Yet Another Finite-Rate Chemistry Module for Compressible Flow Codes

Rowan J. Gollan

Division Report 9/2003
July 2003

Contents

List of Figures

1 Introduction

1.1 Thermochemical nonequilibrium processes

Thermochemical nonequilibrium can be thought of as nonequilibrium processes in the energy relaxation and the chemical relaxation. In its most general form, both these processes occur simultaneously. Consider a characteristic time for nonequilibrium processes to occur (eg. the equilibriation of a chemical reaction) τ_c and also a characteristic time for flow in the domain (eg. the time it takes a fluid particle to traverse the shock layer of a blunt body vehicle) τ_f. Based on these timescales, the flow can be classified in one of three ways:

- frozen flow —
$$\tau_c >> \tau_f$$
- nonequilibrium flow —
$$\tau_c \approx \tau_f$$
- equilibrium flow —
$$\tau_c << \tau_f$$

The limiting cases of frozen flow and equilibrium flow are easier to treat in a computational fluid dynamics (CFD) solution of hypersonic gas flows. However, there are significant portions of the flowfield where neither of these limiting case assumptions hold and a general nonequilibrium treatment must be employed. Gnoffo *et al* [1] explain the presence of nonequlibrium flow (in blunt body flowfields) as follows:

> The combination of low density in the upper atmosphere (which lowers collision frequency) and high vehicle velocity (which lowers transit time) creates the conditions which make nonequilibrium phenomena an important aspect of the shock-layer flow.

1.2 Outline of the report

This report gives an overview of a finite-rate chemistry module for use in compressible flow codes which use a timestep-splitting approach. In Section 2, the governing equations for the reacting hypersonic flowfields are presented. Section 3 describes the formulation and implementation of a module to compute the thermodynamic and transport properties of a multi-species reacting gas. The discussion of the finite-rate chemistry module is given in Section 4. The final section of the report, Section 5 presents some test cases of the finite-rate chemistry module. This section can also be used as a “User’s Guide” to the finite-rate module as detailed setups of each of the test cases are given.

2 Background

2.1 Governing equations for reacting flows

The integral form of the reacting Navier-Stokes equations is presented here. This follows the form used by Jacobs [2] with the modifications used by Oran and Boris [3] for reacting flow. The two-dimensional equtions are as follows

$$\frac{\partial}{\partial t}\int\int_{\Omega} U dxdy + \int_{S}(F - F_v)dy - \int_{S}(G - G_v)dx = \int\int_{\Omega} Q dxdy, \tag{1}$$

where

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \\ \rho f_{is} \end{bmatrix}, \tag{2}$$

is the vector or conserved quantities,

$$F = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho v u \\ \rho E u + P u \\ \rho f_{is} u \end{bmatrix}, \quad G = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ \rho E v + P v \\ \rho f_{is} v \end{bmatrix}, \tag{3}$$

are the inviscid flux vectors,

$$F_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{xx} u + \tau_{yx} v + q_x \\ \rho f_{is} V_{x,is} \end{bmatrix}, \quad G_v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{xy} u + \tau_{yy} v + q_y \\ \rho f_{is} V_{y,is} \end{bmatrix}, \tag{4}$$

are the viscous flux vectors and

$$Q = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \dot{\omega}_i \end{bmatrix} \tag{5}$$

is the vector of source terms which includes the species production to model the chemical effects. The above conservation equations include only one conservation equation for energy thus it is assumed that one thermodynamic temperature characterises the flow. This is the equivalent of assuming thermal equilibrium among the gas molecules.

2.2 Equation of state

In order to close the equation set above, an equation of state is required that relates the pressure to the internal energy and density of the form,

$$p = p(\rho, e). \tag{6}$$

In hypersonic reacting flows, it is appropriate to treat the gas as a mixture of thermally perfect gases [4]. In this case, the thermodynamic state is characterised by two state variables, along with the fractions of each species. Thus the equation of state is more correctly expressed as,

$$p = p(\vec{\rho}, e), \tag{7}$$

where $\vec{\rho}$ is a vector of partial densities for each of the species in the mix. The details of computing this equation of state are presented in Secion 3.1.

2.3 Physical model for the chemical source terms

The chemical effects are included in the reacting Navier-Stokes equations through the source term coupled to the species continuity equations. The model required needs to describe the production rate of species i in each cell over the flow timestep. Following Anderson's [4] notation, a general chemical reaction can be written as,

$$\sum_{i=1}^{N} \nu_i' X_i \underset{k_b}{\overset{k_f}{\rightleftharpoons}} \sum_{i=1}^{N} \nu_i'' X_i \tag{8}$$

where ν_i' and ν_i'' represent the stoichiometric coefficients for the reactants and products respectively. For a given reaction j, the rate of production of species i can be expressed as,

$$\left(\frac{d[X_i]}{dt}\right)_j = (\nu_i'' - \nu_i') \left\{ k_f \prod_i [X_i]^{\nu_i'} - k_b \prod_i [X_i]^{\nu_i''} \right\}. \tag{9}$$

The total rate of production of species i is found from summation over all reactions

$$\frac{d[X_i]}{dt} = \sum_{j=1}^{N_r} \left(\frac{d[X_i]}{dt}\right)_j . \tag{10}$$

In the expression for rate of species production, k_f and k_b are the forward and backward reaction rate coefficients respectively. In this model, they are computed from a modified Arrhenius expression of the form,

$$k_f = A_f T^{B_f} \exp(-T_{D_f}/T) \tag{11}$$

where A_f and B_f are constants for the reaction and T_{D_f} is the activation energy for the reaction. These constants are measured experimentally and can be found in the literature for the reaction schemes of interest.

3 Calculating thermodynamics and transport properites

Closely related to the finite-rate chemistry work is a model for the thermodynamic state of the gas mixture. This work needed to be completed prior to the chemistry model and is presented in Section 3.1 The equations that describe the chemistry were presented previously in Section 2.3 and the numerical techniques for solving these equations are presented in 4.2.

3.1 A thermochemical model for a mixture of perfect gases

In a compressible flow CFD simulation, a model of the gas is required to link the value of pressure to the density and internal energy as in

$$p = p(\rho, e). \tag{12}$$

The form of this equation is related to the gas model chosen. For the simulation of chemically reacting flows for hypersonic regimes, it is most appropriate to model the gas as a mixture of thermally perfect gases. In this instance, the model used assumes thermal equilibrium of all the species in the flow. The assumption of thermal equilibrium means that all internal energy modes (translational, rotational, vibrational and electronic) are in equilibrium at the one common thermodynamic temperature. This assumption may only cause problems in the vicinity immediately behind a shock.

In the next sections, the properties of a single thermally perfect gas and those of a mixture of such gases are described. Following this is a discussion of the implementation in the equation of state routines.

3.2 Thermodynamic properties for a mixture of perfect gases

3.2.1 Properties of a single thermally perfect gas

A thermally perfect gas is one where c_p and c_v are functions of temperature (and not contants as is the case for a calorically perfect gas) [4]. The assumption that intermolecular forces are negligible is valid for almost all hypersonic applications [4] and thus the perfect gas equation of state still holds

$$pv = RT. \tag{13}$$

The temperature variation of the specific heats is due to the excitation of the vibrational and electronic energy modes within the gas molecules (only electronic excitation can occur in atoms). The values of specific heats can be calculated from statistical mechanics using the partition functions for the species. This calculation would be too costly in a CFD simulation. The alternatives used most often are look-up tables of values for specific heat or a polynomial curve fit to the data.

In this model, the polynomial fits suggested by Gordon and McBride [5] have been adopted. These polynomials are used as the basis for the NASA program CEA (Chemical Equilibrium Analysis) [5]. The form of the polynomial is given as

$$\frac{C_p^\circ}{R} = a_1 T^{-2} + a_2 T^{-1} + a_3 + a_4 T + a_5 T^2 + a_6 T^3 + a_7 T^4 \tag{14}$$

The coefficients $a_1 \dots a_7$ are different for each species and different for various termperature ranges. The nominal temperature ranges for gases in the CEA tables are 200 to 1000K, 1000 to 6000K and 6000 to 20 000K.

The enthalpy for the gas at this temperature is found by integrating the curve for C_p as follows

$$H^\circ = \int_{T_{ref}}^{T} C_p^\circ dT + H_{ref}^\circ. \tag{15}$$

As the form of polynomial in Eq. 14 can be integrated analytically, this gives rise to

$$\frac{H^\circ}{RT} = -a_1 T^{-2} + a_2 T^{-1} \ln T + a_3 + a_4 \frac{T}{2} + a_5 \frac{T^2}{3} + a_6 \frac{T^3}{4} + a_7 \frac{T^4}{5} + \frac{a_8}{T} \tag{16}$$

In the values given for the coefficients in the CEA tables [5], the constant of integration is included in a_8. Based on the practice of Gordon and McBride [5] the reference chosen means that the enthalpy for a species at 298.15K is the enthalpy of formation at 298.15K.

3.2.2 Properties for a mixture of perfect gases

The thermodynamic state of a mixture of perfect gases is uniqely defined by two state variables and the proportions of each species (ie. mass fractions, mole fractions or concentrations). The internal energy for the mixture is a weighted sum of the internal energies of each species present

$$e = \sum_{i=1}^{N} f_i e_i. \tag{17}$$

Similarly, the mixture enthalpy can also be found

$$h = \sum_{i=1}^{N} f_i h_i. \tag{18}$$

From the definition of enthalpy and using the perfect gas equation of state, an expression for the temperature can be linked to the internal energy and enthalpy

$$e = h - RT \tag{19}$$

where

$$R = \sum_{i=1}^{N} f_i R_i.$$

With a knowledge of temperature, the pressure can be evaluated by Dalton's law of partial pressures as

$$p = \sum_{i=1}^{N} \rho_i R_i T. \tag{20}$$

Also the frozen sound speed can be calculated using

$$a = \sqrt{\gamma R T} \tag{21}$$

where γ is the ratio of specific heats defines as

$$\gamma = \frac{C_p}{C_v}. \tag{22}$$

The specific heat at constant pressure, C_p, for the mixture is found from

$$C_p = \sum_{i=1}^{N} f_i C_{p_i}, \tag{23}$$

and thus C_v can be found

$$C_v = C_p - R. \tag{24}$$

Most of the above equations require knowledge of the temperature in order to use them in the explicit forms shown here. However, temperature is not a value that is readily obtained from the conservation equations. The process to find temperature based on internal energy and the density vector is an iterative one and described in the next section.

3.2.3 Implementation of themodynamics calculations

The problem of calculating temperature is overcome by using an iteration technique. A given internal energy is denoted as e_{given} which comes from subtracting the kinetic energy from the total gas mixture energy. The total energy is a conserved quantity and comes directly from the solution of the energy equation. Then using a guessed temperature, T_{guess}, it is possible to compute an internal energy of the mixture based on that guess as follows,

$$e_{guesss} = \sum_{i=1}^{N} f_i e_i = \sum_{i=1}^{N} f_i \left(h_i - R_i T_{guess} \right). \tag{25}$$

If a function f is written such that

$$f = f(T) = e_{given} - e_{guess}, \tag{26}$$

then the zero of this function will give the correct value for temperature. The zero finding method used here is the secant method. Gerald and Wheatley [6] provide a discussion on the secant method in their text.

The routine for finding temperature is summarised here.

1. guess T
2. compute h_i using polynomials
3. compute h_{mix} using $h_{mix} = \sum_{i=1}^{N} f_i h_i$
4. $e_{guess} = h_{mix} - R_{mix} T$ where $R_{mix} = \sum_{i=1}^{N} f_i R_i$
5. compute $f(T) = e_{given} - e_{guess}$
6. if $|f(T)| < tol$ finish; else new guess using secant formula and repeat from 2.

With temperature found, the other thermodynamic properties follow using the equations in Section 3.2.2

Test case for thermodynamics calculations This test case provides a demonstration that the perfect gas mixture model has been implemented correctly with regards to computing thermodynamic properties. The test case chosen is Sod's shock tube problem [7] which has become a standard test case for compressible flow CFD codes.

The shock tube considered is 1m in length. The active computational domain is 100 cells extended over $0 < x < 1$m. Viscous effects are neglected. The mixture of perfect gases chosen is 0.7778 N_2 and 0.2222 O_2 by mass fraction. This was chosen as a "fake air" which should closely match the analytical model for air at these temperatures. It will be a test of the perfect gas mixture "fake air" if it can recreate the solution computed analytically.

The initial state, for $x \leq 0.5$m is

$$\rho = 1.0 \mathrm{kg/m^3}, \quad P = 10^5 \mathrm{Pa}, \quad u = 0 \mathrm{m/s}, \quad e = 2.5 \times 10^5 \mathrm{J/kg/K},$$

while, for $x > 0.5\mathrm{m}$, it is

$$\rho = 0.125 \mathrm{kg/m^3}, \quad P = 10^4 \mathrm{Pa}, \quad u = 0 \mathrm{m/s}, \quad e = 2.0 \times \mathrm{J/kg/K}.$$

The hypothetical diaphragm is removed at $t = 0$. The solution was stopped at $t = 0.603 \times 10^{-3}$ s and is shown in Figure 1. The computed solution shows good agreement with the analytical solution. This is a fair indication that the mixture of perfect gases model has been implemented correctly. The discrepancies near the shock are the standard numerical errors involved with any finite-voloume solution. Most of the discrepancies occur where the flow changes most rapidly. This is to be expected in any scheme where discontinuities are "captured" over a few cells that have large flow gradients.

3.3 Transport properties for a mixture of perfect gases

3.3.1 Properties for a single perfect gas

Just as for the thermodynamic data, the least squares curve fits of Gordon and McBride [5] were used to compute the transport properties. The form of these curve fits is

$$\left.\begin{array}{l} \ln \eta \\ \ln k \end{array}\right\} = A \ln T + \frac{B}{T} + \frac{C}{T^2} + D \qquad (27)$$

where η is the viscosity and k is the thermal conductivity. Using the given coefficients in the CEA data [5] the unit of viscosity are micropoise (µP) and the units for thermal conductivity are in microwatts per centimetre-Kelvin ($\mu\mathrm{W/cm\ K}$).

3.3.2 Properties for the mixture of gases

In order to calculate the mixture viscosity and thermal conductivity, an approximate mixing method must be chosen. The method chosen here follows that of Gordon and McBride [5] where the mixture formulae take these forms:

$$\eta_{mix} = \sum_{i=1}^{N} \frac{x_i \eta_i}{x_i + \sum_{\substack{j=1 \\ j \neq i}}^{N} x_j \phi_{ij}} \qquad (28)$$

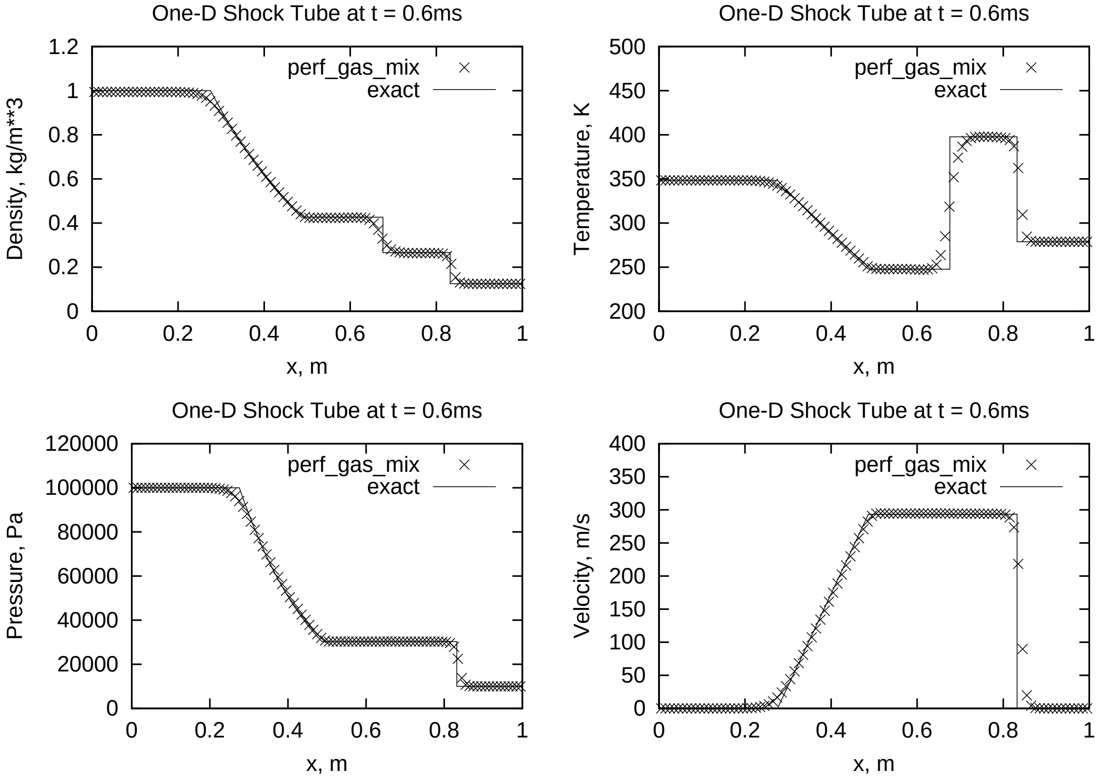


Figure 1: Solutions for the one-dimensional shock tube at t = 0.603×10^{-3}s

and

$$k_{mix} = \sum_{i=1}^{N} \frac{x_i k_i}{x_i + \sum_{\substack{j=1 \\ j \neq i}}^{N} x_j \psi_{ij}} \tag{29}$$

where x_i is the mole fraction of species i.

The interaction potentials, ϕ_{ij} and ψ_{ij}, can be calculated a number of ways. Again, the formulae suggested by Gordon and McBride [5] have been used,

$$\phi_{ij} = \frac{1}{4} \left[1 + \left(\frac{\eta_i}{\eta_j} \right)^{1/2} \left(\frac{M_j}{M_i} \right)^{1/4} \right]^2 \left(\frac{2M_j}{M_i + M_j} \right)^{1/2} \tag{30}$$

and

$$\psi_{ij} = \phi_{ij} \left[1 + \frac{2.41(M_i - M_j)(M_i - 0.142M_j)}{(M_i + M_j)^2} \right] \tag{31}$$

where M_i and M_j refer to the molecular weights of species i and j respectively.

Once the mixture viscosity and thermal conductivity have been computed, it is possible to compute the Prandtl number using the following expression

$$Pr = \frac{\mu_{mix} C_{p_{mix}}}{k_{mix}}. \tag{32}$$

3.3.3 Test case for transport properties calculations

This test case aims to demostrate that the transport properties are calculated correctly by the new routines. The test case chosen is the internal viscous flow through a cylinder. Jacobs [8] presented this test case for a gas model of perfect air. It has been repeated here with “fake air” as a test of the transport properties calculation.

The hollow cylinder is 1 m and has a radius of 0.005 m. The flow conditions are:

$$p = 257\mathrm{Pa}, \quad T = 222\mathrm{K}, \quad u_x = 597.3\mathrm{m/s}, \quad u_y = 0.0\mathrm{m/s}.$$

The internal wall of the cylinder along which the boundary layer develops is fixed at 222 K.

The solution for this test case has been plotted against the solution by Jacobs with a perfect gas model. The agreement is excellent and the small discrepancies can be attributed to the difference between the perfect air model and the perfect gas mixture model mimicing “fake air”. The velocity and temperature profiles through the boundary layer are shown in Figure 2 at 0.916 m from the leading edge.

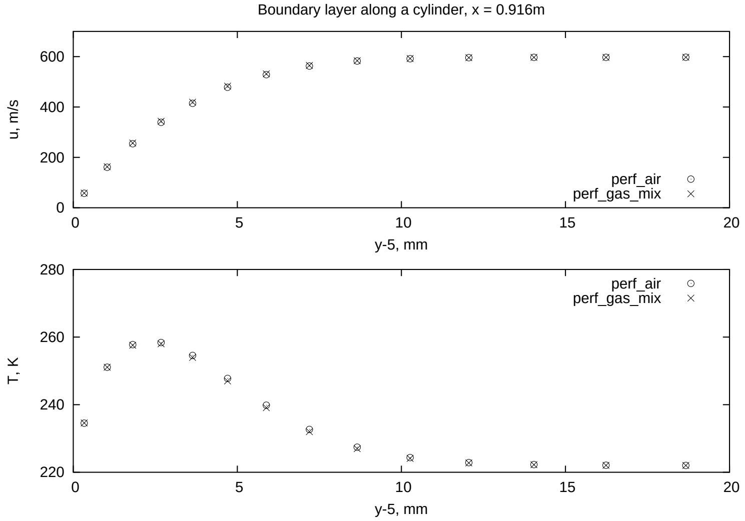


Figure 2: Velocity and temperature profiles through the boundary layer along a cylinder at $x = 0.916\mathrm{m}$

4 A Finite-Rate Chemistry Calculator

The problem of calculating finite-rate chemical effects is reduced to computing the production (or loss) of species in each cell due to chemical reactions at each timestep. In the full set of Navier-Stokes equations for compressible flow, the chemistry effect appears as a source term in each of the species continuity equations. The equations that describe the chemistry were presented previously in Section 2.3 and the numerical techniques for solving these equations are presented in 4.2.

4.1 Coupling the fluid dynamics and chemistry

In modelling reacting flows, the question arises as how to best couple the different physical processes. The approach taken here is a timestep-splitting method suggested by Oran and Boris [3]. This approach works for CFD schemes based on explicit timestep updates as is the case for MB_CNS [8]. In this method, each of the major physical processes is computed in isolation from the others. Although it might be argued that all of the processes occur simultaneously, the philosophy is that with a small enough timestep this approach can provide accurate results.

4.2 Integration schemes for the chemistry time-step

Having presented the equation for rate production of a single species i in Equation 10), the problem remains of solving the set of equations for all species. The equations form a set of ordinary diffential equations (ODEs). These equations are local only, meaning that they do not rely on any spatial gradients in the surrounding regions.

The problem faced in integrating sets of chemical equations is the so called "stiff ODE" problem. A "stiff ODE" has no strict definition in the literature but the one that serves best for the systems considered here is one given by Oran and Boris [3]:

> In practical computations, a system is stiff if the stepsize, based on cost or running time, is too large to give a stable, accurate answer.

In the case of chemical systems, this occurs if one or more reaction mechanisms has a time constant significantly smaller than the majority of other reactions. These fast reactions limit the timestep for stability.

There are, however, a number of integration schemes designed specifically to combat this problem of stiffness. Oran and Boris [3] list a number of general approaches:

- *backward differentiation formulae* — require recording of variable from a few previous time-steps;
- *exponential methods* — an implicit method;
- *asymptotic methods* — choose what the solution asymptotes to at end of timestep; and
- *quasi-steady-state approach* — predictor-corrector method using an analytical asymptotic solution as a first guess.

Not all are practical in a reacting flow simulation. The backward differentiation formulae approach requires the recording of information a few timesteps previously. This information is virtually useless in a timestep-splitting simulation because of all the other flow physics that occurs to change the variable. From a more pragmatic view, there are tedious memory management issues in recording data a few timesteps previously at every cell and there are issues to overcome at startup when no previous data is known.

In this chemistry calculator for the present work, three integration schemes have been implemented. In order to combat the stiffness problem, one method is an asymptotic method and the other is a more modern quasi-steady-state method. It is easiest to discuss the integration methods if we consider Equation 10 of the form

$$\frac{d[X_i]}{dt} = Q_i - [X_i]L_i \tag{33}$$

$$\frac{d[X_i]}{dt} = Q_i - \frac{[X_i]}{\tau_i} \tag{34}$$

where Q_i is a production term, L_i is a loss term and τ_i is the inverse of L_i and represents a time constant.

The first method implemented is a simple Euler timestepping. This can be used for simple chemical systems like the thermal dissociation of nitrogen. The update using a simple Euler timestepping is as follows,

$$[X_i]^{n+1} = [X_i]^n + \Delta t \left(Q_i^n - \frac{[X_i]^n}{\tau_i^n} \right). \tag{35}$$

where Δt is the chemical timestep.

The second method is an asymptotic method and makes use of Young and Boris' [9] selected asymptotic integration method. In this method, two different updates are available. The update chosen for each species is dependent upon the time constant for that species' production. A fast time constant indicates a stiff mechanism and so the asymptotic update is applied. The normal species production terms are treated by a modified Euler type update. Young and Boris provide an algorithm for deciding which species are treated stiffly and which are treated normally in a given timestep.

Regardless of whether a normal update or a stiff update is used, the scheme is based on a predictor-corrector scheme. The corrector step is iterated upon until a desired convergence is acheived. The predictor and corrector equations for the normal update are as follows,

$$[X_i]^1 = [X_i]^0 + \Delta t \left(Q_i^0 - \frac{[X_i]^0}{\tau_i^0} \right) \tag{36}$$

$$[X_i]^{n+1} = [X_i]^0 + \frac{\Delta t}{2} \left(Q_i^n - \frac{[x_i]^n}{\tau_i^n} \right) \tag{37}$$

The updates used for a stiff term are,

$$[X_i]^1 = \frac{[X_i]^0 \left(2\tau_i^0 - \Delta t\right) + 2\Delta t \tau_i^0 Q_i^0}{2\tau_i^0 + \Delta t} \tag{38}$$

$$[X_i]^{n+1} = \frac{\left\{ \frac{\Delta t}{2} \left[\tau_i^n + \tau_i^0\right] \left[Q_i^n + Q_i^0\right] + [X_i]^0 \left[\tau_i^n + \tau_i^0 \Delta t\right] \right\}}{\left[\tau_i^n + \tau_i^0 + \Delta t\right]}. \tag{39}$$

The third method used in the present work is a quasi-steady-state method. The method chosen was presented by Mott [10] in his thesis. The quasi-steady-state method by Mott makes use of an α parameter. The calculation of this parameter considers whether the species production is stiff or not and modifies the update accordingly. This simplifies the implementation as only one set of predictor-corrector equations are needed for this scheme. The equations are as follows,

$$\begin{aligned} [X_i]^1 &= [X_i]^0 + \frac{\Delta t Q_i^0}{1+\alpha_i^0 \Delta t L_i^0} & (40) \\ [X_i]^{n+1} &= [X_i]^0 + \frac{\Delta t \left(\bar{Q}_i - [X_i]^0 \bar{L}_i\right)}{1+\bar{\alpha}_i \Delta t \bar{L}_i}. & (41) \end{aligned}$$

In the above formula,

$$\bar{L}_i = \frac{1}{2}\left(L_i^0 + L_i^n\right) \tag{42}$$

and

$$\bar{Q}_i = \bar{\alpha}_i Q_i^n + \left(1-\bar{\alpha}_i\right) Q_i^0. \tag{43}$$

The key to the scheme is calculating α correctly. This α parameter controls how the update works on a given species integration. Noting that α is defined by,

$$\alpha(L\Delta t) \equiv \frac{1-\left(1-e^{-L\Delta t}\right)/(L\Delta t)}{1-e^{-L\Delta t}}. \tag{44}$$

Using Pade's approximation,

$$e^x \approx \frac{360+120x+12x^2}{360-240x+72x^2-12x^3+x^4} \tag{45}$$

it is possible to write a form of the expression for α which is much more amenable to numerical computation as the expensive exponetial function evaluation is avoided. The approximation for α becomes,

$$\alpha(L\Delta t) \approx \frac{180r^3+60r^2+11r+1}{360r^3+60r^2+12r+1} \tag{46}$$

where $r \equiv 1/(L\Delta t)$.

5 Test Cases and User Guide

In this section a number of test cases are presented which show validation of the finite-rate chemistry module. Along with each of these test cases is a description of the necessary setup required to run these simulations. It is anticipated that these examples can be used as a guide to using the finite-rate chemistry module within MB_CNS and L1d.

5.1 Setting up a finite-rate chemistry problem

In order to run a finite-rate simulation, the flow solvers need access to two files in the working directory `species.dat` and `reactions.dat`. The file `species.dat` contains all the information about the species mixture and all the necessary data to compute the thermodynamics and transport properties of the mixture. The data related to the chemical reaction scheme is specified in `reactions.dat`.

However, in general it is not anticipated that the user will need to be familiar with the details of these files. A python script called `script_reac.py` can be used along with an input file to automatically generate the `species.dat` and `reactions.dat` files. Some examples and descriptions of these input files is given in the following examples.

5.2 Static Reactor

The static reactor was used as a simple example to test the coupling of the finite-rate module to the flow code. The problem consists of a reacting mixture in a closed container thus ensuring that internal energy and density remain constant. The conditions were chosen such that the reacting mixture is not in chemical equilibrium initially. The initial conditions were:

$$p = 10^5\mathrm{Pa}, \quad T = 4000.0\mathrm{K}, \quad f_{N_2} = 0.8, \quad f_N = 0.2.$$

The reacting mixture is composed of nitrogen and two dissociation reactions are considered:

$$\begin{aligned} N_2 + N_2 &\rightleftharpoons N + N + N_2 \\ N_2 + N &\rightleftharpoons N + N + N \end{aligned}$$

Using the CEA program [5] it is possible to predict the composition of this mixture at chemical equilibrium. The predicited mass fractions are:

$$f_{N_2} = 0.86941, \quad f_N = 0.13059.$$

The computed solution using the finite-rate chemistry calculator is shown in Figure 3. This graph shows the evolution of the species mass fractions with time. Very good agreement is achieved between the equilibrium predicted by the finite-rate evolution and that found by the CEA program (which are shown as points on the graph). The small discrepancies are due to the manner in which the equilibrium prediction is made. The CEA program relies on thermodynamic arguments and computes equilibrium based on a minimisation of the Gibbs free energy of the mixture. The equilibrium prediction of the finite-rate method is just a by-product of the nonequilibrium system relaxing towards equilibrium.

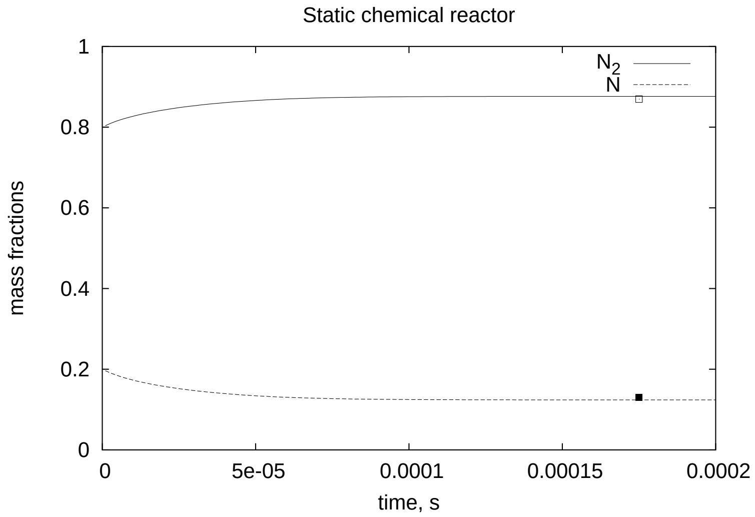


Figure 3: Evolution of species mass fractions in a reacting mixture of nitrogen

This occurs when the forward and backward rates match. Thus slight discrepancies come about due to the selection of reaction rate schemes. Although not shown here, it is possible to get quite significant errors from a poor choice in reaction rate coefficients.

Setting up the problem The first file listed here is the input file which describes the species and reactions considered in the simulation. A detailed description of the input file format is given in Appendix A. In this file, we see that there are two reacting species N_2 and N and they participate in two dissociation reactions.

nitrogen_2s2r.chm

```
# park.chm
# A description of thermal dissociation of Nitrogen
#
# Reaction rates taken from:
# Park (1990) [for a 2-T model]
#
# Rowan Gollan
# 04-Nov-02
SPECIES POLY
        N2 N
END

REACTIONS         saim        cms    k_b
N2 + N2 <=> 2 N + N2        k_f   7.0e21      -1.6    1.132e5
                        k_b   1.09e16     -0.5    0.0
N2 + N  <=> 2 N + N     k_f   3.0e22      -1.6    1.132e5
                        k_b   2.32e21     -1.5    0.0
END
```

The MB_CNS scriptit file `reactor.sit` is also listed here. It shows the necessary changes to a `.sit` file required for a reacting flow simulation. These changes are:

- use of `perf_gas_mix` as the gas type
- listing the the mass fractions of all species on the GAS_STATE description line (the order of species mass fractions listing follows that in the chemistry input file)
- the presence of the keyword “`reacting`” in the control section

Note it is possible to run a frozen multi-species simulation by omitting the keyword “`reacting`” from the control definition.

reactor.sit

```
# reactor.sit
# A box reactor containing static a mixture of perfect gases

# Set up one square in the (x,y)-plane.
# The small size of the reactor sides will force the gas-dynamics
# time step to be small.
BEGIN_GEOMETRY
   NODE a 1.0e-2 0.0
   NODE b 1.0e-2 1.0e-2
   NODE c 0.0    1.0e-2
   NODE d 0.0    0.0

   LINE ab a b
   LINE bc b c
   LINE cd c d
   LINE da d a
   POLYLINE   north0   1 - bc
   POLYLINE   east0    1 + ab
   POLYLINE   south0   1 + da
   POLYLINE   west0    1 - cd
END_GEOMETRY

BEGIN_FLOW
   # Gas and flow properties for an N2 + N mix
   GAS_TYPE   perf_gas_mix
   #                                       SPECIES:   N2   N
   GAS_STATE   initial   1.0e5  0.0  0.0  4000.0           0.8  0.2

   # Set the boundary discretisation before building the blocks
   DISCRETISE   north0   2 0 0 0.0
   DISCRETISE   east0    2 0 0 0.0
   DISCRETISE   south0   2 0 0 0.0
   DISCRETISE   west0    2 0 0 0.0
   BLOCK  reactor  + north0  + east0  + south0  + west0
   FILL_BLOCK   reactor   initial
END_FLOW

BEGIN_CONTROL
   TITLE  Reactor  with  perfect  gas  mixture  of  N  and  N2.
   CASE_ID   0
   REACTING
   FLUX_CALC   ausmdv
   MAX_TIME    0.0002
   MAX_STEP    2000
   TIME_STEP   1.0e-7
   HISTORY_CELL    reactor   1 1
   DT_HISTORY   1.0e-6
END_CONTROL

# Name the output files and build them.
BEZIER_FILE   reactor.bez
PARAM_FILE    reactor.p
BUILD

EXIT
```

To run this problem, the script below reactor.sh was used. This can be found in the directory cfd/mb_cns/examples/reactor.

reactor.sh

```
#!/bin/sh
# Run the static reactor test case with finite-rate chemistry.

# 1. Prepare the finite-rate chemistry description
script_reac.py   -i nitrogen_2s2r.chm   -s species.dat   -r reactions.dat

# 2. Prepare the CFD description for MB_CNS
scriptit.exe   < reactor.sit   > reactor.log
mb_prep.exe   -f reactor

# 3. Run the simulation
time  mb_cns.exe   -f reactor
```

```
# 4. Extract some simulation data
mb_hist.exe -fi reactor.h -fo reactor.data -ncell 1 -nsp 2 -cell 0

# 5. Make some plots
gnuplot <<EOF
set term postscript eps enhanced 20
set output "reactor_T.eps"
set title "Static chemical reactor"
set xlabel "time, s"
set ylabel "T, K"
set xrange [0:0.0002]
set xtics 0.00005
plot "reactor.data" using 1:9 with lines
EOF

gnuplot <<EOF
set term postscript eps enhanced 20
set output "reactor_f.eps"
set title "Static chemical reactor"
set xlabel "time, s"
set ylabel "mass fractions"
set xrange [0:0.0002]
set xtics 0.00005
set yrange [0:1]
plot "reactor.data" using 1:10 title "N_2" with lines, \
     "reactor.data" using 1:11 title "N" with lines, \
     "equil.data" using 1:2 notitle with points 4, \
     "equil.data" using 1:3 notitle with points 5
EOF
```

Note that it is necessary to have the files `thermo.inp` and `trans.inp` in the working directory when `script_reac.py` is run. This is because data is used from the these CEA input files to construct the `species.dat` file.

5.2.1 Cylinder in dissociating nitrogen flow

The next test case considers the dissociating flow of nitrogen about a cylinder. The first case simulates a two-species (N_2 and N) mixture of nitrogen gas over a 90 mm diameter cylinder. The result is compared to a calculation made by Sebastian Karl (personal communication at DLR) using the CEVCATS code of the German Aerospace Center (DLR).

For the Mach 10 flow of nitrogen over a 90 mm cylinder, a body-fitted grid with 60 cells across the shock and 40 cells around the body was used. This grid is shown in Figure 4. The inflow conditions are a supersonic stream with:

$$p = 500.0\mathrm{Pa}, \quad T = 700.0\mathrm{K}, \quad f_{N_2} = 1.0, \quad f_N = 0.0.$$

The flow was treated inviscidly. The two dissocation reactions used earlier in the static reactor are used here. The reaction rates of Park [11] have been used.

The results of this simulation are shown in Figure 5 where the temperature along the stagnation line has been plotted. For a comparison, solutions have been computed for an ideal N_2 gas and one in chemical equilbrium. The nonequilibrium solution as part of this work compares very well to the solution from CEVCATS. The small discrepanices are

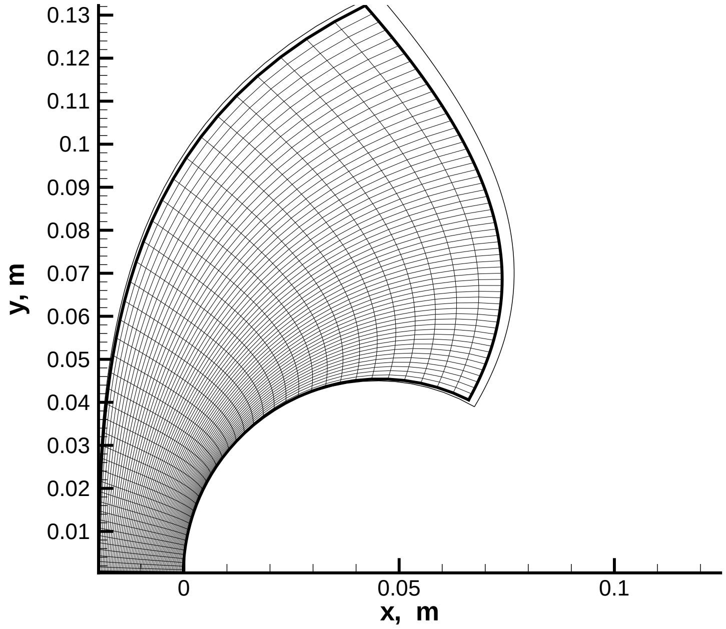


Figure 4: Grid for the 90 mm cylinder

most likely due to differences in the numerical schemes for both the fluid dynamics and the finite-rate chemistry.

In this simulation, the physics of the different gas models has behaved as would be expected. One can see that the ideal gas model vastly overpredicts temperature in the shock layer. This is because the internal energy is all stored in the gas' translational mode. The ideal gas model is a calorically perfect gas so none of the other internal energy modes of the molecule are accounted for. Furthermore, the ideal N_2 remains as diatomic nitrogen and has not been allowed to dissociate which would otherwise absorb some of the internal energy as chemical energy. In the equilibrium gas model, the relaxation processes are assumed to occur at an infinite rate. As such the energy modes have relaxed quicker than in the finite time taken in the nonequilibrium simulation. This leads to the lowered temperature in the shock layer. Shock stand-off distance is proportional to the inverse of the average density in the shock layer. The shock stand-off distances behave as expected for the different gas models. The stand-off distance is closely related to the temperature in the shock layer (through its effect on density) and thus should also be expected to be correct.

Setting up the problem The chemistry input file is identical to the one used in the static reactor problem and so is not be repeated here. Shown below is the shell script file `n90_run.sh` used to run this simulation.

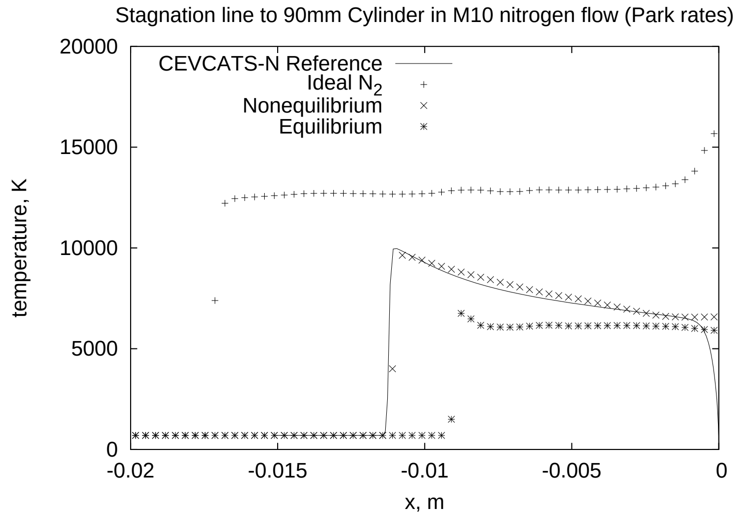


Figure 5: Temperature profiles along the stagnation streamline for a 90 mm diameter cylinder

n90_run.sh

```
# n90_run.sh
# Shell script to set up and run the cylinder simulation
#

# 1. Prepare the chemistry description
script_reac.py -i nitrogen_2s2r.chm -s species.dat -r reactions.dat

# 2. Prepare the CFD description
scriptit.exe < n90.sit > n90.log
mb_prep.exe -f n90

# 3. Run the simulation
time mb_cns.exe -f n90
```

The rest of the files associated with the post-processing can be found in the directory `cfd/mb_cns/n90/`.

5.3 Relaxation of nitrogen behind a shock wave

The test case considered here is the chemical relaxation of nitrogen species after the gas has been processed by 6.4 $\mathrm{km/s}$ shock. This problem was computed by Park [11] in an attempt to match the calculated emission spectrum of his model with that of a shock tube experiment. The results used here do not focus on the radiation results but rather on the nonequilibrium flowfield computed behind the shock. The conditions for this simulation are a free stream density, ρ_∞, of $1.498 \times 10^{-6}\ \mathrm{g/cm^3}$, room temperature (25 K) and a free stream velocity, u_∞, of $6.4\mathrm{km/s}$.

The reaction scheme used is that recommended by Gupta *et al* [12] for flows at speeds less than 8 $\mathrm{km/s}$. Five reactions were considered in the finite-rate chemistry model. The reactions include two heavy particle impact dissociations

$$\begin{aligned} N_2 + N_2 &\rightleftharpoons N + N + N_2 \\ N_2 + N &\rightleftharpoons N + N + N, \end{aligned}$$

the associative ionisation reaction

$$N + N \rightleftharpoons N_2^+ + N,$$

the charge exchange reaction

$$N_2 + N^+ \rightleftharpoons N_2^+ + e,$$

and the electron-impact ionisation

$$N + e \rightleftharpoons N^+ + e + e.$$

The results of this simulation are shown in Figure 6. This graph shows the relaxation of mole fractions behind the shock wave. For comparison purposes, the results of Park [11] are also included. The agreement is fairly good and provides validation of the finite-rate chemistry model. The present solution only uses a one-temperature model and so does not consider thermal nonequilibrium. Park has made use of a two-temperature model and as such the relaxation distance for the species evolution is slightly longer. This is to be expected because the vibrational temperature takes some finite-time to rise in Park's model. The vibrational temperature has an influence on the rate of the dissociation reactions and the lowered temperature means that dissociation occurs less rapidly. The important feature is that the present work has captured the general characteristics of the flow and has settled to the correct equilibrium values. The equilibrium values for species concentration were computed with the CEA program [5].

Setting up the problem This time the chemistry input file is more detailed as it needs to describe five reacting species and five reactions. This is shown here in `nitrogen_5s5r.chm`.

nitrogen_5s5r.chm

```
# nitrogen_5s5r.chm

# A five species model for nitrogen
#
# Reference:
# Gupta, R.N., Yos, J.M., Thompson, R.A. and Lee, K-P. (1990)
# A Review of Reaction Rates and Thermodynamic and Transport
# Properties for an 11-species Air Model for Chemical and
# Thermal Nonequilibrium Calculations to 30 000K.
# NASA RP-1232
#
# Rowan Gollan
# 13-Jan-03

SPECIES POLY
```

```
            N2  N  N2+  N+  e-
END

REACTIONS                qss              cms       k_b
N2  + N2  <=>   N  + N + N2           k_f    1.92e17       -0.5      1.131e5
                                      k_b    1.09e16       -0.5      0.0
N2  + N   <=>   N + N + N               k_f    4.15e22       -1.5      1.131e5
                                      k_b    2.32e21       -1.5      0.0
N   + e-  <=>   N+  + e-  + e-              k_f    1.5e32        -3.14     1.69e5
                                      k_b    1.5e40        -4.5      0.0
N2  + N+  <=>    N  + N2+             k_f    2.02e11        0.81     1.3e4
                                      k_b    7.8e11         0.5      0.0
N   + N   <=>   N2+  + e-              k_f    1.7e13          0.0      6.78e4
                                      k_b    1.0e22        -1.5      0.0
END
```

The scriptit file used in MB_CNS is also shown.

nit_relax.sit

```
# nit_relax.sit

# Relaxation   of  five  species   of  nitrogen   behind
# a  shock.
#
# Test  case  by  Chul  Park  for  his  multi-temperature     models.

BEGIN_GEOMETRY
   NODE  a    -5.0e-2    0.0
   NODE  b    -5.0e-2    2.0e-2
   NODE  c     0.0       0.0
   NODE  d     0.0       2.0e-2
   NODE  e     5.0e-2    0.0
   NODE  f     5.0e-2    2.0e-2

   LINE  ab  a  b
   LINE  ac  a  c
   LINE  bd  b  d
   LINE  cd  c  d
   LINE  ce  c  e
   LINE  df  d  f
   LINE  ef  e  f

   POLYLINE   n0    1 + bd
   POLYLINE   e0w1  1 + cd
   POLYLINE   s0    1 + ac
   POLYLINE   w0    1 + ab

   POLYLINE   n1    1 + df
   POLYLINE   e1    1 + ef
   POLYLINE   s1    1 + ce
END_GEOMETRY

BEGIN_FLOW
   # Gas  and  flow  propeties   for  a 5  species   mix  of  Nitrogen
   GAS_TYPE   perf_gas_mix
   #                                                 Species:    N2        N         N2+       N+        e-
   GAS_STATE   quiescent     133.322        0.0      0.0   298.0     1.0       0.0       0.0       0.0       0.0
   GAS_STATE   post_shock    57.649e3    5961.77     0.0  6562.23    0.65567   0.34418   4.5245e-5 1.0719e-4 5.0831e-9

   DISCRETISE   n0    200  0  0  0.0
   DISCRETISE   e0w1    2  0  0  0.0
   DISCRETISE   s0    200  0  0  0.0
   DISCRETISE   w0      2  0  0  0.0

   DISCRETISE   n1    200  0  0  0.0
   DISCRETISE   e1      2  0  0  0.0
   DISCRETISE   s1    200  0  0  0.0

   BOUNDARY_SPEC    w0  SUP_IN   post_shock
   BOUNDARY_SPEC    e1  SUP_OUT

   BLOCK   left  + n0  + e0w1  + s0  + w0
   BLOCK   right  + n1  + e1   + s1  + e0w1

   CONNECT_BLOCKS     left  east  right  west

   FILL_BLOCK    left  post_shock
   FILL_BLOCK    right  quiescent
END_FLOW
```

```
BEGIN_CONTROL
   TITLE  Nitrogen  relaxation  behind  a  shock
   CASE_ID  0

   REACTING
   PLANAR
   FLUX_CALC   adaptive
   MAX_TIME   0.4e-5
   MAX_STEP   20000
   TIME_STEP   1.0e-8
   CFL  0.125
   DT_PLOT   0.01e-5
END_CONTROL

# Name  the  output  files  and  build  them.
BEZIER_FILE    nit_relax.bez
PARAM_FILE     nit_relax.p
BUILD

EXIT
```

Again, all the necessary post-processing files are included in the directory `cfd/mb_cns/examples/nit_relax/`.

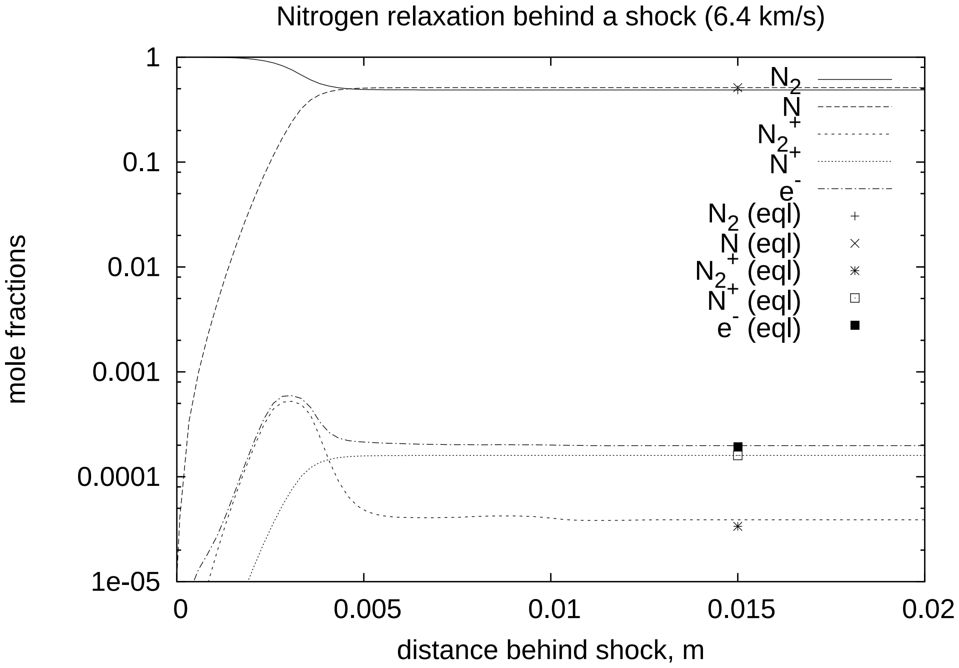
5.4 Summary for running a finite-rate simulation

Briefly outlined here is a summary of the steps involved in running a finite-rate simulation. This could be used as a checklist when setting up a new problem.

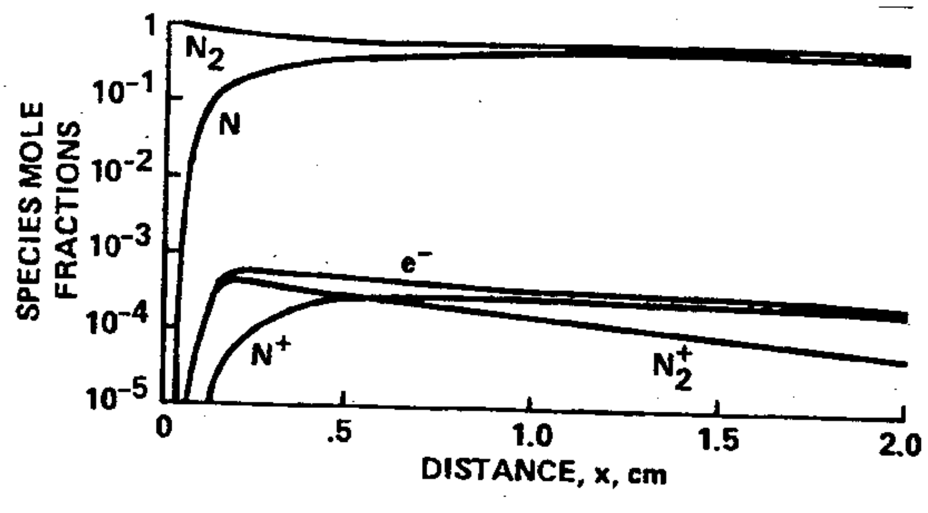
1. Decide on species present in simulation and the reactions they partake in. [1]
2. Gather the necessary forward and reverse rate coefficients from the relevant literature. [2]
3. Prepare an input chemistry file for the `script_reac.py` program. The files `thermo.inp` and `trans.inp` will need to be present from the CEA distribution.
4. Write a standard L1d or MB_CNS input file including the tags to run a finite-rate simulation.
5. Proceed as normal with a simulation using either L1d or MB_CNS.

[1] It is possible and quite likely that some inert species appear in the chemistry definition. In this case, the inert species will need to be declared as species but will not appear in the reaction definitions.

[2] In the not too distant future, the implementation will also allow for reverse rates compuetd from the equilibrium constant.



(a)



(b)

Figure 6: Relaxation of nitrogen species behind a shock wave at 6.4 $\mathrm{km/s}$ (a) present work, and (b) Park 2-temperature model [11].

References

[1] P.A. Gnoffo, R.N. Gupta, and J.L. Shinn. Conservation equations and physical models for hypersonic air flows in thermal and chemical nonequilibrium. NASA Technical Paper 2867, NASA, July 1989.

[2] P. A. Jacobs. Single-block Navier-Stokes integrator. ICASE Interim Report 18, 1991.

[3] E.S. Oran and J.P. Boris. *Numerical Simulation of Reactive Flow*. Cambridge University Press, 2001.

[4] J.D. Anderson Jr. *Hypersonics and High Temperature Gas Dynamics*. McGraw Hill, 1989.

[5] S. Gordon and B.J. McBride. Computer program for calculation of complex chemical equilibrium compositions and applications: I Analysis. NASA Reference Publication 1311, NASA, 1994.

[6] C.F. Gerald and P.O. Wheatley. *Applied Numerical Analysis*. Addison-Wesley, 6th edition, 1999.

[7] G. A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27(1):1–31, 1978.

[8] P. A. Jacobs. MB_CNS: A computer program for the simulation of transient compressible flows. Department of Mechanical Engineering Report 10/96, The University of Queensland, Brisbane, December 1996.

[9] T.R. Young and J.P. Boris. A numerical technique for solving stiff ordinary differential equations associated with the chemical kinetics of reactive-flow problems. *Journal of Physical Chemistry*, 81(25):2424–2427, 1977.

[10] D.R. Mott. *New Quasi-Steady-State and Partial Equilibrium Methods for Integrating Chemically Reacting Systems*. PhD thesis, The University of Michigan, 1999.

[11] C. Park. Assessment of a two-temperature kinetic model for dissociating and weakly ionizing nitrogen. *AIAA Journal of Thermophysics and Heat Transfer*, 2(1):8–16, 1988.

[12] R.N. Gupta, J.M. Yos, R.A. Thompson, and K.-P. Lee. A review of reaction rates and thermodynamic and transport properties for an 11-species air model for chemical and thermal nonequilibrium calculations to 30 000K. NASA Reference Publication 1232, NASA, 1990.

[13] M.T. Butler. Equilibrium thermo-chemical models for high-temperature gas dynamics. Undergraduate thesis, Divsion of Mechanical Engineering, Univeristy of Queensland, 2002.

A Format for the input file

An input file for the Python script `script_reac.py` contains two pieces of information: the species present in the mix; and the data describing the reactions in the mix. Comments are indicated by # symbol at the beginning of a line and blank lines are ignored.

A.1 Species declaration

LINE S.1
The first part of the file begins with a keyword `SPECIES` followed by an argument of `POLY` or `BEZ`. The `POLY` option will compute the thermodynamic properties based on the curve fits suggested by Gordon and McBride [5]. The `BEZ` option uses the Bezier curve fits found by Meghan Butler and further described in her thesis [13].[3]
LINE S.2+
On this line (and following lines) list the species present in the multi-species mix. The names for the species must be as they appear in the CEA file `thermo.inp`
LINE S.END
After all species are delcared, finish with the keyword `END`.

A.2 Reactions declaration

LINE R.1
The keyword `REACTIONS` begins the processing of the reactions declaration. It takes three mandatory arguments:

1. units of the rate constants
   - `CMS` for $\mathrm{cm}^3/\mathrm{mol}/\mathrm{s}$; or
   - `MMS` for $\mathrm{m}^3/\mathrm{mol}/\mathrm{s}$.
2. chemical integration scheme
   - `SAIM` for the Selected Asymptotic Integration Method;
   - `QSS` for the Quasi-Steady State method; or
   - `EULER` for the simple Euler method.
3. reverse rate calculation method
   - k_b for specified reverse rate constants; or
   - K_eq for reverse rates computed from the equilibrium constant. [4]

[3] At this point in time the Bezier curve fits need more care to setup and are an order of magnitude slower to use in terms of simulation time. Their advantage is an increased accuracy beyond 20,000K. Please see the author if it is necessary to use the Bezier curve fits.

[4] This option is implemented internally in the finite-rate chemistry module but has not been setup in a transparent manner for the end-user. See the author if this option is required.

LINE R.2a
On these lines reactions are specified as one would write out a reaction by hand. For example, the thermal dissociation of nitrogen could be expressed as:

```
N2 + N2 <=> N + N + N2
```

Not also that the notation:

```
N2 + N2 <=> 2 N + N2
```

is entirely equivalent when written by hand and is entirely equivalent in terms of the finite-rate chemistry module also.

Also on this line the three values for a modified Arrhenius form of the forward reaction rate coefficient must be given. They must be in the order A_f, B_f, T_{D_f} corresponding to

$$k_f = A_f T^{B_f} \exp(-T_{D_f}/T).$$

Note that the dissociation temperature is given as a positive quantity — the negative sign appears in the implementation. Conversely, the power for the temperature dependency must be given exactly as it is in the formulation (often this will be a negative number but not always).

LINE R.2b
This line contains either:

- the reverse rate values in an identical form to the forward rate values; or
- six coefficients for a least-squares curve-fit for the equilibrium constant [5]

LINE R.3+
The above format of two lines describing each reaction continues until all reactions have been listed.

LINE R.END
After all reactions have been specified, end with the keyword `END` .

[5] As mentioned earlier this is available but not easily usable for an end-user at this point in time