} Z)^T,
\]

which cannot be split. The convergence rate is not severely affected by this restriction (Rubin and Khosla, 1981).

4.4 Boundary conditions on a pseudotransient formulation

In a pseudotransient formulation, Dirichlet boundary conditions that are not time dependent generate zero boundary values. This is an advantage, because the inflow boundary conditions can be modified without changing the structure of the code. Neuman boundary conditions are also easily implemented, since the algorithm suggested above first solves for interior points and then updates the boundary values. If a fully implicit method is used the approximate factorization becomes slightly different (Fletcher, 1991). Neuman boundary conditions are implemented with second order accuracy, except for the vorticity boundary condition, where a first order implementation is done.

4.5 Iterations within time step

An iterative procedure at each physical time level (inner-iteration), serves to eliminate linearization and approximate factorization errors. Moreover, the velocities in Eqs. (4.6) and (4.7) and the boundary values are evaluated at the explicit time level making inner-iterations important. Since only the steady state solution is of interest the number of time steps to convergence should be minimized. One is then faced with either making a large number of time steps with rapid inner-iteration convergence or taking few outer time steps with slow inner-iteration convergence. In the numerical code the number of inner-iterations are provided as user specified input. Alternatively, one can perform inner-
iterations until a convergence criteria is satisfied either for each sweep or after both sweeps (Kansa, 1981).

4.6 Freezing and numerical evaluation of Jacobians

From detailed numerical computations in premixed and counterflow diffusion flames it can be concluded that over 90% of the total CPU time is used for forming and factorizing the Jacobian matrices (Smooke et al., 1985). Thus to reduce computation time the Jacobians can be formed numerically, and factorized only periodically.

It is possible to “freeze” a Jacobian for a number of time steps, and hence it needs to be updated periodically. This can be done by storing the factorization of the Jacobian and performing inexpensive back substitutions. However, more iterations are usually required, typically, to obtain converged solution. To determine if a Jacobian needs to be factorized or not depends on the convergence rate. If the rate is too slow, a new Jacobian needs to be formed and factorized, if the rate of convergence is acceptable an “old” Jacobian can be used. For Newton’s method, a criteria can be derived based on the quadratic convergence rate that estimates if the convergence rate is acceptable (Smooke, 1983). A mathematical criteria for the approximate factorization method does not exist, and most methods use a predetermined number of time steps. Hager and Lee (1995) have reported a slight benefit for ADI-based schemes with Jacobian freezing. An alternative to freezing the Jacobian (in a pseudotransient formulation) is to increase the time step.

For problems involving detailed transport and complex chemistry, it is more efficient to evaluate the Jacobian matrices numerically as opposed to analytically. For a flame sheet solution it is possible, but very tedious, to derive analytical expressions for the elements in the Jacobian matrices. For a finite rate chemistry computation a numerical evaluation is almost the only choice. The elements of the Jacobian matrices can be formed by approximating the partial derivatives with a finite difference formula:
\[ J_{j,k}(U) = \frac{\partial F_i}{\partial U_k} = \lim_{\varepsilon \to 0} \frac{F_j(U_k + \varepsilon) - F_j(U_k)}{\varepsilon}, \] (4.50)

where \( \varepsilon = aU_k + b \). The constants \( a \) and \( b \) are chosen so that \( \varepsilon \) is small enough to ensure no loss of accuracy due to truncation errors, and large enough for round-off errors not to play a role. A good choice is to let \( a \) and \( b \) be the square root of the computer's unit round-off (Kee et al., 1980). On the computers where computations are performed the unit round-off is approximately \( 1 \times 10^{-16} \).

Since the Jacobians are banded (block tridiagonal) several columns of the Jacobians can be evaluated simultaneously, to minimize the number of function evaluations (Curtis et al., 1974, Curtis and Reid, 1974). First \( \mathbf{F} \) is evaluated at some vector \( \mathbf{U} \). Due to the block tridiagonal structure, every \( 3N \)-th element of \( \mathbf{U} \) can be perturbed beginning with the first, where \( N \) is the number of dependent variables. Hence, the "grouping" of columns is done in such way that no unknowns are in common. \( \mathbf{F} \) is evaluated at this new vector and the difference quotient in Eq. (4.50) is formed. The procedure is repeated until all the elements in \( \mathbf{F} \) have been perturbed.

### 4.7 Adaptive gridding

Due to the large difference between the scale of the burner and the flame thickness, where for example the temperature may vary a thousand degrees per millimeter, a highly non uniform mesh must be used. Equispaced meshes are prohibitively expensive for complicated one-dimensional combustion problems and practically all multidimensional problems even on the largest scientific computers (Smooke et al., 1985). One way to implement the non uniform mesh is to use a predetermined mesh. The problem with such an approach is that regions with high spatial activity are generally not known a priori. A numerical algorithm which places grid points in regions where they are needed, while
minimizing the total number of grid points to represent the solution accurately is needed. The adaptive gridding algorithm is illustrated in Figure 4.2.

![Adaptive gridding](image)

Figure 4.2. Illustration of adaptive gridding for arbitrary variable profile.

There are several techniques to choose optimal grid spacing. The most frequently used in the combustion field is to sub-equidistribute positive weight functions on a given interval (Kautsky and Nichols, 1980). Thereby, the value of the weight function is kept constant from one spatial interval to another. The weight functions can be chosen as a combination of first and second derivatives of the dependent variables (Smooke, 1982 and 1986). This can be seen as not allowing the slope and the curvature of the dependent variables to vary too much from one grid point to another. A solution is first obtained on a given mesh $\mathcal{M}_2$ on which adaptive gridding is performed. Let $U_i$ denote the $i$-th dependent variable. For each horizontal grid line if either one of the inequalities

$$
\int_{x_j}^{x_{j+1}} \left| \frac{\partial U_i}{\partial x} \right| dx \leq \delta_x \left| \max U_i - \min U_i \right|, \quad j = 1, 2, ..., M_x-2, \tag{4.51}
$$

$$
\int_{x_j}^{x_{j+1}} \left| \frac{\partial^2 U_i}{\partial x^2} \right| dx \leq \gamma_x \left| \max \frac{\partial U_i}{\partial x} - \min \frac{\partial U_i}{\partial x} \right|, \quad j = 1, 2, ..., M_x-2, \tag{4.52}
$$

is not satisfied a vertical grid line is inserted at the midpoint of the sub-interval in question. Similar tests are performed in the vertical direction. The constants $\delta_x$, $\gamma_x$, $\delta_y$, and $\gamma_y$ are all less than unity, and can be determined from user specified values which are lowered for each grid refinement level. Depending on the values of these constants a solution can be obtained to a desired level of accuracy. The maximum and the minimum values of the dependent variables and their first spatial derivatives are obtained from a solution on a previously determined mesh.
Once all the sub-intervals in both directions have been tested, if grid lines were inserted, the coarse mesh solution is interpolated linearly onto the new mesh. A linear interpolation is acceptable since the overall accuracy of the numerical scheme is of the first order. If a higher order scheme is used to preserve the accuracy of the solution, interpolation by cubic Hermite polynomials has to be used.

A disadvantage with the adaptive gridding procedure described above is that the mesh may not vary smoothly after a couple of grid refinements. The ratio of consecutive mesh intervals may differ by several orders of magnitude. It is therefore important to ensure that the ratio of adjacent mesh intervals are bounded above and below by constants (Smooke and Mattheij, 1985). This tends to smooth out rapid changes in the size of the mesh, that otherwise might cause numerical instabilities. A restriction,

\[
\frac{1}{R_x} \leq \frac{X_{j+1} - X_j}{X_j - X_{j-1}} \leq R_x, \quad j = 1, 2, ..., M_x - 2, \quad X = x, y, \quad (4.53)
\]

where \( R_x \geq 1 \), is imposed. Due to the properties of the weight functions chosen the practical implementation, as suggested by Kautsky and Nichols (1980), may vary slightly.

To reduce the computational time for simple chemistry computations, one can take advantage of the fact that the regions where the steepest gradients occur coincide for most variables in combustion problem. Therefore, it suffices to base the grid refinement, for example, on the first spatial derivatives of the temperature (De Lange and De Goey, 1994). For finite rate chemistry computations, where the peak values of different species appears at different spatial locations, all the \( N \) dependent variables have to be considered.

Similar to the technique of inserting grid lines in regions of high spatial activity, they can also be removed from regions with low spatial activity. This is particularly important when the problem under consideration is of transient nature, since the high activity regions can change from time step to time step. All grid lines except the initial ones can be removed, to avoid too drastic changes in the mesh.
5. Numerical code

The numerical code used to simulate the Wolfhard-Parker burner is written in the C++ programming language, while most existing combustion codes are written in FORTRAN. C++ is chosen since it provides support for object-oriented design and programming while retaining C's simplicity of expression, speed of execution and ability to dynamically allocate and release memory. This is important when adaptive gridding is applied, since the memory requirements change when the mesh is refined. Moreover, C++ will probably be the dominant programming language of the future, and its use is therefore justified.

The mesh $\mathcal{M}_2$ is mimicked as a two-dimensional linked list. A linked list is defined as a linear collection of self-referential class objects, called nodes, that are connected by pointers (Deitel and Deitel, 1985). A node can be seen as a grid point that contains certain information. List of data can be stored in arrays, but linked lists provide several advantages. A linked list is appropriate when the number of nodes to be used for an iteration is not known a priori. Linked lists are dynamic, so the length of a list can increase or decrease at runtime as necessary. When adaptive gridding is performed nodes are inserted (or removed) dynamically. The size of a “conventional” array, however, cannot be altered, because the array size is fixed at compile time. Insertion in a “conventional” array is time consuming since all the elements following the inserted element must be shifted appropriately. “Conventional” arrays can become full, whereas linked lists become full only when the system has insufficient memory to satisfy dynamic storage allocation requests.

The node is the fundamental building block of the mesh. The definition in Figure 5.1 defines a type ListNode. The copy constructor ListNode initializes the pointers to null, and dynamically allocates memory for an array of doubles that are being initialized. The private data members nextPtr and y_nextPtr point to objects of type ListNode, i.e. objects of the same type.
class ListNode {
friend class List;
public:
    ListNode(const double[], const int);
private:
    double pos[SIZE];
    ListNode *nextPtr;
    ListNode *y_nextPtr;
};

Figure 5.1. Definition of class ListNode.

Member nextPtr is a link that can "tie" an object of type ListNode to another object of the same type in the horizontal direction. Similarly, member y_nextPtr is a link in the vertical direction. The private member pos is an array of dimension SIZE, where for the flame sheet model SIZE = 16. Array pos contains information on location of grid point (i.e. x,y), dependent variables and other variables together with transport properties. The elements of the array are shown in Figure 5.2.

\[
\begin{array}{cccccccccccc}
   x & T & \rho & \mu & \psi & \omega & Z & Y_i & u & v & \rho D & y \\
 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\
\end{array}
\]

Figure 5.2. Elements of array pos.

The mesh is modeled as class List shown in Figure 5.3, which is a linked list of ListNode objects.
class List {
public:
List();
~List();
void insertAtBack( const double[], const int);
void y_insertAtBack( const double[], const int);
int insertInBetween( ListNode *, double[], const int);
int y_insertInBetween( ListNode *, double[], const int);
int removeInBetween(const int);
int y_removeInBetween(const int);
private:
ListNode *firstPtr;
ListNode *lastPtr;
ListNode *y_lastPtr;
ListNode *getNewNode(const double[], const int);
};

Figure 5.3. Definition of class List. Only a small portion of the member functions is shown.

The private data member firstPtr, points to the first node in the list, and the private data members lastPtr and y_lastPtr point to the last node on the first and last horizontal grid line, respectively. Hence, all nodes “know” about these pointers and if, for example, the outflow boundary needs to be post processed one can directly find the node to which y_lastPtr points to, and from there on use the nextPtr. With the nextPtr it is possible to step from one node to another in the horizontal direction. Note that it is not possible with this design to move “backwards” in the list, i.e. toward the inflow and the centerline, from a node somewhere in the middle of the list. To enable this the ListNode objects must have two extra pointers. In most functions used, only node information just next to the node under consideration is needed. Therefore, in these cases one can define utility pointers that “follow” after the node. A schematic illustration of the list is shown in Figure 5.4. The nodes at the wall and the outflow have a null pointer that indicates that the link does not point to another object in that specific direction.
The default constructor initializes the pointers to null. The destructor ensures that all \texttt{ListNode} objects are destroyed when the \texttt{List} object is destroyed. The private utility function \texttt{getNewNode} creates a node dynamically and returns a pointer to the newly created node. The public part of \texttt{List}, as described in detailed in appendix, provides member functions for insertion and deletion of grid points dynamically.

The drawback with the linked list is that the \texttt{ListNode} objects are not stored contiguously in memory, which is done for arrays. Thus, arrays allow immediate access to any array element because the address of any element can be calculated directly based on its position relative to the beginning of the array. Linked lists do not offer such immediate
access to their elements. Therefore, for computational efficiency many of the functions used to form and invert Jacobian matrices use "conventional" arrays.
6. Results and discussion

Flame sheet computations utilizing the numerical code described in chapter 5 were performed for laminar diffusion flames for methane and propane as fuels under different conditions. The same boundary conditions for the two oxidizer streams at the inflow were imposed since only single mode flames are studied in this work. Methane was chosen as fuel since it has a Lewis number close to unity which is essential for coupling the energy and species conservation equations as used for the flame sheet approximation. A comparison of numerical results for methane with experimental data from Smyth et al. (1985) and the analytic solution derived in chapter 3 are made. The effects of buoyancy on the numerical solution are investigated by performing a zero gravity computation. Compared to methane, propane is a more realistic fuel for a soot formation study. The Lewis number for propane, however, is approximately two, making the flame sheet model fairly inaccurate. The importance of unity Lewis number for the flame sheet approximation can be studied by a comparison with an infinite rate chemistry computation performed by Liu et al. (1993) in which non unity Lewis number effects were accounted for.

6.1 Computations with methane as fuel

The numerical model of the chemistry consists of a single step, infinitely fast irreversible chemical reaction,

\[ CH_4 + 2O_2 + 7.52N_2 \rightarrow CO_2 + 2H_2O + 7.52N_2. \]  

(6.1)

Reaction (6.1) does not represent realistic chemical reactions which usually involve many other chemical species (CO, NO, etc.). However, in the experiments of Smyth et al. (1985) the mole fractions of these species were rather small and their influence on temperature is minimal.
The boundary conditions for the inflow are chosen to match the experimental flow rates

\begin{align*}
0 < x < x_1: & \quad u = 0.0 \text{ cm/s}, \\
& \quad v = 9.7 \text{ cm/s}, \quad (6.2a)
\end{align*}

\begin{align*}
Y_{CH_4} = 1, \quad Y_i = 0, \quad i \neq CH_4, \quad (6.2b)
\end{align*}

\begin{align*}
T = 298 \text{ K}, \quad (6.2c)
\end{align*}

\begin{align*}
x_1 < x < x_*: & \quad u = 0.0 \text{ cm/s}, \quad (6.3a)
\end{align*}

\begin{align*}
v = 19.4 \text{ cm/s}, \quad (6.3b)
\end{align*}

\begin{align*}
Y_{O_2} = 0.232, \quad Y_{N_2} = 0.768, \quad Y_i = 0, \quad i \neq O_2, N_2, \quad (6.3c)
\end{align*}

\begin{align*}
T = 298 \text{ K}. \quad (6.3d)
\end{align*}

Physical and chemical parameters used in the numerical computation are shown in Table 1.

Table 1. Physical and chemical parameters used in the methane computations.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>(q/c_p)</td>
<td>32500</td>
<td>K</td>
</tr>
<tr>
<td>(R)</td>
<td>8.314</td>
<td>J/(mole K)</td>
</tr>
<tr>
<td>(\rho)</td>
<td>1.013\times10^5</td>
<td>Pa</td>
</tr>
<tr>
<td>(g)</td>
<td>9.81</td>
<td>m/s(^2)</td>
</tr>
<tr>
<td>((Pr)_{ref})</td>
<td>0.75</td>
<td>1</td>
</tr>
<tr>
<td>(W_F)</td>
<td>16</td>
<td>g/mole</td>
</tr>
<tr>
<td>(W_{OX})</td>
<td>32</td>
<td>g/mole</td>
</tr>
<tr>
<td>(W_{N_2})</td>
<td>28</td>
<td>g/mole</td>
</tr>
<tr>
<td>(W_{CO_2})</td>
<td>44</td>
<td>g/mole</td>
</tr>
<tr>
<td>(W_{H_2O})</td>
<td>18</td>
<td>g/mole</td>
</tr>
<tr>
<td>(v_F)</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>(v_{OX})</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>(v_{H_2O})</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>(v_{CO_2})</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>(x_1)</td>
<td>0.40</td>
<td>cm</td>
</tr>
<tr>
<td>(x_2)</td>
<td>1.60</td>
<td>cm</td>
</tr>
<tr>
<td>(x_*)</td>
<td>2.00</td>
<td>cm</td>
</tr>
<tr>
<td>(\delta_1)</td>
<td>2.00</td>
<td>cm</td>
</tr>
<tr>
<td>(\delta_2)</td>
<td>3.00</td>
<td>cm</td>
</tr>
<tr>
<td>(y_*)</td>
<td>40.0</td>
<td>cm</td>
</tr>
</tbody>
</table>
As an initial estimate for the numerical algorithm the dependent and independent variables are approximated with Gaussian profiles for an overventilated flame with a flame height of 25 cm. The initial estimate is not crucial since the radius of convergence of the pseudotransient, approximative factorization method is large.

The Reynolds number based on the coflowing oxidizer velocity

\[ \text{Re} = \frac{\rho_{ox} v_{ox}(x_w - x_1)}{\mu_{ox}} = 200, \]  

hence the flow can be considered to be fully laminar. Computations are performed until the relative Euclidean norm, \( E_2 \), defined as

\[ E_2 = \sqrt{\sum_{i=1}^{3M} [(U_i^{n+1} - U_i^n)^2 / (3M)] }, \]

where \( M \) is the number of interior grid points, is reduced below \( 5 \times 10^{-7} \).

**6.1.1 Mesh and adaptive gridding**

A solution is first obtained on a coarse mesh, typically 10×14. Grid points are then inserted adaptively. The grid refinement is based on the temperature and only the gradient is considered as criterion, since no spiked profiles had to be resolved for the flame sheet. The final non uniform mesh, shown in Figure 6.1, consists of 30×37 grid points that are clustered next to the inflow boundary and at the flame front.
Figure 6.1. Mesh used in the methane computation. Solid line indicates the location of the computed flame front.

Smooke (1982) defined the flame zone as the region in which

$$T_w + 0.1 \Delta T_w \leq T \leq T_w + 0.9 \Delta T_w,$$

(6.6)

where

$$\Delta T_w = T_{max} - T_w,$$

(6.7)

and $T_{max}$ is the peak temperature. With this definition 24 percent of the grid points are placed in the flame zone. The ratio of the largest to the smallest mesh interval in the horizontal direction is

$$\frac{\Delta x_{max}}{\Delta x_{min}} = 15.2,$$

(6.8)

and in the vertical direction
\[
\frac{\Delta y_{\text{max}}}{\Delta y_{\text{min}}} = 20.5.
\] (6.9)

Thus, the mesh is highly non uniform. The temperature does not vary more than 275 K in the horizontal direction from one grid point to another. In the vertical direction the flame sheet formulation of the problem always gives the peak temperature on the horizontal grid line above the inflow. Neglecting the steep gradient at the inflow, the temperature difference between adjacent grid points in the vertical direction is less than 200 K. To obtain the same accuracy on an equispaced mesh over the entire domain, 111 and 267 grid points will have to be used in the horizontal and vertical directions, respectively. Hence, a big saving on memory and a reduction of computational time are achieved by applying adaptive gridding.

6.1.2 Comparison with experiment

To enable calibration of the numerical model adopted in this study, it is imperative to compare the results from numerical simulations with experimental measurements. However, since no experimental data is available from the study conducted in conjunction with the modeling study reported in this thesis, the experiments of Smyth et al. (1985) are used for that purpose. These experiments were conducted on a Wolfhard-Parker burner with a central fuel slot 0.8 cm wide and 4.1 cm long for methane as fuel. The fuel slot had two adjacent rectangular slots, each 1.6 cm wide, through which oxidizer was supplied. A stabilizing screen anchored the flame at a height of 4.5 cm, and measurements were therefore only carried out up to 2.1 cm. The temperature was measured using uncoated 125 \( \mu \)m diameter Pt/Pt-10\% Rh fine wire thermocouples. Uniform inflow conditions were established with glass beads that filled the fuel and oxidizer slots, thus the assumed velocity profile at the inflow should be a good approximation. The velocities were measured by laser Doppler velocimetry and thermophoresis effects were reported small
(≤1.5 cm/s). Major species were sampled with quartz microprobes and then analyzed with a gas chromatograph.

Horizontal profiles for the temperature and the velocities were reported at three different axial locations: 0.3 cm, 0.9 cm and 1.5 cm. Mass fractions were only reported at a location of 0.9 cm, since this was considered a convenient region in which to analyze hydrocarbon growth chemistry. At this axial location significant amount of soot had not yet formed. Since radiation losses have not been accounted for in the numerical model, a comparison is therefore only made at a height of 0.9 cm above the inflow boundary.

In Figure 6.2 experimental and computational temperature profiles are compared. Uncorrected thermocouple values were reported from the experiments, since radiation losses were not taken into account. The experimental temperature data are therefore about 130 K too low in the region of highest temperature.

![Graph of temperature vs distance from centerline](image)

**Figure 6.2.** Temperatures at a height of 0.9 cm above the inflow.

The location of the flame front is predicted within 0.075 cm and occurs just on the oxidizer side. This shift is caused by finite reaction rates as well as the difficulty in
specifying the temperature exponent $r$ in Eq. (3.47), which has a tendency to move the flame laterally (Keyes and Smooke, 1987). In Table 2 the location of the peak experimental and computational temperatures are shown as a function of height above the burner.

Table 2. Location of peak experimental and computational temperatures at three different axial locations.

<table>
<thead>
<tr>
<th>Height above burner (cm)</th>
<th>Experimental location of $T_{\text{max}}$ (cm)</th>
<th>Computational location of $T_{\text{max}}$ (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>0.60</td>
<td>0.656</td>
</tr>
<tr>
<td>0.9</td>
<td>0.66</td>
<td>0.735</td>
</tr>
<tr>
<td>1.5</td>
<td>0.70</td>
<td>0.74</td>
</tr>
</tbody>
</table>

Excellent agreement is observed on the oxidizer side where the flame sheet is a good assumption. The temperature is overpredicted on the fuel side, mainly due to finite rate chemistry effects as energy-absorbing fuel pyrolysis reactions. The largest temperature gradient occur on the oxidizer side close to the flame front and is approximately 750 K/mm.

Figure 6.3 compares mole fractions of reactants in the flame. Clearly, from Figure 6.4 the methane and oxygen diffuse toward the flame front where they are being consumed. In the flame sheet model the only region for consumption of reactants is at the fuel-oxygen interface. The underprediction of methane concentration at the centerline is caused by the overprediction of the temperature in this region. The mole fractions of water and carbon dioxide shown in Figure 6.5 peak at the flame front. Good agreement is observed between numerical and experimental maximum values.
Figure 6.3. Mole fractions of methane and oxygen at a height of 0.9 cm above the inflow.

Figure 6.4. Mole fractions of carbon dioxide and water at a height of 0.9 cm above the inflow.
Figure 6.5. Vertical velocities at a height of 0.9 cm above the inflow.

In Figure 6.5, the computed vertical velocity profile shows higher values than the experimental at the centerline. The temperature is overpredicted in this region, which causes the gases to speed up to conserve momentum. Next to the flame front and out towards the wall excellent agreement is found. The horizontal velocity, compared in Figure 6.6, indicates strong acceleration in towards the centerline. The computed peak horizontal velocity appears at the computed flame front location, as a result of gas expansion.
Figure 6.6. Horizontal velocities at a height of 0.9 cm above the inflow.

Figure 6.7 shows the plot of the mixture fraction, $Z$, in the computational domain as a function of the horizontal and vertical coordinates. The value of $Z$ in the fuel stream decreases rapidly immediately above the inlet, mainly due to diffusion and convection of nitrogen and products in towards the centerline. The location of the flame front (denoted by the solid line in Figure 6.7) is given by the stoichiometric value of the mixture fraction, which can be shown to be $Z_f = 0.05482$ for methane-air mixtures.

The presence of the flame near the inflow boundary causes an expansion of the gases. This phenomena results in an initial expansion of the stream function observed in Figure 6.8, which shows the variation of the stream function in the burner. The initial expansion is later followed by a compression (acceleration) of the flow further up in the burner due to buoyancy. Regions of higher values than the wall value occur approximately 2 to 5 cm above the burner inlet next to the wall, indicating small recirculation zones.
Figure 6.9 shows the variation of the vorticity in the region of interest. Just above the inflow boundary at the interface between the fuel and the oxidizer, the vorticity takes on a negative value (counter clockwise rotation), due to the larger oxidizer velocity. The negative vorticity region is limited to the area directly above the inflow. At the flame front two large vorticity cells can be observed. These are the regions of highest clockwise rotation in the flame and is a result of the strong acceleration of the gases toward the centerline. Similar computational results for the vorticity have previously been observed by Smooke et al. (1989) for the axisymmetric geometry. The no slip boundary condition induces high vorticity at the wall, with a maximum value of 165.

The variation of the temperature in the burner is shown in Figure 6.10. As expected, the temperature field shows high gradients directly above the inflow boundary. In the flame sheet model the fuel and the oxidizer first meet in stoichiometric proportions in the vicinity of this boundary. This is not realistic physically, where finite rate chemistry computations performed by Smooke et al. (1989) have shown that there exists a region just above the inlet where the temperature is too low for combustion, resulting in a leakage of oxygen and radicals (in particular OH) into the fuel zone.

The velocity components and the streamlines are shown in Figures 6.11-6.13. The vertical velocity increases at the centerline with height above the inflow as discussed above. This strong acceleration leads to significant entrainment of nitrogen and combustion products, as can be seen from Table 3.

<table>
<thead>
<tr>
<th>Height above burner (cm)</th>
<th>Computed centerline velocity (m/s)</th>
<th>Computed peak negative horizontal velocity (m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>0.467</td>
<td>-0.126</td>
</tr>
<tr>
<td>0.9</td>
<td>0.715</td>
<td>-0.220</td>
</tr>
<tr>
<td>1.5</td>
<td>0.977</td>
<td>-0.229</td>
</tr>
</tbody>
</table>

From Figure 6.13 it is clear that the streamlines exhibit trajectories which originate in the lean region of the flame (on the oxidizer side), cross the high temperature reaction zone,
Figure 6.7. Mixture fraction field.

Figure 6.8. Stream function field.
Figure 6.9. Vorticity field.

Figure 6.10. Temperature field.
Figure 6.11. Vertical velocity field.

Figure 6.12. Horizontal velocity field.
and continue into the fuel-rich region. Smyth et al. (1985) and Kent and Wagner (1982) reported similar results with rapid acceleration of the vertical velocity along the centerline, strong entrainment form the air side, and buoyant flow conditions.

![Diagram of streamlines](image)

**Figure 6.13.** Streamlines in the Wolfhard-Parker burner for methane as fuel.

In Smyth et al. (1985) a visible overventilated flame height of 33 cm was observed. Despite the fact that the Lewis number is close to unity for methane, hence making the flame sheet an accurate approximation, an underventilated flame is computed. There are several reasons behind this discrepancy. First, the inflow boundary conditions are slightly different in the experiments compared to those used in Eqs. (6.2) and (6.3). The computational air composition does not include argon, which accounts for the value of $Y_{N_2} = 0.768$ used in the computation, compared to $Y_{N_2} = 0.7552$ for the experimental
analysis. The burner preheats the inflowing gases in the experiments to 328K. Furthermore, the inflowing velocities for fuel and oxidizer have been re-calibrated in a later paper by Norton et al. (1993) to 11.0 and 21.7 cm/s, respectively. Secondly, adiabatic wall conditions were not established in the experiments, and fixed wall temperature probably better resemble the experimental wall conditions. Finally, the effect of the screen that stabilizes the flame in the experiments is not known.

6.1.3 Buoyancy effects and comparison with analytic solution

To demonstrate the importance of buoyancy in low momentum flames, as the one under investigation, computations were performed with zero gravity. Such computations have previously been performed by Edelman et al. (1972) and Miller and Kee (1977) for the axisymmetric geometry. Figure 6.14 compares vertical centerline velocities with and without the effect of gravity. When buoyancy is included in the model a stronger acceleration occurs yielding a maximum velocity about 6 cm above the burner inlet. The non gravity flame exhibits a slow increase in velocity with a somewhat lower value at the outflow.
Figure 6.14. Comparison of vertical velocity at the centerline for buoyant and non-buoyant flames.

In Figure 6.15 buoyant and non-buoyant numerical flame shapes are compared with the analytical solution. The physical flame front is recovered from the dimensionless flame front by evaluating the viscosity at an average temperature of 1200 K. All solutions predict under ventilated flames. The agreement with the analytical solution is surprisingly good considering some of the drastic assumptions made in its the derivation. The larger flame heights for the numerical solutions are expected since vertical diffusion has been neglected in the analytical case. Roper (1977) reported that the Burke-Schumann approach applied on slot burners predicts the flame shape accurately for momentum controlled flames with a flame height above 6 cm, hence the numerical results presented here seems to be trustworthy. The non-buoyant numerical flame shape bow out farther from the centerline and are somewhat longer compared to the buoyant flame. That is, the buoyancy force accelerates the flow in the reaction zone, causes shearing, and thus
increases the air entrainment rate. This effect results in a more rapid depletion of the fuel explaining the shorter flame height, as previously observed by Roper (1977).

Figure 6.15. Comparison of flame shapes for methane as fuel.
6.2 Computation with propane as fuel

Infinite rate chemistry computations for a two-dimensional slot burner with propane as fuel have been performed by Liu et al. (1993). In their study non-unity Lewis number effects were accounted for, and gravity effects were neglected. To test the numerical model and to study non-unity Lewis number effects, computations are performed with inflow conditions and physical parameters similar to the one in the computation of Liu et al.. The only difference is the height of the computational domain where Liu et al. used 30 cm, whereas in the present computation a value of 40 cm is used to better allow the gradients in the vertical direction to approach zero at the outflow boundary. The scaled heat release parameter has a value of 32998 K corresponding to the adiabatic flame temperature of propane-air combustion. Other physical and chemical parameters are summarized in table 5.

The oxidation of propane (C$_3$H$_8$) can be described with the reaction

$$C_3H_8 + 5O_2 + 18.8N_2 \rightarrow 3CO_2 + 4H_2O + 18.8N_2.$$ \hfill (6.10)

The boundary conditions at the inflow are

$0 < x < x_1$:

$$u = 0.0 \text{ cm/s},$$ \hfill (6.11a)

$$v = 3.0 \text{ cm/s},$$ \hfill (6.11b)

$$Y_{C_3H_8} = 1, \quad Y_i = 0, \quad i \neq C_3H_8,$$ \hfill (6.11c)

$$T = 298 \text{ K},$$ \hfill (6.11d)

$x_1 < x < x_w$:

$$u = 0.0 \text{ cm/s},$$ \hfill (6.12a)

$$v = 9.88 \text{ cm/s},$$ \hfill (6.12b)

$$Y_{O_2} = 0.232, \quad Y_{N_2} = 0.768, \quad Y_i = 0, \quad i \neq O_2, N_2,$$ \hfill (6.12c)

$$T = 298 \text{ K}.$$ \hfill (6.12d)
Table 4. Physical and chemical parameters used in the propane computation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q/C_p$</td>
<td>32998</td>
<td>K</td>
</tr>
<tr>
<td>$R$</td>
<td>8.314</td>
<td>J/(mole K)</td>
</tr>
<tr>
<td>$p$</td>
<td>$1.013 \times 10^5$</td>
<td>Pa</td>
</tr>
<tr>
<td>$g$</td>
<td>0</td>
<td>m/s$^2$</td>
</tr>
<tr>
<td>$(Pr)_{ref}$</td>
<td>0.7</td>
<td>1</td>
</tr>
<tr>
<td>$W_F$</td>
<td>44</td>
<td>g/mole</td>
</tr>
<tr>
<td>$W_{OX}$</td>
<td>32</td>
<td>g/mole</td>
</tr>
<tr>
<td>$W_{N2}$</td>
<td>28</td>
<td>g/mole</td>
</tr>
<tr>
<td>$W_{CO2}$</td>
<td>44</td>
<td>g/mole</td>
</tr>
<tr>
<td>$W_{H2O}$</td>
<td>18</td>
<td>g/mole</td>
</tr>
<tr>
<td>$\nu_F$</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$\nu_{OX}$</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>$\nu_{H2O}$</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>$\nu_{CO2}$</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>$x_1$</td>
<td>0.318</td>
<td>cm</td>
</tr>
<tr>
<td>$x_2$</td>
<td>1.60</td>
<td>cm</td>
</tr>
<tr>
<td>$x_w$</td>
<td>2.54</td>
<td>cm</td>
</tr>
<tr>
<td>$\delta_1$</td>
<td>2.00</td>
<td>cm</td>
</tr>
<tr>
<td>$\delta_2$</td>
<td>5.40</td>
<td>cm</td>
</tr>
<tr>
<td>$\nu_w$</td>
<td>40.0</td>
<td>cm</td>
</tr>
</tbody>
</table>

The viscosity law used is the same as that Liu used

$$\mu = aT + b,$$

where $a = 0.002532 \times 10^{-5}$ and $b = 1.426 \times 10^{-5}$. The Reynolds number based on the coflowing oxidizer in the computation is approximately 120. A steady state solution is obtained with adaptive gridding on a 29×27 mesh, where the relative Euclidean norm is reduced below $5 \times 10^{-7}$. The adaptive gridding procedure is the same as that for methane, with similar mesh characteristics.
6.2.1 Comparison with analytic solution

Since gravity is neglected, buoyancy does not affect the solution. The analytic solution is obtained by evaluating the viscosity at an average temperature of 1288 K. The numerical flame shape compare favorably with the analytical as shown in Figure 6.15. In contrast to the methane computations, a smaller numerical flame height compared to analytical is predicted despite the absence of vertical diffusion in the analytic solution.

![Graph showing comparison between numerical and analytical flame shapes](image)

Figure 6.16. Comparison of numerical and analytical flame shapes for propane as fuel.
6.2.2 Comparison with computation

The variation of the dependent variables, the velocities and the temperature in the performed computation in the computational domain are shown in Figures 6.17-6.22. Even though numerical results are in quantitative agreement with the methane computation there are some interesting differences worth noticing. Most notably in the flow field. The initial expansion of the stream function shown in Figure 6.18 is less significant, and no recirculation zone is observed. The streamlines in Figure 6.23 clearly show less and delayed entrainment from the air side, as a result of the non gravity field imposed. Only a slight increase of vertical velocity at the centerline is observed, as can be seen in Figure 6.21 and Table 5. The vorticity field in Figure 6.19 has a region of counter clockwise rotation next to the inflow.

Table 5. Computed vertical velocity at the centerline and computed maximum inward horizontal velocity as a function of height above the burner.

<table>
<thead>
<tr>
<th>Height above burner (cm)</th>
<th>Computed centerline velocity (m/s)</th>
<th>Computed peak negative horizontal velocity (m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.385</td>
<td>0.240</td>
<td>-0.052</td>
</tr>
<tr>
<td>0.769</td>
<td>0.269</td>
<td>-0.018</td>
</tr>
<tr>
<td>1.54</td>
<td>0.315</td>
<td>-0.042</td>
</tr>
</tbody>
</table>
Figure 6.17. Mixture fraction field.

Figure 6.18. Stream function field.
Figure 6.19. Vorticity field.

Figure 6.20. Temperature field.
Figure 6.21. Vertical velocity field.

Figure 6.22. Horizontal velocity field.
Figure 6.23. Streamlines in the Wolfhard-Parker burner for propane as fuel.

Liu et al. (1993) obtained an overventilated flame with a flame height of 22 cm under almost identical boundary conditions. The discrepancy with present analytical and numerical results is caused by the non-unity Lewis number rendering the flame sheet model inaccurate. Since the Lewis number of propane is almost two the assumption of Lewis number unity, as made in the flame sheet, suppresses diffusion of heat and increases diffusion of fuel. The higher mass diffusion in the present computation makes the propane flame underventilated. Thus, it can be concluded that a Lewis number of unity is essential for the flame sheet approximation. To account for non-unity Lewis number effects one has to solve the energy equation separate from the species equations. The species equations can still be coupled and can be solved for a mixture fraction variable as done by Liu et al. (1993).
7. Conclusions and recommendations for future work

Flame sheet computations for methane and propane as fuels are capable of predicting thermal, concentration and fluid dynamical fields established in a Wolfhard-Parker burner at atmospheric pressure. In both cases considered in this work the resulting flames are underventilated, with flame shapes similar to analytic predictions. Previous investigators have reported overventilated flames under similar conditions, and reasons behind the different flame shapes have already been discussed in chapter 6.

The pseudotransient, approximative factorization method used to obtain steady state solutions requires modest memory. Moreover, it proved to be a very robust method. Even though overventilated flame shapes were used as initial estimates, underventilated flames were computed. However, the convergence rate is low, and to obtain flame sheet solutions with high accuracy more than hundred thousand sweeps in each direction must be performed. Large number of sweeps are required since small time steps must be taken to avoid stability problems, that are usually encountered at the inflow, despite the use of smoothing functions and iterations within time step. The lack of stability is thought to be caused by insufficient numerical diffusion when the mesh is refined at the inflow, thus seriously restricting the time step size.

There are several methods available to increase the rate of convergence and the accuracy of the solution. The most frequently used is to combine the present method with Newton’s method, which can be applied when the transient solution is close to steady state (Rice et al., 1988). Another useful technique to reduce the computational cost to reach steady state is to utilize multigrid methods. Liu et al. (1993) reported a two-order of magnitude decrease in CPU time for problems of modest complexity. The computational efficiency of multilevel methods can be further enhanced in a parallel environment where computations on different levels of grids are performed simultaneously by different processors. Dual time stepping is a method where a second time derivative is used for iterations within time step, which can be appropriately preconditioned to
maximize convergence (Venkateswaran and Merkle, 1995). The method is effective over a wide range of flow Mach numbers and physical time scales, and provides optimum convergence for low speed compressible flows, like in the case of diffusion flames, and shows significant savings in CPU time. Flux vector splitting might also be considered as a technique to increase the convergence rate (Kansa, 1981).

To produce a more accurate numerical algorithm, the adopted solver has to be modified to handle higher order accurate vorticity boundary conditions at the wall. The upwind-centered difference scheme used can then be replaced, for example, with a fourth order compact scheme. Maximum norms in the performed computations frequently occurred next to the wall for the vorticity, and an accurate representation in this region of the burner is therefore crucial. The vorticity boundary condition implemented in the numerical code is derived from a Taylor series expansion of the stream function at the wall. An interesting alternative is available in a pseudotransient formulation, where one guesses values for the boundary vorticity and start an iterative procedure which decreases the errors in the initial guess. The iterative procedure can be optimized by a careful choice of iteration parameters (Israeli, 1972).

The adaptive gridding algorithm implemented proved to be extremely efficient in adding grid points in areas with high spatial activity. The non uniform mesh spacing for the methane computation is such that 26 times more equispaced grid points are needed to obtain comparable accuracy. The savings in CPU time and memory requirements that result from applying adaptive gridding are what make two-dimensional combustion modeling feasible on work stations. To base the grid refinement on the first spatial derivative of the temperature is sufficient for flame sheet computations.

For future studies, it is recommended that the convergence characteristics be improved by applying any of the methods discussed above. The pseudotransient, approximative factorization method has potential to be efficiently implemented in parallel, which can reduce the CPU time drastically (Smooke and Giovangigli, 1992). Attempts were also made to vectorize the numerical code on a CRAY C90 at the San Diego National Super
Computer Center. Only a slight increase in computational efficiency was observed, since C++ codes (as well as C codes) do not fully vectorize. The vectorization is hampered by several factors, most notably that the C standard defines the behavior of a function even when pointers and/or arrays are aliased together and one or more have side effect changes. This requires a runtime dependency check on almost every vector loop, something FORTRAN codes rarely need. Even if runtime checks can be avoided full vectorization is difficult to achieve. Therefore, the focus should be on parallelization rather than vectorization.

Once the numerical algorithm is more computationally efficient, finite rate chemistry computations are possible without significantly changing the structure of the code. The dependent variable vector, \( U \), introduced in chapter 4, has to be modified to \( U = (\psi, \alpha, T, Y_1, Y_2, \ldots, Y_N)^T \). Hence, instead of solving for three dependent variables one now has to solve for \( N + 3 \) dependent variables. The same boundary conditions used for \( Z \) in the flame sheet model can be imposed on \( T \) and \( Y_i \).

The flame sheet model provides an excellent initial estimate for the finite rate chemistry computation. The minor species of interest can be approximated by Gaussian profiles centered at the flame front predicted by the flame sheet model. Their peak values can be estimated from experimental data. The full set of governing equations can be solved in a two step procedure (Keyes and Smooke, 1987). A solution is first obtained by imposing a fixed flame sheet temperature to facilitate convergence. This solution is then used as an initial estimate for the full fluid dynamic-thermochemistry model which includes the energy equation. Transport and thermodynamic properties necessary for the computations can be obtained from CHEMKIN (Kee et al., 1987). The solution procedure is illustrated in Figure 7.1.
To obtain finite rate chemistry computations under dual-mode conditions at high pressures, which is the ultimate goal of this project, the following steps are recommended:

- The initial finite rate chemistry computations should be performed on a single-mode flame for methane as fuel. Methane has only one carbon atom and thus behaves different from other hydrocarbons. For example, it does not soot as much, which hinder reliable soot measurements. However, there are two reasons for choosing methane as a fuel with which to start performing computations:
  1) It has a Lewis number close to unity, which is essential for coupling the energy and species conservation equations as used for the flame sheet approximation.
  2) The number of intermediate species participating in the reaction process is much less compared to other hydrocarbons, due to the simple structure of methane, thus reducing the computational time.

- When numerical results show good agreement with experiments for the methane-air case, ethane and propene are suggested as fuels at atmospheric pressure under single-
mode conditions. These fuels are appropriate for a soot formation study since they present two different routes for benzene formation, the acetene for ethane and the propargyl for propene (Miller and Melias, 1992, Seshadri et al., 1990). At this point it is very important that the numerical code is computationally very efficient, since the number of intermediate species are large.

- To simulate dual-mode conditions, the inflowing oxidizer can be vitiated by water and set to a higher temperature. Numerically, only a few parameters need to be changed in the input file to the numerical code compared to the single-mode simulations. Water is the only product of combustion of H2-air/O2, and its effect on soot formation is purely thermal. The effect of carbon dioxide in the oxidizer can therefore be eliminated (Du et al., 1991).

- Simulations at high pressures require that the gas phase chemistry is modeled with a finite-rate chemical mechanism applicable at high pressures. The mathematical formulation of the physics of particulate processes is rather simple but a numerical approach, where one has to consider simultaneous processes of nucleation, coagulation, sedimentation and surface reactions, is not yet computationally feasible (Frenklach and Harris, 1987). In order to overcome this difficulty, approximate methods have to be used. In the sectional model method the aerosol-size distribution is approximated with a finite number of sections with the properties within each section averaged (Warren and Seinfeld, 1985). The moments method is based on the solution of differential equations for the moments of the particle-size distribution function and does not require any prior knowledge of the distribution of this function (Frenklach and Harris, 1987). The two methods discussed above can be used to model the inception, coagulation, growth and oxidation of the soot particles.
• Experiments can be carried out up to a pressure of 10 bar. However, the pressure in many combustion devices, as a modern jet engine combustor, is in the range of 30 bars (Cohen et al., 1987). Extensive comparison with experiments has to be made, and when enough confidence with the numerical code is gained, simulations at pressures higher than 10 bars can be performed to elucidate the physical processes operative in practical combustion devices.
References


Deitel, H. M. and Deitel P. J., How To Program C++ (1994)


94


Appendix

Member functions for class List
Member functions for class List. \textit{nd} = \texttt{SIZE}.

\begin{verbatim}
// Insert a node at back of list in the horizontal direction.
void List::insertAtBack(const double pos[], const int nd)
{
    ListNode *newPtr = getNewNode(pos, nd);

    if (isEmpty()) //isEmpty checks if List is empty
        firstPtr = lastPtr = newPtr;
    else {
        lastPtr->nextPtr = newPtr;
        lastPtr = newPtr;
    }

    y_lastPtr = lastPtr;
}

// Insert a node at back of list in vertical direction.
void List::y_insertAtBack(const double pos[], const int nd)
{
    ListNode *newPtr = getNewNode(pos, nd);

    y_lastPtr->y_nextPtr = newPtr;
    y_lastPtr = newPtr;
}

//Insert a node somewhere in the middle of the list in the horizontal direction.
//cPtr is a pointer to the node “left” of the node to be inserted.
int List::insertInBetween(ListNode *cPtr, double interpolateA[], const int nd)
{
    int ierr = 0;

    ListNode *newPtr = getNewNode(interpolateA, nd);
    ListNode *tempPtr;

    tempPtr = cPtr->nextPtr;
    cPtr->nextPtr = newPtr;
    newPtr->nextPtr = tempPtr;

    return ierr;
}
\end{verbatim}
// Insert a node somewhere in the middle of the list in the vertical direction.
// cPtr is a pointer to the node "beneath" the node to be inserted.
int List::y_insertInBetween(ListNode *cPtr, double interpolateA[], const int nd)
{
    int ierr = 0;
    ListNode *newPtr = getNode(interpolateA, nd);
    ListNode *tempPtr;

    tempPtr = cPtr->y_nextPtr;
    cPtr->y_nextPtr = newPtr;
    newPtr->y_nextPtr = tempPtr;

    return ierr;
}

// Remove a vertical grid line with low spatial activity.
// col is the column to the "left" of the column to be removed.
int List::removeInBetween(const int col)
{
    ListNode *removePtr, *currentPtr = firstPtr;

    for( int i=0; i < col; i++ )
        currentPtr = currentPtr->nextPtr;

    while( currentPtr != 0 )
    {
        removePtr = currentPtr->nextPtr;
        currentPtr->nextPtr = removePtr->nextPtr;
        delete [] removePtr;
        currentPtr = currentPtr->y_nextPtr;
    }

    return 0;
}
// Remove a horizontal grid line with low spatial activity.
// rowA is the row “beneath” the row to be removed.
int List::y_removeInBetween(const int rowA)
{
    ListNode *removePtr, *y_currentPtr = firstPtr;

    for (int i=0; i < rowA; i++)
        y_currentPtr = y_currentPtr->y_nextPtr;

    while (y_currentPtr != 0)
    {
        removePtr = y_currentPtr->y_nextPtr;
        y_currentPtr->y_nextPtr = removePtr->y_nextPtr;
        delete [] removePtr;
        y_currentPtr = y_currentPtr->nextPtr;
    }

    return 0;
}
Vita

Henrik G. Johansson was born on July 23, 1970 in Karlstad, Sweden, to Margareta and Willy Johansson. During his three years at Nils Ericsonska gymnasium in Trollhattan, Sweden, he gradually gained interest for science and engineering, which encouraged him to pursue further education in the field of engineering. Henrik graduated from Nils Ericsonska in June 1989 and was celebrated by family and friends.

Henrik entered Lulea University in the north part of Sweden in August 1989 as a freshman mechanical engineering student. In 1992 he had a break in his studies when he did his military service as a technical staff sergeant in the Swedish Airforce. Highly motivated he returned to the mechanical engineering program in January 1993, specializing in applied mechanics. When the required course work was completed in June 1994, Henrik and his fellow students traveled to south-east Asia to celebrate the occasion.

To gain further knowledge and international experience Henrik entered the Master’s degree program at Virginia Polytechnic Institute and State University in August 1994. The author decided to work in the challenging field of combustion. Henrik received Master of Science Degrees in Mechanical Engineering from the above mentioned universities in July of 1996.

Henrik G. Johansson