Numerical analysis of steady one-dimensional flame near auto-ignition temperature using Cantera

A Report submitted

in partial fulfilment for the Degree of

Bachelor of Technology

in

Aerospace Engineering

by

Ardhendu Barman

(SC13B006)

pursued in

Department of Aerospace Engineering

Indian Institute of Space Science and Technology

to

INDIAN INSTITUTE OF SPACE SCIENCE AND TECHNOLOGY

THIRUVANANTHAPURAM

April 2017

CERTIFICATE

This is to certify that the project report entitled **"Numerical analysis of steady one-dimensional flame using Cantera"** submitted by **Ardhendu Barman (Student Code - SC13B006)**, to the Indian Institute of Space Science and Technology, Thiruvananthapuram, in partial fulfillment for the award of the degree of **Bachelor of Technology** in **Aerospace**, is a bonafide record of the project research work carried out by him under my supervision. The contents of this report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Salih.A
Head of the department
Department of Aerospace Engineering

Dr. Prathap C
Assistant Professor
Department of Aerospace Engineering

Dr. Mahesh S
Assistant Professor
Department of Aerospace Engineering

Place: IIST, Thiruvanathapuram
May 2017

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Place: IIST, Thiruvanathapuram
May 2017

Ardhendu Barman
SC13B006

Acknowledgements

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to Dr. Prathap C, Assistant professor and Dr. Mahesh S, Assistant professor for his guidance and constant supervision as well as for providing necessary information regarding the project and also for their support in completing the project. I would like to express my gratitude towards my parents and collegues of the institutes for their kind co-operation and encouragement which helped me in completion of this project.

I would like to express my special gratitude and thanks to all the members of IIST Thermal Lab for giving me such attention and time. My thanks and appreciation also goes to my colleagues who have willingly helped me out with his abilities and over-coming several up-hill tasks through out the internship.

Abstract

The objective of the work was to investigate the stabilization mechanism of flames developing in hot vitiated air environment and at elevated pressures like in staged combustion system. Real time combustors are highly turbulent in nature. To start with, the problem was analyzed at thermodynamic conditions which mimic the real time applications but at laminar conditions. In order to better understand the structure of turbulent flames at inlet temperature well above the auto-ignition temperature, the behavior of laminar flames at those conditions needed to be analyzed. In the present work, the laminar burning velocities and flame structures of premixed stoichiometric methane/air mixtures for inlet temperatures from 300 to 1450 K and absolute pressures from 1 to 8 bar had been calculated using a freely propagating laminar, one dimensional, planar flame model. In order to ensure an accurate converged and grid independent solution, the adaptive mesh criteria for the largest relative gradient and curvature were toughly selected as 0.04 and 0.07. The prediction showed that at inlet temperatures below the auto-ignition temperature, the predicted laminar burning velocity which corresponded to the unburned mixture velocity in order to create a steady laminar flame decreased with increase in pressure. When the inlet temperature of the mixture goes well beyond the auto-ignition temperature of the mixture, the unburned mixture velocity increased steeply at higher pressure level, with increase in pressure due to a complete transition of the flame structure. An important inference is that if the initial temperature is well above auto ignition temperature, then the flame did not need diffusion for self-sustainability and hence it propagates faster then temperatures less than auto-ignition temperature.

Table of contents

List of figures

Nomenclature

Greek Symbols

λ	thermal conductivity (W/mK)
ν	stoichiometric coefficient of a species in a reaction
ρ	mass density $(kg\ m^{-3})$
$\dot{\omega}$	molar rate of production $(mol/m^3 s)$
τ_{HR}	characteristic time scale of heat release (s)
ξ	ime integral contribution of a reaction to the formation of a species
ϕ	equivalence ratio
τ_{id}	ignition delay time
Λ	pressure eigenvalue

Other Symbols

h	specific enthalpy (J/kg)
p	pressure (Pa)
q	net rate of progress of a reaction $(mol/m^3 s)$
u	velocity of fluid mixture (m/s)
t	time (s)
A	cross sectional area (m^2)
$\dot{M}$	mass flow rate (kg/s)

$\dot{M}''$ mass flow rate per unit area ($kg/s/m^2$)

N_R, N_S number of considered reactions or species in a chemical kinetic scheme

$\bar{R}$ universal gas constant ($J/molK$)

S_L laminar burning velocity (m/s)

T temperature (K)

V_k diffusion velocity of the kth species (m/s)

W molar mass (kg/mol)

Y mass fraction

x, z 1D spatial coordinate (m)

j_k diffusive mass flux of species k

Acronyms / Abbreviations

PFR Plug Flow Reactor

NASA National Aeronautical and Space Administration

CSTR Continuous stirred-tank reactor

Chapter 1

Introduction

Staged combustion system in gas turbines looks promising for future power generation to increase the inlet temperature of turbine. Also, in the higher stages, low calorific value fuels can also be used which is feasible in stationary power engines (Prathap et al. (2012)). Recently, Prathap et al. (2012) reported that in an axial staged combustion system, they generated vitiated air in the first stage at the desired temperature. Then, they injected premixed methane/air mixture into the vitiated air in the second stage at a given equivalence ratio. Using exhaust gas probes, they measured the penetration depth. With increase in chamber pressure, they reported that the penetration depth decreased significantly even though all other parameters remain constant. The problem looks challenging as the second stage mixture was injected into cross flow stream of vitiated air at a temperature well above the self ignition temperature of the injected mixture. Unfortunately, at the ultra-high preheating temperatures of more than 1000 K experiments on laminar, premixed flame propagation can hardly be performed and therefore numerical simulations using detailed reaction schemes are often used. Habisreuther et al. (2013) investigated the same problem at laminar conditions using freely propagating planar flame model in PREMIX with GRI Mech3.0 mechanism. They reported that the change in the flame structure was responsible for the decrease in the penetration depth.

In the present work, results produced by Habisreuther et al. (2013) was revisited using open source software, CANTERA(Goodwin et al. (2017)) coupled with Python 3.5. Additionally, flame thickness was also estimated in the present work following the definition of Spalding (1979).

Chapter 2

INTRODUCTION TO CANTERA

Cantera is a suite of object-oriented software tools for problems involving chemical kinetics, thermodynamics, and/or transport processes. It provides types (or classes) of objects which represent phases of matter, interfaces between these phases, reaction managers, time dependent reactor networks, and steady one-dimensional reacting flows. Currently it is used for applications including combustion, detonations, electrochemical energy conversion and storage, fuel cells, batteries, aqueous electrolyte solutions, plasmas, and thin film deposition. It can be used from Python and Matlab, or in applications written in C++ and Fortran 90. In the present study CANTERA was used coupled with the Python environment to obtain the necessary results. The following sub-sections are mostly adapted from CANTERA documentation.

2.1 Thermodynamic Properties

The phase and $ideal\_interface$ entries of Cantera implement specific models for the thermodynamic properties as appropriate for the type of phase or interface represented by them. Although user may opt for different expressions to compute the properties, they all require thermodynamic property information for the individual species. At present to implement the phase types, the properties needed are:

1. the molar heat capacity at constant pressure $\hat{c}_p^0(T)$ for a range of temperatures and a reference pressure P_0 ;

2. the molar enthalpy $\hat{h}(T_0, P_0)$ at P_0 and a reference temperature T_0 ;

3. the absolute molar entropy $\hat{s}(T_0, P_0)$ at (T_0, P_0).

2.1.1 The NASA 7-Coefficient Polynomial Parameterization

The NASA 7-coefficient polynomial parameterization is used to compute the species reference-state thermodynamic properties $\hat{c}_p^0(T)$, $\hat{h}(T)$ and $\hat{s}(T)$. The NASA parameterization represents $\hat{c}_p^0(T)$ with a fourth-order polynomial:

$$\frac{c_p^0(T)}{R} = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^4$$
$$\frac{h^0(T)}{RT} = a_0 + \frac{a1}{2} T + \frac{a_2}{3} T^2 + \frac{a_3}{4} T^3 + \frac{a_4}{5} T^4 + \frac{a_5}{T}$$
$$\frac{s^0(T)}{R} = a_0 \ln T + a_1 T + \frac{a_2}{2} T^2 + \frac{a_3}{3} T^3 + \frac{a_4}{4} T^4 + a_6$$

This is the "old" NASA polynomial form, used in the original NASA equilibrium program and in Chemkin, which uses 7 coefficients in each of two temperature regions. It is not compatible with the form used in the most recent version of the NASA equilibrium program, which uses 9 coefficients for each temperature region.

2.2 Transport Properties

For most applications, gas mixture properties can be determined from pure species properties via approximate mixture averaging rules. However, there are some applications in which the approximate averaging rules are not sufficient. Cantera therefore addresses both the mixture-averaged approach and the full multicomponent approach to transport properties. The multicomponent methods are based on the work of Dixon-Lewis (1968) and the methods for mixture-averaged approach are reported in Peters and Warnatz (2013) and Kee et al. (1986).

2.2.1 The Pure-Species Fitting Procedure

To expedite the evaluation of transport properties in a computer program, such as the CANTERA, temperature dependent parts of the pure species property expressions are fitted. Now, rather than evaluating the complex expressions for the properties, only comparatively simple fits need to be evaluated.
A polynomial fit of the logarithm of the property versus the logarithm of the temperature is

used. For the viscosity:

$$ln\ \eta_k\ =\ \sum_{n=1}^{N} a_{n,k}(lnT)^{n-1} \tag{2.1}$$

and the thermal conductivity,

$$ln\ \lambda_k\ =\ \sum_{n=1}^{N} b_{n,k}(lnT)^{n-1} \tag{2.2}$$

The fits are done for each pair of binary diffusion coefficients in the system.

$$ln\ D_{jk}\ =\ \sum_{n=1}^{N} d_{n,jk}(lnT)^{n-1} \tag{2.3}$$

By default fits of third-order polynomial are used(i.e., N = 4) and the fitting errors are found well within one percent. The fitting procedure must be carried out for the particular system of gases that is present in a given problem. Therefore, the fitting cannot be done "once and for all," but must be done once at the beginning of each new problem.

2.2.2 The Mixture-Averaged Properties

The objective of this section is to determine mixture properties from the pure species properties. In the case of viscosity, we use the semi-empirical formula due to Wilke (1950) and modified by Bird et al. (1960). The Wilke formula for mixture viscosity is given by:

$$\eta = \sum_{k=1}^{K} \frac{X_k \eta_k}{\sum_{j=1}^{K} X_j \Phi_{kj}} \tag{2.4}$$

where

$$\Phi_{kj} = \frac{1}{\sqrt{8}} \left(1 + \frac{W_k}{W_j}\right)^{-1/2} \left(\left(1 + \frac{\eta_k}{\eta_j}\right)^{1/2} \left(\frac{\eta_k}{\eta_j}\right)^{1/4}\right)^2 \tag{2.5}$$

For the mixture-averaged thermal conductivity the combination averaging formula used:

$$\lambda = \frac{1}{2} \left(\sum_{j=1}^{K} X_k \lambda_k + \frac{1}{\sum_{j=1}^{K} X_k / \lambda_k}\right) \tag{2.6}$$

2.3 PROCESSING INPUT FILES

2.3.1 A Two-step Process

CANTERA imports the input file, and uses the information there to construct the object representing the phase or interface in the application. While this is the net effect, it is actually a two-step process. When a function like *importPhase* is called to import a phase definition from a file, a preprocessor runs automatically to read the input file and create a string that contains the same information but in an XML-based format called CTML. After the preprocessor finishes, Cantera imports the phase definition from this CTML data.

2.3.2 Two File Formats

There are two types input files available for Cantera. XML is a widely-used standard for data files, and it is designed to be relatively easy to parse. This makes it possible for other applications to use Cantera CTML data files, without requiring the substantial chemical knowledge that would be required to use .cti files. Supporting an XML-based data file format facilitates using Cantera in web services or other network computing applications.
The CTML version is much more verbose, and is much more tedious to write by hand, but is much easier to parse, since it is not necessary to write a custom parser—virtually any standard XML parser, which there are readily available, and can be used to read the CTML data.
Thus generally files that are easy for users to write are more difficult for machines to parse, because they make use of high-level application-specific information and conventions to simplify the notation. Conversely, files that are designed to be easily parsed are tedious to write because too much information has to be spelled out explicitly. A natural solution is to use two formats, one designed for writing by humans, the other for reading by machines, and provide a preprocessor to convert the human-friendly format to the machine-friendly one. For the present study GRI Mech3.0 mechanism in the .xml format was used as input file.

Chapter 3

One-Dimensional Flames

Cantera includes a set of models for representing steady-state, quasi-one-dimensional reacting flows, which can be used to simulate a number of common flames, such as:

- freely-propagating premixed laminar flames
- burner-stabilized premixed flames
- counterflow diffusion flames
- counterflow (strained) premixed flames

Additional capabilities include simulation of surface reactions, which can be used to represent processes such as combustion on a catalytic surface or chemical vapor deposition processes. All of these configurations are simulated using a common set of governing equations within a 1D "flow" domain, with the differences between the models being represented by differences in the boundary conditions applied. Here, we describe the governing equations and the various boundary conditions which can be applied.

3.1 Freely-propagating premixed laminar flow Governing Equations

The governing equations for a freely-propagating premixed laminar flow are as follow(assuming unit cross-sectional area):
Continuity:

$$\dot{M}'' = \rho u \tag{3.1}$$

Energy:

$$\rho c_p u \frac{\partial T}{\partial z} = \frac{\partial}{\partial z}\left(\lambda \frac{\partial T}{\partial z}\right) - \sum_k j_k c_{p,k} \frac{\partial T}{\partial z} - \sum_k h_k W_k \dot{\omega}_k \tag{3.2}$$

Species:

$$\rho u \frac{\partial Y_k}{\partial z} = -\frac{\partial j_k}{\partial z} + W_k \dot{\omega}_k \tag{3.3}$$

Equation of State:

$$\rho = \frac{p\bar{W}}{\bar{R}T} \tag{3.4}$$

3.2 Boundary Conditions

3.2.1 Inlet boundary

For a boundary located at a point $z = 0$ where there is an inflow, values are supplied for the temperature T_0, the species mass fractions $Y_{k,0}$ the scaled radial velocity V_0, and the mass flow rate $\dot{m}_0$(except in the case of the freely-propagating flame).

The following equations are solved at the point $z = 0$:

$$T(z=0) = T_0 \tag{3.5}$$

$$V(z=0) = V_0 \tag{3.6}$$

$$\dot{m}_0 Y_{k,0} - j_k(z=0) - \rho(z=0)u(z_0)Y_k(z=0) = 0 \tag{3.7}$$

If the mass flow rate is specified, we also solve:

$$\rho(z_0)u(z_0) = \dot{m}_0 \tag{3.8}$$

3.2.2 Outlet boundary

For a boundary located at a point $z = l$ where there is an outflow, we solve:

$$\left.\frac{\partial T}{\partial z}\right|_{z=l} = 0 \tag{3.9}$$

$$\left.\frac{\partial Y_k}{\partial z}\right|_{z=l} = 0 \tag{3.10}$$

3.3 Plug-Flow Reactor

To simulate ignition delay time, plug flow reactor model is used. A Plug-Flow Reactor (PFR) represents a steady-state channel with a cross-sectional area $\boldsymbol{A}$. Typically an ideal gas flows through it at a constant mass flow rate $\dot{m}$. Perpendicular to the flow direction, the gas is considered to be completely homogeneous. In the axial direction z, the states of the gas is allowed to change. However, all diffusion processes are neglected. Plug-Flow Reactors are often used to simulate ignition delay times, emission formation, and catalytic processes.
Even though this problem extends geometrically in one direction, it can be modeled via zero-dimensional reactors: Due to the neglecting of diffusion, downstream parts of the reactor have no influence on upstream parts. Therefore, PFRs can be modeled by marching from the beginning to the end of the reactor.
Unfortunately Cantera does not provide dedicated class to solve the PFR equations yet. However, there are two ways to simulate a PFR with the reactor elements previously presented. Both rely on the assumption that pressure is approximately constant throughout the Plug-Flow Reactor and that there is no friction. The momentum conservation equation is thus neglected.

3.3.1 PFR Modeling as a Series of CSTRs

For our study the following methodology to simulate our PFR model. The Plug-Flow Reactor is spatially discretized into a large number of axially distributed volumes. These volumes are modeled to be steady-state Continuously Stirred Tank Reactors(CSTR).
In Cantera, it is sufficient to consider a single reactor and march it forward in time, because there is no information traveling upstream. The mass flow rate $\dot{m}$ through the PFR enters the reactor from an upstream reservoir. For the first reactor, the reservoir conditions are the inflow boundary conditions of the PFR. By performing a time integration as described in Continuously Stirred Tank Reactor until the state of the reactor is converged, the steady-state CSTR solution is computed. The state of the CSTR is the inlet boundary condition for the next CSTR downstream.

Chapter 4

PROCEDURE

In the present study, the numerical simulations of steady-state, one-dimensional laminar premixed flames were performed using the CANTERA library integrated with Python 3.5. The code solves the one-dimensional, laminar, steady state equations of continuity, species and energy balance. It accounts for finite-rate chemical kinetics and multi-component molecular transport. The freely propagating planar flame model was used along with the mixture averaged transport properties evaluated using the CANTERA library. At the inlet or cold flow boundary the mole fractions and the temperature is specified. The solution was calculated on an adaptive mesh that took into account a relative gradient and curvature of the solution vector. In order to ensure an accurate converged and grid independent solution, the adaptive mesh criteria for the largest relative gradient and curvature were toughly selected as 0.04 and 0.07. The important objective of the present work was to estimate the contribution of each term in the energy equation to address the importance of diffusion energy transfer term. The equation which can be stated as:

Energy:

$$\rho c_p u \frac{\partial T}{\partial z} = \frac{\partial}{\partial z}\left(\lambda \frac{\partial T}{\partial z}\right) - \sum_k j_k c_{p,k} \frac{\partial T}{\partial z} - \sum_k h_k W_k \dot{\omega}_k \tag{4.1}$$

Equation of State:

$$\rho = \frac{p\bar{W}}{\bar{R}T} \tag{4.2}$$

The individual contribution of the different terms of the energy equation for a given set of operating condition is required in this work for the discussion of the results. The four terms of the energy equation, as to be observed from left to right, are: (1) heat convection, (2) heat conduction, (3) thermal diffusion and (4) heat release due to chemical reaction.

Chapter 5

RESULTS AND DISCUSSION

5.1 Validation

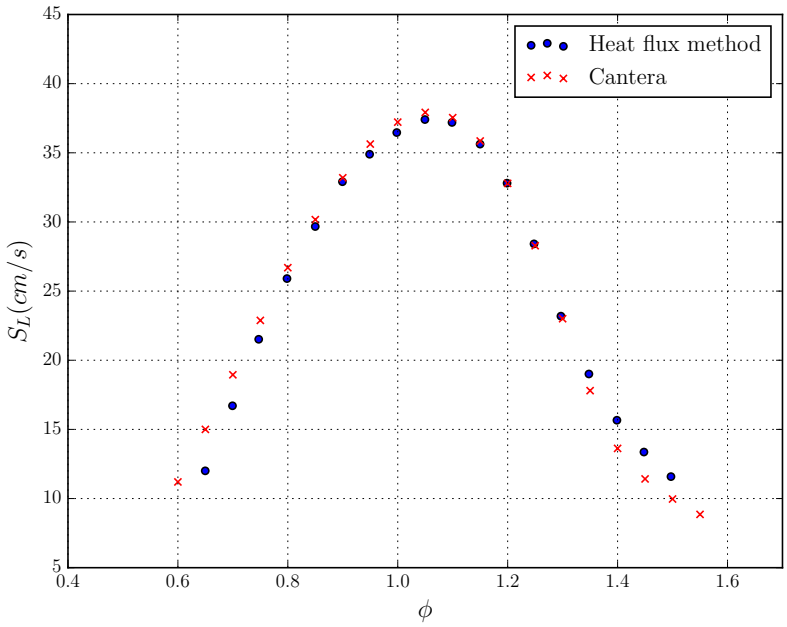


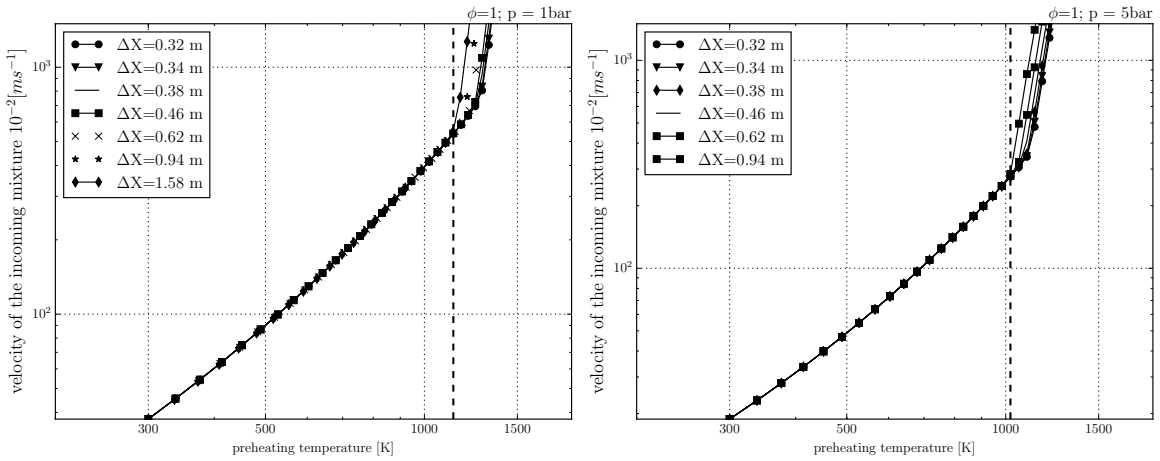
Fig. 5.1 Experimental result from literature of the adiabatic burning velocity of methane-air flames at ambient conditions. $p_0 = 1\ bar$ and $T_0 = 298K$.

It is essential to validate the present numerical procedure and the implemented computational schemes with measured data from literature. For this purpose the simulations are performed to calculate adiabatic burning velocity of methane air mixture as a function of equivalence ratio at 1 bar, 298K. The results were compared with results by Hermanns (2007) who measured using experimental methods (Heat flux method) in figure 5.1. The discrepancy

between the present simulations and that of literature data is small as shown in figure 5.1. This is due to the very fine resolution in the current simulations that was forced by the very strict convergence criteria used mentioned in chapter 4. With this validation, the other simulations were performed.

5.2 Velocity of incoming mixture or laminar burning velocity

As the objective of the present work was to understand the stabilization mechanism of premixed combustible mixture in hot vitiated air, initially, the premixed combustible methane air mixture was assumed to be preheated to the desired vitiated air temperature and then the possible mechanism of flame propagation was studied. Freely propagating planar flame model was used to perform simulations. Laminar burning velocity was estimated for different preheating temperatures. Domain length of simulations was varied from 0.32-1.58 m to alter the residence time to ensure that the simulated laminar burning velocity was independent of it. The reason for changing the residence time was to provide sufficient duration for the combustible mixture to auto ignite if the unburned gas temperature was sufficiently higher. Figure 5.2(a) shows the velocity of the incoming mixture or laminar burning velocity was plotted as a function of initial gas temperature for different lengths of computational domain at 1 bar. Till an initial temperature of 1200K, the simulated values of velocity of incoming mixture are independent of domain length. After that the velocity of incoming mixture started increasing steeply indicating the change of flame stabilization mechanism from flame propagation to auto ignition. For longer domain length or longer residence time, the auto ignition occurs at lower initial temperatures than that of the case for shorter residence time. Similar trend was reported in Smyth and Bryner (1997). Even though the auto ignition event was independent of diffusion transport mechanism, in the present freely propagating planar flame simulations, it clearly indicates that the presence of diffusion leads to a flame propagation event. Figure 5.2(b) shows the results at 5 bar. With increase in pressure for a given temperature, in the flame propagation domain, i.e., before the inflection point(the point at which maximum change in temperature occurs), laminar burning velocity decreased. In the auto ignition domain, i.e., after the inflection point, with increase in pressure the velocity of the incoming mixture increased because of decrease in the ignition delay time.



(a) (b)

Fig. 5.2 Predicted velocity of the incoming mixture, i.e., laminar burning velocity, of stoichiometric CH_4/air mixture as a function of initial temperature from GRI 3.0 mechanism for different calculation domains at 1 *bar* (left) and 5 *bar* (right). The vertical line shows the inflection point for the domain of $\Delta x = 0.62m$

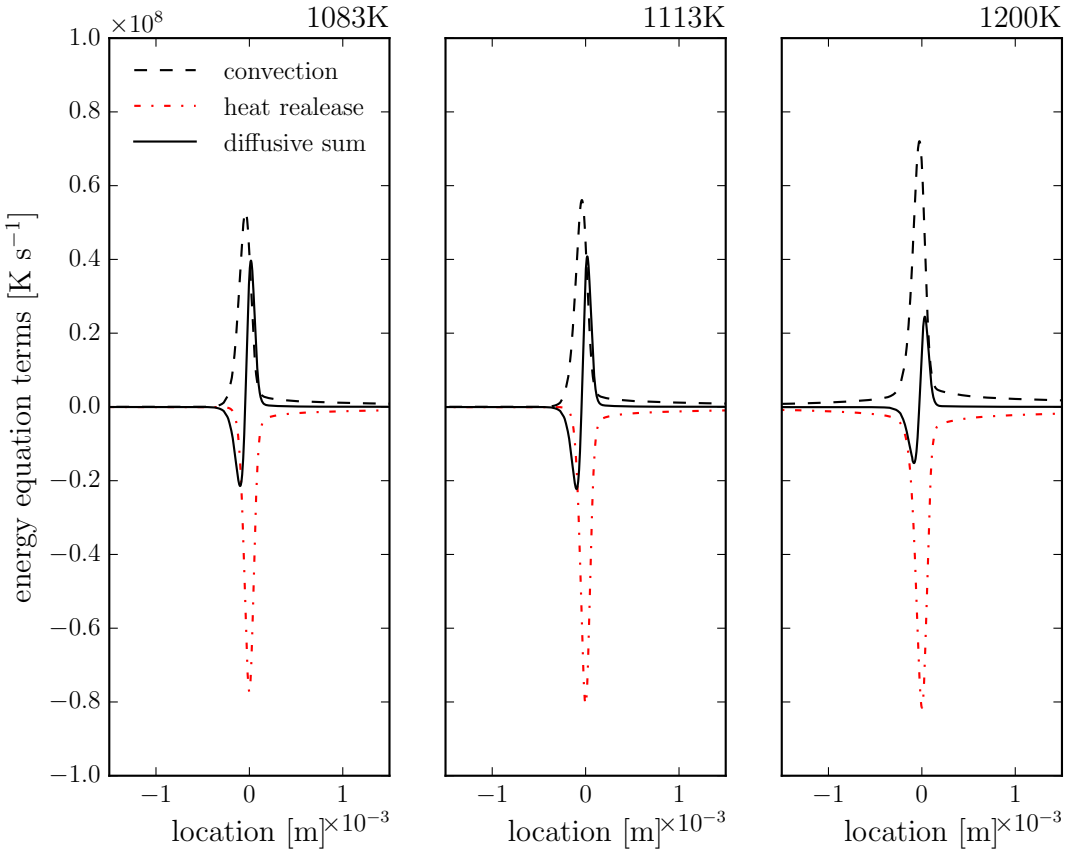


Fig. 5.3 Profiles of the terms of the energy equation for CH_4/air mixture, $\phi = 1.0$, p_0 = 1 bar, with unburned mixture temperatures T_0 of 1083, 1113 and 1200 K using the GRI 3.0 mechanism.

To reason out the occurrence of inflection point with respect to initial temperature for a given domain length and pressure, three different simulations were performed near the inflection point, for obtaining energy flux diagrams as shown in Fig. 5.3. As the temperature reaches 1200K for initial pressure of 1 bar and domain length of $\Delta x = 1.58m$, the diffusion energy term becomes negligible indicating the change of the flame stabilization from diffusion controlled to convection controlled.

5.3 Flame Structure

In order to understand the balance between heat release and transport mechanisms in the flame front, the variation of the magnitudes of the different terms in the energy equation within the flame structure of stoichiometric methane–air mixture at $T_0 = 298K$ and $p_0 = 5\ bar$ was plotted in Fig. 5.4.

1. The dashed line represents the energy transfer due to convection from the reaction zone to the unburned reactants in the preheat zone (first term in Eq. (4.1),i.e. $\rho c_p u \frac{\partial T}{\partial z}$)
2. The dash-dot-line represents the energy release due to chemical reaction (fourth term in Eq. (4.1),i.e., $\sum_k h_k W_k \dot{\omega}_k$)
3. The continuous line represents the energy transfer due to diffusion from reaction zone to the unburned reactants in the preheat zone (summation of second and third term in Eq. (4.1), i.e.,$\frac{\partial}{\partial z}\left(\lambda \frac{\partial T}{\partial z}\right) - \sum_k j_k c_{p,k} \frac{\partial T}{\partial z}$).

It is important to mention that all the terms was divided by the product of density and cross-sectional area to improve the readability of the profiles. In Fig. 5.4, the flow direction is from the left side (fresh mixture) to the right (hot products) side.
The method used for calculation of flame thickness is as described by Spalding (1979),

$$b = \frac{T_b - T_u}{\left.\frac{dT}{dx}\right|_{max}} \tag{5.1}$$

Using the above formulation flame thickness at 1450K was found to be $9.298 \times 10^{-3} m$. Fig. 5.4 shows that the flame is 0.154 mm thick. Before -0.2mm, there were only reactants at uniform initial temperature. After 0.2mm, there were only products. As the temperature gradient is zero in the products, all the energy terms approach zero after the flame thickness. All the terms in the energy equation is active within the flame zone. The zone between the onset of convection and heat release is called pre-heat zone. Fig. 5.4 indicates that in the pre-heat zone, energy transported by reactants into the reaction zone balance the heat conduction

from reaction zone into the preheat zone. The zone comprising the location of onset of heat release till its end is called reaction zone. Fig. 5.4 shows that in the reaction zone, the magnitude of energy from the source term matches with the diffusion term. To understand the effect of increase in initial temperature on the flame structure, the simulations were performed for 400, 600 and 1450 K. The auto ignition temperature of premixed methane-air mixture is 853 K . Initial temperature 1450 K is well above the auto ignition temperature. The energy profiles across the flame front for higher initial temperatures are plotted in Fig. 5.5 with respect to distance and time.

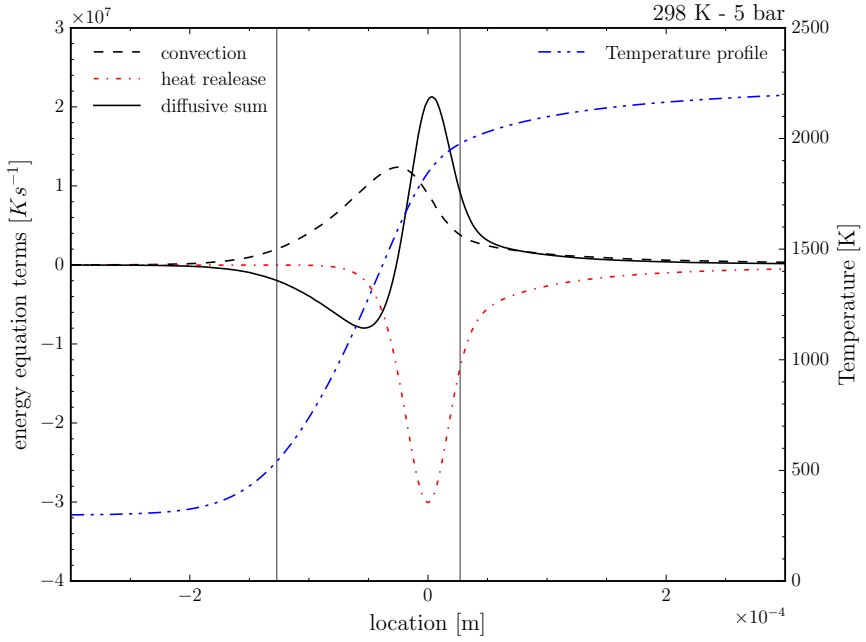


Fig. 5.4 Profiles of the terms of the energy equation for CH_4/air mixture, $\phi = 1.0$, $p_0 = 5\,bar$ and $T_0 = 298K$. The two vertical lines indicate the flame thickness.

.

Transformation from distance to time was performed using $dt = \rho A dx/\dot{M}$. For better comparison, the time range of time coordinate in Fig. 5.5 has been chosen to show the position of the peak value of the reaction source term. Energy profiles across the flame front simulated at initial temperatures of 400K and 600K look similar to that of 298K at 5 bar. At 1450K, the flame structure shows a drastic change. The profiles of the energy terms at 1450 K displayed that the energy transported due to diffusion terms lose their importance as compared to the heat release and convection term. At 1450 K, the profile of the convection term almost mirrors the profile of the heat release term, while for lower preheating temperatures such at 298K, 400 K and 600 K, energy transportation due to diffusion plays a predominant role in comparison to the convection and heat release terms.

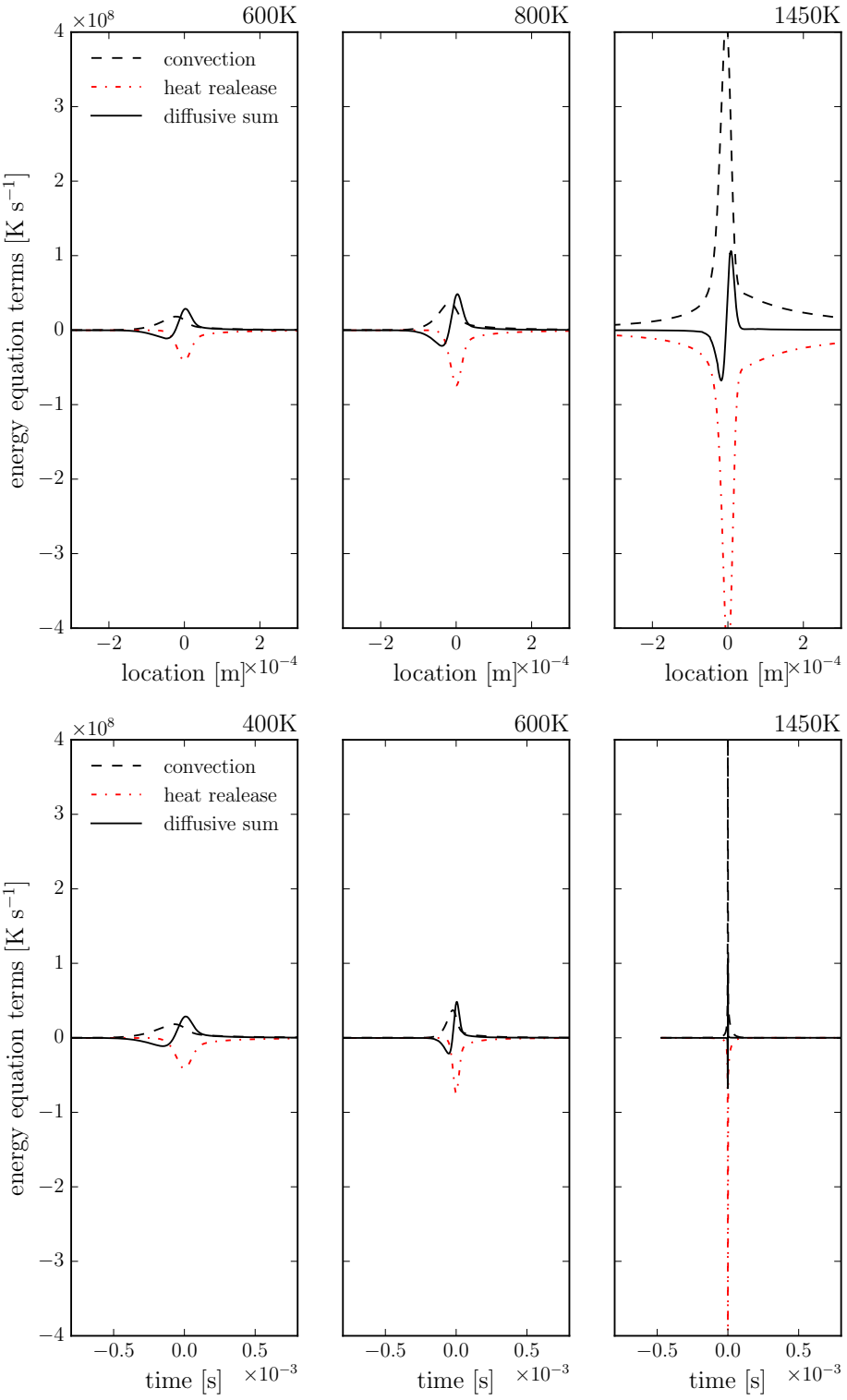


Fig. 5.5 Profiles of the terms of the energy equation for stoichiometric CH_4/air mixture at $p_0 = 5$ *bar*, and initial temperatures 400, 600 and 1450 K calculated by the GRI 3.0 mechanism. Diagrams displayed in space (top) and time (bottom) coordinates.

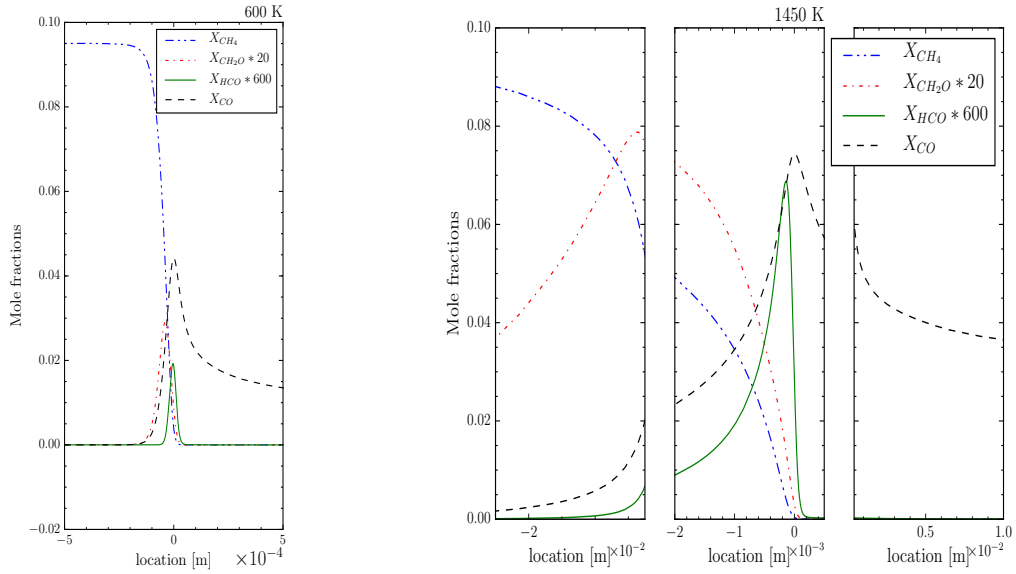


Fig. 5.6 Comparison of mole fraction profiles of CH_4, CH_2O, HCO and CO for laminar flame calculations of stoichiometric CH_4/air mixture at 5 *bar* at initial temperatures 600 K and 1450 K.

It is important to mention here that 1450 K is well above the auto ignition temperature. In the time coordinates, the reaction time scale of the flame front is very short compared to other lower initial temperatures. This clearly indicates that the flame propagation velocity of premixed combustible mixture at 1450K and 5 bar will be significantly higher. Figure 5.5 suggests that with increase in the preheating temperature well above the auto-ignition temperature, the importance of diffusion decreases and the structure of the flame approaches that of a plug flow reactor model in which the transport due to diffusion is being neglected completely. The intermediate species which are representatives of the progress of the reactions are plotted in Fig. 5.6. The mole fraction of all intermediate active species reaches its peak values in a much thinner regime in the flame front at 1450K when compared to 600K.

These observations are supported by comparing the mole fraction profiles of CH_4, CH_2O, HCO and CO for the initial temperatures 600 K and 1450 K in Fig. 5.6. While for the 600 K case the mole fractions remain constant up to below 1 mm prior to the flame region, in the 1450 K case the species CH_2O and HCO that are known to be important for auto-ignition start to increase immediately after the inlet boundary. This also reflects the results of the reaction flux analysis shown above. For sake of a better visibility of the main reaction zone the abscissa of the right plot in Fig. 5.6 has been scaled towards the inlet and outlet boundary.

5.4 Ignition delay time

The above discussion indicated that at preheating temperatures well above the auto-ignition temperature, the structure of the flame or reaction domain is similar to a reaction under plug flow conditions, i.e. without the contribution of mass diffusion or heat conduction. In order to validate the plug flow behavior of the reaction at high initial temperatures, the reaction progress has been quantified using a PFR model as mentioned in section 3.3.1 which calculates the species and temperature evolution in a transient homogeneous reactor.

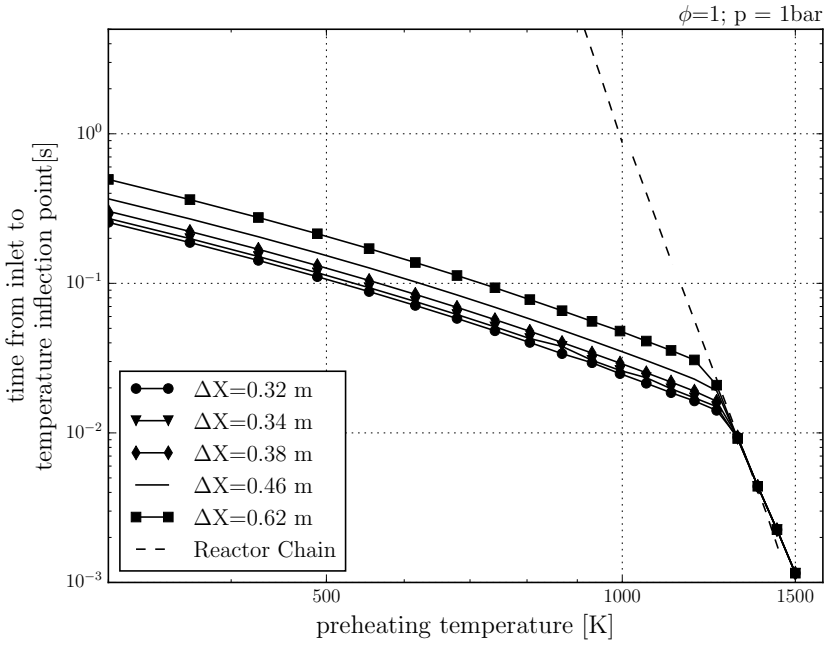


Fig. 5.7 Residence time from inlet to temperature inflection point (being characteristic for auto-ignition) as predicted from laminar, premixed flame calculations (lines with symbols). Ignition delay time (dashed line) predicted by the plug flow reactor model, as a function of unburned mixture preheating temperature

Assuming a constant mass flow in a plug flow reactor the position was transformed to the time coordinate of the homogeneous reaction using the transformation rule $dt = \rho A dx/\dot{M}$. The definition of the ignition delay time used in the present work is the residence time of the reactants between the start of the calculation and the occurrence of an inflection point in the time evolution of the temperature in the homogeneous reactor.
In Fig. 5.7, these residence times are plotted together with the ignition delay time of PFR. It was observed that for very high initial temperatures, the residence time up to the occurrence of the temperature inflection point from laminar flame calculations coincide with the curve of the ignition delay time of the homogeneous reactor calculation. Also it was observed that

if the incoming mixture is given sufficient time, the start of reaction is always controlled by auto-ignition provided the incoming mixture temperature is beyond a mixture-dependent threshold value called the "auto-ignition temperature". As mentioned in section 5.2 the laminar burning velocity is independent of the domain length as long as the preheating temperature T_0 is below the auto-ignition temperature, as shown in figure 5.2.

5.5 Effect of pressure

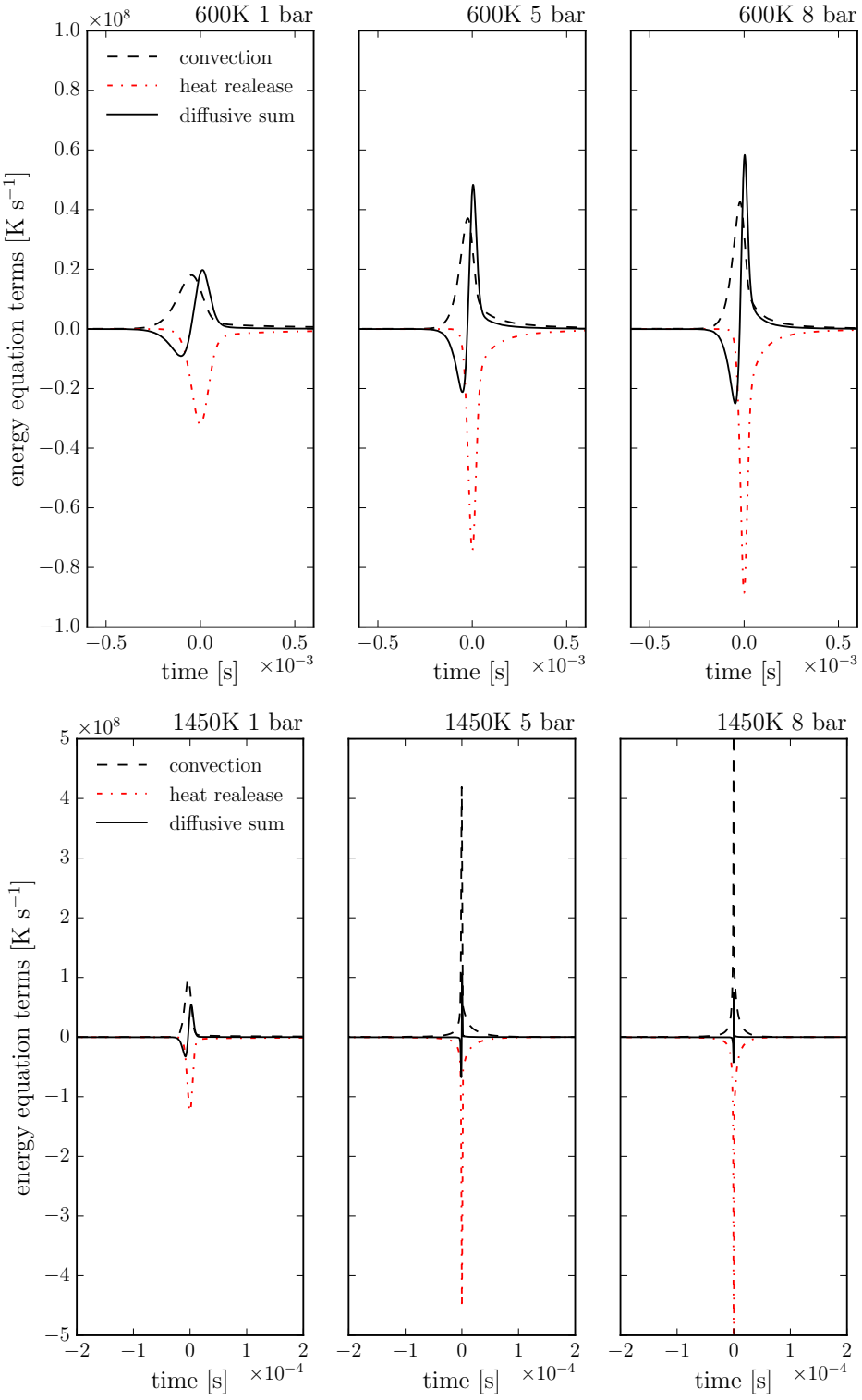


Fig. 5.8 Profiles of the terms of the energy equation for CH_4/air mixture, $\phi = 1.0$, with pressures of 1, 5 and 8 bar using the GRI 3.0 mechanism. Unburned mixture temperature T_0 is 600 K (top) and 1450 K (bottom).

To understand the effect of pressure on the burning velocity at higher initial temperatures, the energy equation terms were plotted in Fig. 5.8 for two initial temperatures, $T_0 = 600K$ in the top row and $T_0 = 1450K$ in the bottom row for different pressures such as 1, 5 and 8 bar. In figure 5.8 at $T_0 = 600K$ and 1 bar, the value of the diffusive sum term in the energy equation starts changing before the energy release term. The same trend was observed for other pressures such as 5 and 8 bar. However figure 5.8 indicates that at $T_0 = 1450K$, the profile of the diffusive sum term in the flame undergoes a drastic change with an increase in the pressure from 1 to 5 bar. At 1 bar and 1450 K, the diffusive sum terms begins to change slightly before the change in the energy release term, while at 5 and 8 bar change is of the opposite pattern. This is another indication that the chemical reaction at 1450 K and pressures at 5 and 8 bar is approaching a condition similar to that of a plug flow reactor, i.e. in the auto-ignition domain.

Chapter 6

Conclusions

The objective of the present work was to understand the stabilization mechanism of flame for initial temperatures well above the self ignition temperature. When it was modelled using 1D model, the laminar burning velocity obtained was very high. It also increased with increase in initial pressure. To understand the same, the contribution by different terms in the energy equation was investigated. It showed that the flame structure changed completely if the initial temperature increased beyond auto ignition temperature.

When inlet temperatures are below the auto-ignition temperature, the flame structure comprises a thick preheating zone and a comparatively thin reaction zone, and the flame is stabilized by means of diffusive controlled flame propagation. But when inlet temperatures were well above the auto-ignition temperature of the studied mixture, the relative influence of the diffusive sum terms in the energy balance equation decreases very rapidly whereas there is an increase of the heat release term which results in very high burning velocities. The system at this condition has the same ignition delay time as that of plug flow reactor irrespective of domain lengths.

If the initial temperature is kept constant and the initial pressure, for temperatures below auto ignition the flame remains diffsuion controlled even for high pressures of 8 bar. But with temperatures beyond auto-ignition even 1 bar the flame has a very low influence of diffusion term comparatively. For higher pressure there is a drastic increase of the heat release term and diffusion term is negligible.

References

Bird, R. B., Stewart, W. E. and Lightfoot, E. N. (1960), 'Transport phenomena'.

Dixon-Lewis, G. (1968), 'Flame structure and flame reaction kinetics, ii: Transport phenomena in multicomponent systems', *Proceedings of the Royal Society* **A**(304), 111–111.

Goodwin, D., Malaya, N., Moffat, H. and Speth, R. (2017), 'Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes. version 2.3.0'.

Habisreuther, P., Galeazzo, F. C. C., Prathap, C. and Zarzalis, N. (2013), 'Structure of laminar premixed flames of methane near the auto-ignition limit', *Combustion and Flame* **160**(12), 2770–2782.

Hermanns, R. T. E. (2007), 'Laminar burning velocities of methane-hydrogen-air mixtures', pp. 28–36.

Kee, R. J., Dixon-Lewis, G., Warnatz, J., Coltrin, M. E. and Miller, J. A. (1986), 'A fortran computer code package for the evaluation of gas-phase multicomponent transport properties', *Sandia National Laboratories Report SAND86-8246* **6**, 80401–1887.

Peters, N. and Warnatz, J. (2013), 'Numerical methods in laminar flame propagation: A gamm-workshop', **6**.

Prathap, C., Galeazzo, F. C., Kasabov, P., Habisreuther, P., Zarzalis, N., Beck, C., Krebs, W. and Wegner, B. (2012), 'Analysis of NO_x formation in an axially staged combustion system at elevated pressure conditions', *Journal of Engineering for Gas Turbines and Power* **134**(3), 031507.

Smyth, K. C. and Bryner, N. P. (1997), 'Short-duration autoignition temperature measurements for hydrocarbon fuels near heated metal surfaces', *Combustion Science and Technology* **126**(1-6), 225–253.

Spalding, D. B. (1979), 'Combustion and mass transfer: A textbook with multiple-choice exercises for engineering students', pp. 338–343.

Wilke, C. (1950), 'A viscosity equation for gas mixtures', *The journal of chemical physics* **18**(4), 517–519.

Appendix A

- The results for this study were generated using Python 3.5.

  https://www.python.org/

- The source code of the version of Cantera[Goodwin et al. (2017)] used can be obtained at https://github.com/Cantera/cantera

- Introduction to the documentation of Cantera(chapter 2and 3) has been obtained from http://www.cantera.org/docs/sphinx/html/index.html

- All the programs used for the generation of the results alongwith the results for this study can be obtained in the following Git repository: https://github.com/ardhendubarman/Cantera-Project.git.

```
"""
A freely-propagating, premixed methane flat flame with mixture-averaged
transport properties.
"""

import cantera as ct
import pickle as pkl

# Simulation parameters
p = 5*ct.one_atm  # pressure [Pa]
Tin = 298.0  # unburned gas temperature [K]
reactants = 'CH4:1,O2:2,N2:7.52'  # premixed gas composition

initial_grid = [0.0, 0.001, 0.01, 0.02, 0.029, 0.03]  # m
tol_ss = [1.0e-5, 1.0e-13]  # [rtol atol] for steady-state problem
tol_ts = [1.0e-4, 1.0e-13]  # [rtol atol] for time stepping
loglevel = 1  # amount of diagnostic output (0 to 8)
refine_grid = True  # 'True' to enable refinement, 'False' to disable
```

```
19
20 # IdealGasMix object used to compute mixture properties
21 gas = ct.Solution('gri30.xml')
22 gas.TPX = Tin, p, reactants
23
24 # Flame object
25 f = ct.FreeFlame(gas, initial_grid)
26 f.flame.set_steady_tolerances(default=tol_ss)
27 f.flame.set_transient_tolerances(default=tol_ts)
28
29 # Set properties of the upstream fuel-air mixture
30 f.inlet.T = Tin
31 f.inlet.X = reactants
32
33 #f.show_solution()
34
35 # Solve with the energy equation disabled
36 f.energy_enabled = False
37 f.set_max_jac_age(10, 10)
38 f.set_time_step(1e-5, [2, 5, 10, 20])
39 f.solve(loglevel=loglevel, refine_grid=False)
40 f.save('CH4_adiabatic.xml', 'no_energy', 'solution with the energy
    equation disabled')
41
42 # Solve with the energy equation enabled
43 f.transport_model= 'Mix'
44 f.set_refine_criteria(ratio=3, slope=0.04, curve=0.07)
45 f.energy_enabled = True
46 f.solve(loglevel=loglevel, refine_grid=refine_grid)
47 f.save('CH4_adiabatic.xml', 'energy', 'solution with mixture-averaged
    transport')
48 #f.show_solution()
49 print('mixture-averaged flamespeed = {0:7f} m/s'.format(f.u[0]))
50
51 # write the velocity, temperature, density, and mole fractions to a CSV
    file
52 f.write_csv('CH4_adiabatic.csv', quiet=True)
```

Listing A.1 Adibatic flame calculations

The following snippet was used for modelling Plug Flow Reactor:

```
1 # -*- coding: utf-8 -*-
2 """
```

```
This example solves a plug-flow reactor problem of Methane-air
    combustion.
The PFR is computed by the simulation of a chain of reactors.
"""

import cantera as ct
import numpy as np

#############################################################################
# Input Parameters
#############################################################################

T_0 = 300.0  # inlet temperature [K]
pressure = ct.one_atm  # constant pressure [Pa]
composition_0 = 'CH4:1, O2:2, N2:7.52'
length = 5  # *approximate* PFR length [m]
u_0 = 6  # inflow velocity [m/s]
area = 1.e-4  # cross-sectional area [m**2]

# input file containing the reaction mechanism
reaction_mechanism = 'gri30.xml'

# Resolution: The PFR will be simulated by 'n_steps' time steps or by a
    chain
# of 'n_steps' stirred reactors.
n_steps = 2000
#############################################################################



#############################################################################
# Method 2: Chain of Reactors
#############################################################################
# The plug flow reactor is represented by a linear chain of zero-
    dimensional
# reactors. The gas at the inlet to the first one has the specified
    inlet
# composition, and for all others the inlet composition is fixed at the
# composition of the reactor immediately upstream. Since in a PFR model
    there
# is no diffusion, the upstream reactors are not affected by any
    downstream
# reactors, and therefore the problem may be solved by simply marching
    from
```

```
# the first to last reactor, integrating each one to steady state.
# (This approach is anologous to the one presented in 'surf_pfr.py', which
# additionally includes surface chemistry)


# import the gas model and set the initial conditions
gas2 = ct.Solution(reaction_mechanism)
gas2.TPX = T_0, pressure, composition_0
mass_flow_rate2 = u_0 * gas2.density * area
dz = length / n_steps
r_vol = area * dz

# create a new reactor
r2 = ct.IdealGasReactor(gas2)
r2.volume = r_vol

# create a reservoir to represent the reactor immediately upstream. Note
# that the gas object is set already to the state of the upstream reactor
upstream = ct.Reservoir(gas2, name='upstream')

# create a reservoir for the reactor to exhaust into. The composition of
# this reservoir is irrelevant.
downstream = ct.Reservoir(gas2, name='downstream')

# The mass flow rate into the reactor will be fixed by using a
# MassFlowController object.
m = ct.MassFlowController(upstream, r2, mdot=mass_flow_rate2)

# We need an outlet to the downstream reservoir. This will determine the
# pressure in the reactor. The value of K will only affect the transient
# pressure difference.
v = ct.PressureController(r2, downstream, master=m, K=1e-5)

sim2 = ct.ReactorNet([r2])

# define time, space, and other information vectors
z2 = (np.arange(n_steps) + 1) * dz
t_r2 = np.zeros_like(z2)  # residence time in each reactor
u2 = np.zeros_like(z2)
t2 = np.zeros_like(z2)
states2 = ct.SolutionArray(r2.thermo)
# iterate through the PFR cells
```

```
for n in range(n_steps):
# Set the state of the reservoir to match that of the previous reactor
gas2.TDY = r2.thermo.TDY
upstream.syncState()
# integrate the reactor forward in time until steady state is reached
sim2.reinitialize()
sim2.advance_to_steady_state()
# compute velocity and transform into time
u2[n] = mass_flow_rate2 / area / r2.thermo.density
t_r2[n] = r2.mass / mass_flow_rate2  # residence time in this reactor
t2[n] = np.sum(t_r2)
# write output data
states2.append(r2.thermo.state)

#############################################################################


#############################################################################
# Compare Results in matplotlib
#############################################################################

import matplotlib.pyplot as plt

plt.figure()
##plt.plot(z1, states1.T, label='Lagrangian Particle')
plt.plot(z2, states2.T, label='Reactor Chain')
plt.xlabel('$z$ [m]')
plt.ylabel('$T$ [K]')
plt.legend(loc=0)
plt.show()
plt.savefig('pfr_T_z3.png')

plt.figure()
plt.plot(t2, states2.X[:, gas2.species_index('H2')], label='Reactor
    Chain')
plt.xlabel('$t$ [s]')
plt.ylabel('$X_{H_2}$ [-]')
plt.legend(loc=0)
plt.show()
plt.savefig('pfr_XH2_t.png')
```

Listing A.2 Plug Flow Reactor

The following section is the module for energy equation calculations by Cantera:

```
//-----------------------------------------------------
//     energy equation
//
//     \rho c_p dT/dt + \rho c_p u dT/dz
//     = d(k dT/dz)/dz
//       - sum_k(\omega_k h_k_ref)
//       - sum_k(J_k c_p_k / M_k) dT/dz
//-----------------------------------------------------
if (m_do_energy[j]) {
setGas(x,j);

// heat release term
const vector_fp& h_RT = m_thermo->enthalpy_RT_ref();
const vector_fp& cp_R = m_thermo->cp_R_ref();
double sum = 0.0;
double sum2 = 0.0;
for (size_t k = 0; k < m_nsp; k++) {
double flxk = 0.5*(m_flux(k,j-1) + m_flux(k,j));
sum += wdot(k,j)*h_RT[k];
sum2 += flxk*cp_R[k]/m_wt[k];
}
sum *= GasConstant * T(x,j);
double dtdzj = dTdz(x,j);
sum2 *= GasConstant * dtdzj;

rsd[index(c_offset_T, j)] = - m_cp[j]*rho_u(x,j)*dtdzj
- divHeatFlux(x,j) - sum - sum2;
cout<< m_cp[j]*rho_u(x,j)*dtdzj/(m_rho[j]*m_cp[j])<<" " <<- divHeatFlux(
    x,j)/(m_rho[j]*m_cp[j])<<" " << sum/(m_rho[j]*m_cp[j])<<" " << sum2/(
    m_rho[j]*m_cp[j])<<" "<<( divHeatFlux(x,j)-sum2)/(m_rho[j]*m_cp[j])<<
     endl;
rsd[index(c_offset_T, j)] /= (m_rho[j]*m_cp[j]);
rsd[index(c_offset_T, j)] -= rdt*(T(x,j) - T_prev(j));
rsd[index(c_offset_T, j)] -= (m_qdotRadiation[j] / (m_rho[j] * m_cp[j]))
    ;
diag[index(c_offset_T, j)] = 1;
} else {
// residual equations if the energy equation is disabled
rsd[index(c_offset_T, j)] = T(x,j) - T_fixed(j);
diag[index(c_offset_T, j)] = 0;
}

rsd[index(c_offset_L, j)] = lambda(x,j) - lambda(x,j-1);
diag[index(c_offset_L, j)] = 0;
```