

How to use Chemkin-II

(Summary and condensed instructions for using the Chemkin-II and transport property databases, subroutine libraries, and application codes on the HP workstations)

**Submitted to:
Dr. Thomas Fletcher**

**By:
Jeffrey E. Davidson**

September 18, 1996

Table of Contents

Title Page	i
Table of Contents	ii
Introduction.....	1
Introduction to the HP Work Stations.....	3
The Tool Bar.....	3
Essential UNIX Commands.....	5
Other useful commands and tricks in Unix.....	8
Equilibrium Calculations using Chemkin-II.....	10
Instructions for using the Chemkin Equilibrium Solver.....	10
Example Problems for Chemkin Equilibrium Program.....	13
Laminar, Pre-mixed, One-dimensioned Flame Calculations.....	14
Setting up a Premixed Flame Problem.....	14
Example Problems for Premixed Laminar Flame.....	20
Perfectly Stirred Reactor Calculations.....	22
Setting up a PSR problem.....	22
Example Problems for the Perfectly Stirred Reactor.....	27
Extracting Transport Data from Chemkin	28
Instructions for Extracting Transport Properties	28
Transferring data from the HP workstations to a PC.....	31
Programing using Chemkin-II	32
Calculations using Global Mechanisms.....	36
Calculations Using Global Mechanisms.....	37
Installing the Chemkin Programs into your Directory.....	38
References.....	39
Appendix.....	40

How to use Chemkin-II

Introduction

Chemkin-II is a collection of data bases and subroutines written in text files and in FORTRAN code for solving problems involving gas-phase kinetics, equilibrium and transport properties. There are many different types of problems that can be solved using the Chemkin subroutines. Driver programs (programs that direct the subroutines and control the input/output files) have been written for the following types of problems:

1. Adiabatic flame temperatures.
2. Equilibrium concentrations for a gas mixtures.
3. Species mole fractions as a function of time in laminar, steady state, premixed flames when temperature profile is known.
4. Flame speed, species mole fractions and temperature profile as a function of time in laminar, steady-state, premixed flames.
5. Mole fractions and temperature of product stream from a perfectly stirred reactor.
6. Determination of transport properties of gas mixtures as a function of composition, temperature and pressure.

This is just a small sampling of the numerous driver programs that have been written. The number and type of computer programs which incorporate Chemkin subroutines are constantly increasing.

To use these programs, the user must modify several input files. As will be explained in detail below, each of the input files is in a Unix-shell file and the user will only need to modify the shell file. In order for the subroutines to work, the user must tell Chemkin about the gas mixture and the gas phase reactions. This is done in the file named *mech*. The user must name each of the elements and each of the species which are in the gas mixture. The user must also enter the elementary steps of the chemical reactions being considered with their respective Arrhenius rate coefficients for the equation:

$$k_f = AT^\beta \exp\left(\frac{E}{RT}\right) \quad (1)$$

The user must also specify the problem by modifying the *inp* file. This file generally contains information such as initial temperature of the gas mixture, mole fractions of the reactants, and initial pressure. Depending on the program and the nature

of the problem, the user will need to include other keywords which control the program. As these are specific to each of the programs, they will be discussed separately below. Once the *inp* and the *mech* files have been properly modified, the program can be run by executing the shell file.

For each of the types of problems discussed here, the application code has already been written. The user must only supply the reaction description and the application input. The application input varies from application to application and each will be discussed individually below. The Chemkin applications have been setup and modified to run the HP workstations called the STATES located in room 308 CB. An introduction to these machines will be discussed before the discussion of the applications.

Introduction to the HP Work Stations

The HP work stations are powerful computers which operate much faster than the typical PCs. (Some of these programs would require a few days to run on a PC or MAC). All of the STATES are large enough that multiple users can be logged onto the same state. For example, if you sit down at either the TEXAS or the DALLAS terminal, you will be automatically logged into the TEXAS computer. It is possible to login to UTAH from TEXAS or from a PC or Macintosh via TELNET (a communication software package) or by using the command **rlogin state** from another workstation, but the graphical interface will not be available.

These machines use HP-UNIX for an operating system which incorporates both text and graphical interfaces. When you sit down at one of the terminals, if the screen is blank, move the mouse. The login screen will appear and ask for your user's name and password. After you enter both of these correctly, the screen will be blank except for a border and a tool bar.

The Tool Bar

Mail- If you have new or unread E-mail, this button will show a couple of envelopes. Press this button and a listing the mail stored in your mail box will appear. You can use this program to read and send mail.

CPU Load- This figure shows the load on that state's CPU. As the load increases the area plot increases. Horizontal lines mean that the scale on the original screen had to be reduced. If the computer has more than 3 or 4 horizontal lines it might be wise to find another computer that does not have quite the work load.

One, Two, Three, Four, Five, Six- These are multiple screens from which you can access your files. If the first screen (One) gets too cluttered with windows, you can switch back and forth between the other screens with these buttons.

Printer- Text files can be sent to the laser printer in the next room by clicking the file in the file manager screen with the middle button on the mouse and then dragging the file to this button. Clicking the left mouse button while the mouse is on the printer button will display the printer status.

Text Editor- This button looks like pencil on a piece of paper. When this button is pressed, a window containing the text editor is opened. This editor behaves much like the MS-DOS text editor or Edit on the Macintosh.

X-Term- Text user interface. When you press this button, a screen will pop up with a cursor. From this screen, the user can enter commands to copy, delete, rename and run various files. It is similar to a DOS screen on a PC. The commands are discussed below.

Style Manager- The attributes (colors, background, etc.) of the graphical interface can be changed using the options under this button.

Rename Workspace- Allows you to give names to the One, Two, Three, Four, Five, Six screens discussed above.

? - This is the help button which contains a more detailed explanation of the HP work station environment.

File Manager- This button (looks like a file cabinet) puts a window on the screen which displays all the files in the current directory. If you double-click on a text file in this window, the text editor will pop up with the contents of the file.

Trash- Drag files that you want to erase from your directories on to this button and they will be erased when you logout.

Exit - Press this button to logout. The computer will then ask for conformation.

Essential UNIX Commands

In this section, only the most essential UNIX commands will be presented. This presentation is not intended to be detailed but should be sufficient to survive in HP-UNIX. Other commands and further detail can be found in the books found in the computer laboratory. Several of the commands presented here can also be done using the graphical interface although I have found the text interface to have more options and be more useful. The information for the different commands will be described in terms of function and format or will be presented in a tutorial format.

To begin, press the X-Term text interface button with the left-hand button on the mouse. If you were logged-on to TEXAS, a screen would pop-up with the prompt:

texas\$

Help

HP-UNIX has an on-line help program which has a very detailed description of the use and options of the UNIX commands. To access the help program type:

man *command* [Enter]

Where *command* would be the name of the command for which you had a question. For example, if you typed:

man pwd [Enter]

a help screen will pop up and describe the **pwd** command. To scroll down in help use the space bar and to exit help press *q*.

Directories

Files can be stored in tree-like directories. That is, any directory can have sub-directories. To find out which directory path you are in, type **pwd [Enter]**. The **print working directory** command will show you the absolute path name of the working directory. Whenever you start a new text interface, you will automatically be placed in your main directory. The output for the **pwd** command should be:

cheme/username

All your files should be kept within this directory or sub-directories of *username*. To see what files or sub-directories you have in this directory type **ls [Enter]** for *List files*. The command **ls -l** (l is a lowercase L) will provide more information about the files and sub-directories.

To make a sub-directory use the **mkdir *directoryname*** command. For example at the prompt, type **mkdir junk [Enter]**.

Now type **ls** to see the directory that you created.

To move down to that directory, type **cd junk** [Enter]. The **change directory** command can be used to move up and down in the directory tree. To see in which directory you are working, type **pwd** [Enter]. The result should read:

cheme/username/junk

Now, to move back to your main directory, use the move-up- one-directory command, **cd ..** [Enter]. The two periods mean the parent directory or the directory directly above the current directory. Type **pwd** [Enter] to make sure you are now in your main directory.

To remove the *junk* directory, type **rmdir junk** [Enter]. In order to remove a directory, the directory must be empty of all files and sub-directories.

Note: Every time you call up a text interface screen, it will automatically open to your main directory. In order to access any of your programs or files in other directories, you must first change to the directory in which the files are located.

Files

There are three basic types of files used by Chemkin. Text files can be edited by the text editor. They cannot be executed unless they are a program source code and are first compiled. Shell files are denoted with the **.sh** extension are similar to **.bat** file in MS-DOS. They are text files which can be executed and will direct the compilation of FORTRAN codes, direct a sequence of program calls, and direct file input and file output. They can be edited by the text editor and executed by typing:

sh shellname.sh

If the **.sh** file is already in the execute attribute (see File...Properties in graphical interface File Manager) then the shell file can be executed by:

shellname.sh

For the Chemkin programs for which shell files have been written, you will not need to do any programming in FORTRAN but will need to modify the shell files.

The third type of file is a non-text executable file. These files are created by compilers and cannot be edited by a simple text editor. Since all the compilation of the FORTRAN source code will be directed by the shell files, you do not need to directly run these programs.

There are several commands which are very useful in manipulating files. To make a copy of a file use the **cp** command. When using the format **cp file1 file2**, the computer will make a copy of *file1* and call it *file2*. This will not destroy *file1*, but will destroy any old copies of *file2*. To copy a file into another directory

use the format **cp** *file1 path/directoryname*. This will place a second copy of *file1* into *directoryname*. To copy all the files in one directory to another use **cp** **path/directoryname*.

To move or rename files and directories use the **mv** command. This has the same format as the **cp** command except, the original file is destroyed. For example, if you typed **mv** *file1 file2*, *file2* would be a copy of *file1* and *file1* would be erased.

To erase a file using the text screen, use **rm** *filename*. If you use **rm -i** *filename*, the computer will request conformation before a file is deleted.

Other than using the text editor, text files can be viewed using the **more** command. (Format **more** *filename*). This command will display the contents of a text file to the screen and will pause before scrolling. Because the **more** command only allows you to scroll down, it is often better to use the text editor to view files.

Other useful commands and tricks in Unix

To change your password, type **passwd** at the prompt.

To run a program in the background add an “&” to the end of the file name.

When a program is running in background, you can log-off the system and the program will keep going. To execute the premixed flame shell in the background, you would type:

sh premix.sh &

If the program is already running but not in the background, press `[ctrl]-[z]` to pause the program, and type **bg** at the prompt to put the program in the background.

Once a file is running in the background, you can follow its progress by a number of ways. It is often necessary to follow some of the Chemkin programs, because they tend to crash if the initial guesses are far from the true solution. To see if it is still running type **jobs -l** (lower case L) at the prompt. This will list all the programs currently being executed from that window. To list all the programs currently being executed on that computer type **ps -e**. (Sometimes this list can be long so to control the screen output use **ps -e | more**. The | will pipe the output from the **ps** command through the **more** command.) Both the **ps** and the **jobs** commands will list a number associated each program in execution. This number can be used to stop the execution of the program with the **kill** command. For example, if you want to kill program number 1, type **kill %1**.

The contents of the output file can be followed as the file is being generated by using the **tail** command. For example, to follow the output from the premixed flame shell program, type **tail -f premix.out**. To exit the tail command press `[ctrl]-[c]`. When shell files are executed, their progress is written to the file `.log`. To follow the `.log` file, use **tail -f .log**.

If your program stops running but the output file is empty, look at the files `ckout` and `tpout` (in the text editor) and they will tell you if there are any errors in your `mech` or `inp` files.

If your workstation should freeze, press `[shift]-[ctrl]-[reset]`. This will reset your computer. These buttons are on the left side of the keyboard. Any files that were not previously saved will probably be lost.

To print files, you can either use the text editor or use the **lp** command. The format is **lp filename**. This will print two pages per piece of paper. This command and the printing from the text editor will truncate any line which is over 80 columns. To print files that are wider than 80 columns do the following:

1. **psf -1sc 150 <filename>junk**
2. **lp junk**

Be careful when you are printing some of the *.out* files because they can be quite long and can contain a lot of unnecessary information. In most cases it will be better to first erase the unwanted information by using the text editor and then printing the file. Also be careful not to send non-text files to the printer or hundreds of pages could be wasted.

From certain STATES or terminals a file cannot be directly printed. If your file does not print after sending it to the printer you will need to **rlogin** on ALASKA and print from that machine. Do the following on a *X-term* window of any machine in the room:

1. **rlogin alaska**
2. **cd** to directory where your file is located. (You can use **pwd** find out in which directory you are located.
3. **lp filename**
4. **logout**

There are many other commands available in UNIX. Please refer to a manual to learn other commands.

Equilibrium Calculations using Chemkin-II

The Chemkin equilibrium software is easy to use but is not as powerful as other equilibrium programs like the NASA-LEWIS code or the EDCONV code. The latter two programs, will search their data bases to determine what species should be considered in an equilibrium calculation. The Chemkin code requires that the user input all the species which are to be considered. In the example of methane combustion in air, NO and NO₂ formation will not be considered unless those two species are entered into the program. CO will not even be considered unless it is input into the program.

Instructions for using the Chemkin Equilibrium Solver

1. Login to an HP workstation.
2. Open the Text Editor.
3. Under the File pull-down menu, select Open...
4. Open equil.sh (you may need to change directories)
5. Scroll down to the line below “cat << EOF > mech”.
6. After the word “ELEMENTS” enter the symbols of every element that will be in any species that should be considered in the equilibrium calculation. Isotopes can be considered if their atomic weight follows their symbol in slashes “/”. Lines beginning with an exclamation mark are ignored. Conclude the elements section with the word “END”. Example:

```
ELEMENTS
N
H
! Define Deuterium
D /2.014/
O
END
```

Also Acceptable is:

```
ELEMENTS N H D /2.014/ O END
```

7. After the word “SPECIES”, enter the symbols of every species that should be considered. The format is similar to the elements section. Example:

```
SPECIES H2 O2 H O OH HO2 N2 N NO END
```

If species which are not contained in the thermodynamic data base need to be considered, check out the manual entitled Chemkin-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics. This manual will provide more detail on extra features of the Chemkin software package. Thermal dynamic data for numerous hydrocarbon combustion species are available in the correct format for the database in

Combustion Chemistry, edited by William C. Gardiner or from the NASA-LEWIS equilibrium code data file.

8. Scroll past the driver program to the line: “cat << EOF > inp”

9. The shell file generates a file named *inp* that contains the keyword inputs to specify the problem for the equilibrium program.

10. The following keyword inputs are allowed in any order:

REAC - Specifies the reactants and their molar quantity or mole fraction.

Format: REAC H2 2

CONH - Constant enthalpy can be used in conjunction with constant volume or pressure, but not with constant temperature, entropy or internal energy.

CONP - Constant pressure can be used with constant temperature, volume, enthalpy or entropy, but not with constant internal energy.

CONT - Constant temperature can be used with constant pressure, volume or entropy but not with enthalpy, and constant internal energy.

CONV - Constant volume can be used with constant temperature, pressure, internal energy, enthalpy or entropy.

CONU - Constant internal energy can be used with constant volume only.

CONS - Constant entropy can be used with constant temperature, pressure or volume, not with constant internal energy or enthalpy.

CONX - Constant mole fraction

CHAP - Chapman-Jouguet- cancels all the above specification except REAC. This keyword causes the detonation-wave velocities to be calculated.

TEMP - Required input for the starting temperature in K. Format: TEMP
298

TEST - Estimate equilibrium temperature in K.
Format: TEST 2500

PRES - Starting pressure in atmospheres. Format: PRES 1

PEST - Estimate of equilibrium pressure in atmospheres.
Format: PEST 1.5

CNTN- Tells the computer to look for an additional problem after the keyword END.

END - End of keyword input.

An example of a input file *inp* to calculate the equilibrium concentrations of a stoichiometric hydrogen flame in air and a fuel rich hydrogen flame is:

REAC H2 2

REAC O2 1

```
REAC N2 3.76
CONH
CONP
TEMP 300
TEST 2000
PRES 1
CNTN
END
/ This is the second problem.
REAC H2 3
REAC O2 1
REAC N2 3.76
END
```

11. If you would like to customize the output file name, you must edit the file name on the second to last line in the shell file.

Example:

Before:

```
make equile; equile < inp > equil.out
```

After:

```
make equile; equile < inp > filename
```

12. Under the File pull-down menu of the text editor, select the Save or Save as.. command.

13. Open a text interface window by clicking on the X-term button on the tool bar.

14. Change the directory to the directory where the Chemkin files are located using the **cd** command.

15. Run the shell program by typing: sh equil.sh

16. When the cursor reappears, the program is completed and the output file can be viewed and printed via the text editor.

Example Problems for Chemkin Equilibrium Program

1. Calculate the adiabatic flame temperature and equilibrium composition of CO in air with equivalence ratios of 0.5, 1, 2. (Do not consider NO_x formation). (Feed conditions- 300 K, 1 atm). Consider using the CNTN keyword to save some time.
2. Repeat problem 1 but consider NO_x formation.
3. Repeat problem 2, but make the inlet feed contain 10 percent H₂.
4. Compare the results from problem 3 with either the EDCONV program or the NASA-LEWIS code. If you forgot a significant species in problem 3 as determined by one of the other two programs, repeat problems 3 and 4 again. What are two reasons for using the Chemkin equilibrium program, if one of the other equilibrium codes is available?

Laminar, Pre-mixed, One-dimensional Flame Calculations

“Many practical combustors, such as internal combustion engines, rely on pre-mixed flame propagation. Moreover, burner-stabilized laminar premixed flames are very often used to study chemical kinetics in a combustion environment. Such flames are effectively one-dimensional and can be made very steady, thus facilitating detailed experimental measurements of temperature and species profiles. Also, laminar flame speed is often used to characterize the combustion of various fuel-oxidizer combinations. Therefore, the ability to model chemical kinetics and transport processes in these flames is critical to interpreting flame experiments and to understanding the combustion process itself.” (Kee, 1992)

The Chemkin premixed flame code is designed to handle a variety of problems. The calculations involve solving systems of non-linear mass balances, energy equations, and transport relations. There are two major divisions in the types of problems that the premixed code will solve. The burner stabilized problem requires a known mass flow rate, and the temperature profile can either be specified or calculated from energy equations. Often it is better to specify the temperature profile if it can be experimentally determined because heat losses from a flame can be difficult to quantify. Problems where the temperature is specified are relatively easy to solve compared to the second type of problem. The second type of problem involves an adiabatic freely propagating flame. From the input, the flame speed, temperature profile, and concentration profile are calculated. This type of problem is very difficult to converge and solving them requires some skill, experience and a lot of patience. The code is currently configured for a burner diameter of 1.0 cm².

Setting up a Premixed Flame Problem

1. Login to an HP workstation.
2. Open the Text Editor.
3. Under the File pull-down menu, select Open...
4. Open premix.sh
5. Scroll down to the line below “cat << EOF > mech”.
6. After the word “ELEMENTS” enter the symbols of every element that will be in any species that should be considered in the equilibrium calculation. Isotopes can be considered if their atomic weight follows their symbol in slashes “/”. Lines beginning with and exclamation mark are ignored. Conclude the elements section with the word “END”. Example:


```
ELEMENTS
N
H
! Define Deuterium
D /2.014/
O
END
```

Also Acceptable is:

```
ELEMENTS N H D /2.014/ O END
```

7. After the word “SPECIES”, enter the symbols of every species that should be considered. The format is similar to the elements section. Example:

```
SPECIES H2 O2 H O OH HO2 N2 N NO END
```

If species which are not contained in the thermodynamic data base need to be considered, check out the manual entitled Chemkin-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics. This manual will provide more detail on extra features of the Chemkin software package. Thermodynamic data for numerous hydrocarbon combustion species are available in the correct format for the database in Combustion Chemistry, edited by William C. Gardiner. Transport property data may also need to be added to the transport property data base.

a. This section must begin with the word REACTIONS. The default units for the Arrhenius rate coefficients are cal/mole for the activation energy and the units of A are in terms of cm, sec, K, and moles. Equation 1 is shown here again:

$$k_f = AT^{\beta} \exp\left(\frac{E}{RT}\right) \quad (1)$$

These default units of the activation energy can be changed by adding the words CAL/MOLE, KCAL/MOLE, JOULES/MOLE or KELVINS after the word REACTIONS. The default units for A can be changed by adding MOLECULES so that the units of A are in terms of cm, sec, K, and molecules.

b. The information of the reaction is fairly free from format but certain conventions must be observed. At the beginning of the line the reaction mechanism must be symbolically written. No more than 3 molecules can be on either side of the equality. All coefficients of reactants must be integers. A plus sign (+) is used to separate the reactants from the reactants and the products from the products. There are two symbols used to separate the reactants from the products. The equals sign (=) is used to represent a reversible reaction. When this symbol is used, the reverse reaction is considered and calculated using the equilibrium constant. When the symbol => is used, the reaction is assumed to be irreversible and the reverse reaction is not considered. Third body reactions can be considered by using the symbol M as both a reactant and product. The

M species does not need to be declared in the SPECIES section. On both sides of the equation, M must be the last species. The default third body efficiency is one, but this can be modified as shown in the example elementary step. The three Arrhenius coefficients (in order of A, β , and E) are on the same line as the reaction and must be separated by at least one space. (Multiple spaces or tabs are acceptable for separating the data on a line.) Exclamation marks (!) signify that information following the ! is to be ignored. An example elementary step is:

```
H + O2 + M = HO2 + M          0.361E18          -0.72  0.0
H2O/18.6/ H2/2.86/ N2/1.26/  ! 3rd body efficiencies
```

Refer to the manual, [Chemkin-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics](#) for details on handling photo-chemical reactions, pressure-dependent fall-off reactions and Landau-Teller formulation of rate expressions.

c. End the mechanism description with the word END.

8. Scroll down to the line: “cat << EOF > inp”

9. The shell file generates a file named *inp* that contains the keyword inputs to specify the problem for the premixed laminar flame program. In the Appendix there are several pages directly from the manual and explain all the possible keyword inputs. Some of the most important keywords will be discussed here.

BURN - A required keyword for burner-stabilized flame problems.

FREE - Required for adiabatic freely propagating flame. (Note: either BURN or FREE must be specified by not both but this can be changed on continuation (CNTN) or restart (RSTR).)

TGIV - For burner-stabilized flames, the species equations will be solved using the user-specified temperature profile.

ENRG - The temperature profile will be calculated using the energy equations, but the user must enter initial guesses for a temperature profile using the TEMP keyword. (Note: Either TGIV or ENRG must be specified for BURN type problems.)

MOLE or **MASS** - Used to specify that answers and inputs are in either mole or mass fractions.

PRES - Pressure of the flame in atmospheres. Format: PRES 1.0.

FLRT - Specified flow rate for burner stabilized flames or initial guess for adiabatic freely propagating flames. Units are in g/cm²-sec. Format: FLRT 0.04.

REAC - used to specify initial reactant mole or mass fractions.

INTM - used to give the computer initial guesses of intermediate species mole fractions.

PROD - used to give the computer initial guesses of product species mole fractions

TEMP - used to give a temperature profile as a function of distance along the flame. The units are in cm and K. The temperature profile must be in order starting with the lowest distance along the flame. The temperature can only increase with distance.
Format: TEMP 1.25 1567

CNTN - (continue) Used to solve another problem using the solution of the first. This is very useful when trying to solve the adiabatic freely propagating flame. Once a problem has converged, this keyword can be used to slowly step away from the converged problem and solve similar problems.

RSTR- (restart) Use this keyword to start calculations based on a previously computed solution. As premix proceeds towards a converged solution, it saves intermediate solutions of FORTRAN file unit 14. To use this option you must do the following:

- a. Converge a case using premix.
- b. Delete the following command at the end of the premix shell file: *rm save recover*.
- c. Type: *cp restart recover*.
- d. Include RSTR as a keyword in the next premix input file.
- e. Run *sh premix.sh*.

ASEN- Causes the first order sensitivity coefficients of the temperature, mole/mass fractions, and mass flux with respect to reaction rates to be computed. The results are stored in the *save* file and a separate program *senpremixe* can be used to convert the binary *save* file to a text file. See instruction 16 below.

See problems 1 and 3 for example Keyword inputs.

11. If you would like to customize the output file name, you must edit the file name on the second to last line in the shell file.

Example:

Before:

```
make premix; rm save recover; premix < inp > premix.out
```

After:

```
make premix; rm save recover; premix < inp > filename.out
```

12. Under the File pull-down menu, select the Save.

13. Open a text window by clicking on the X-term button on the tool bar.

14. Change the directory to the directory where the Chemkin files are located using the **cd** command.

15. Run the shell program by typing: **sh premix.sh** or **sh premix.sh &** to run the program in the background.

16. When the program is completed, the output file can be viewed using the text editor and printed via the text editor or using the **lp** command. The output can be further analyzed using *senpremixe*. This program will read the solution for each converged case (*ie.* if CNTN is used) and write the result to a semi-colon separated text file. This text file can then be easily read into a spread sheet or other graphing program. (Note: you will need to set the import text option to semicolon separated.) *senpremixe* will also write the sensitivity coefficients to a semicolon separated file if ASEN was specified. To use this program:

a. type *sh senpremix.sh*

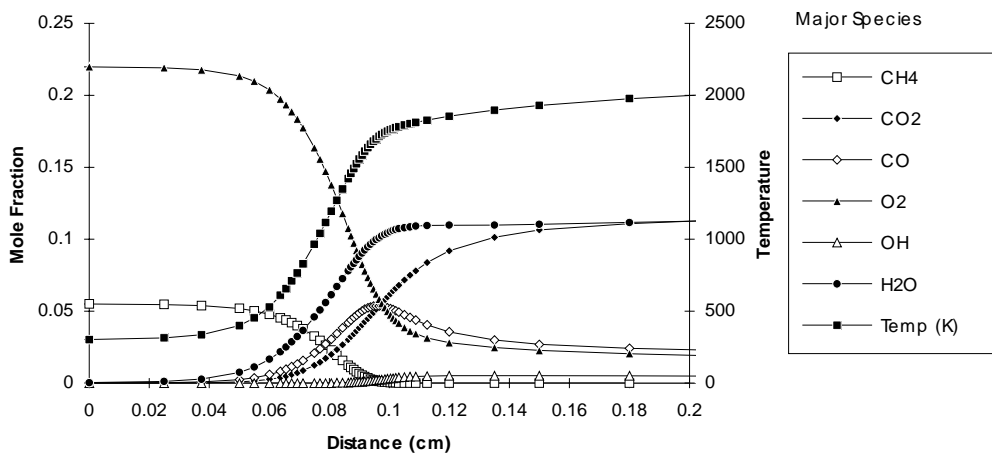
b. type *senpremixe*

senpremixe can normalize the sensitivity coefficients by two methods. The first method is recommended for temperature and mass flux and the second method is recommended for mass/mole fractions. These normalization methods are discussed in the Premix manual-- "A Fortran Program for Modeling Steady Laminar One-Dimensional Premixed Flames." This program (*senpremixe*) will also sort the reactions from most important to least important for a given variable if desired.

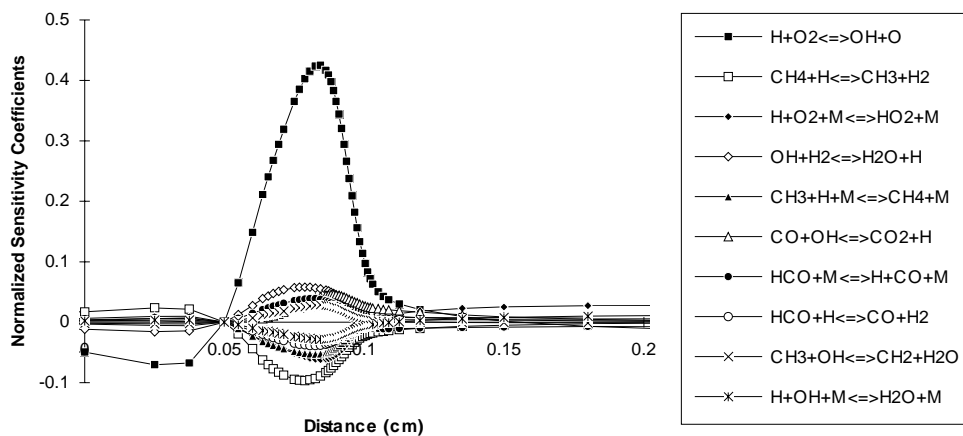
17. See the discussion above (in Other useful commands...) about following the progress of the program. The premix program is very sensitive and will crash unless the initial guesses are sufficiently close to the answer.

18. The following plots are examples of possible post processing of *premixe* output. These calculations are based on a methane in air flame. The energy equation and flame speed were calculated (ENRG and FREE). The sensitivity coefficients (ASEN) were also calculated.

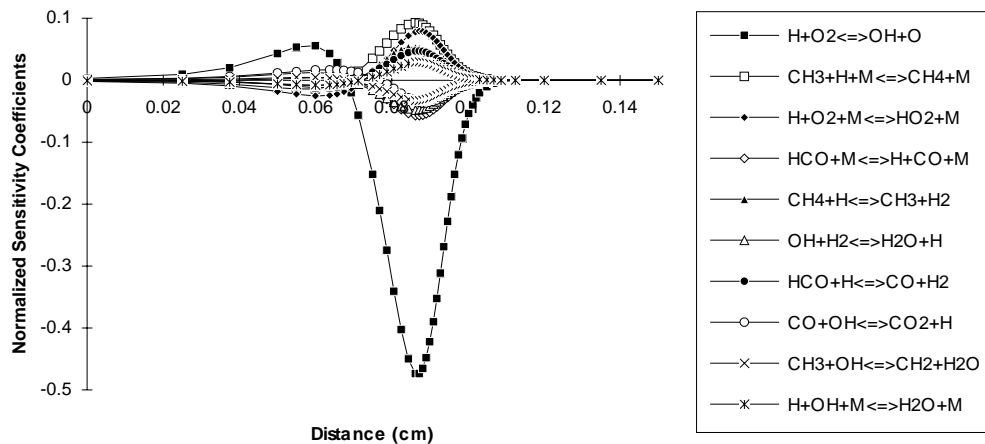
Free CH4 flame in Air calculated from Premix



Most important reactions affecting Temperature in CH4 flame (1atm)



Top reactions affecting CH4 in methane flame



Example Problems for Premixed Laminar Flame

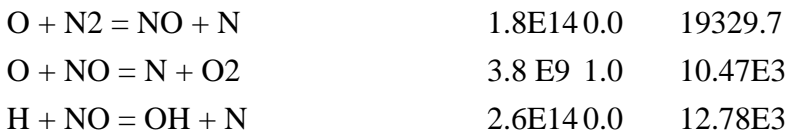
1. Calculate the species mole fractions along a one-dimensional hydrogen in oxygen and argon flame. The temperature profile is known (given in the *inp* file) and the energy balance equations do not need to be solved (use keyword TGIV). Also given: Initial flow rate 0.00463 g/cm²-sec, flame pressure 0.0329 atmospheres. This simple problem is designed to introduce you to some of the keywords and input requirements for the premix flame code. (The *inp* file and the *mech* file for this problem are in the Appendix.)

Answer the following questions:

- a. What is the mole fraction of H at 6 cm?
- b. What do GRAD and CURV tell the equation solver?
- c. What are XEND, XCEN and WMIX?

2. Repeat problem 1, but instead of Ar use N₂. a) include thermal NO formation elementary steps in the mechanism. b) do not include thermal NO formation elementary steps in the mechanism. Remember to change the initial guesses of the intermediate and product species. Does including the thermal NO steps significantly change the results? Would you want to take a breath of the cooled exhaust gas from this flame? NO concentrations above 60 ppm can irritate the nose and throat. The thermal NO formation mechanism in the same units as the H₂ mechanism is:

! Thermal NO_x formation mechanism



3. Calculate the adiabatic flame speed for a methane in air. Ignore thermal NO_x formation. Use the *mech* and *imp* files found in the Appendix for this problem. Study the output file. Discuss the differences between this input file and input file in problem 1. What does the CNTN keyword do? Why do you think the CNTN keyword was repeated three times in the adiabatic freely propagating flame problem but not in the problem where the temperature was specified? (What is being done in the continuation runs?)

4. Plot the results from problem 3 in three figures. In one figure plot the flame speed and temperature as a function of distance along the flame. In another figure plot the concentrations of the reactants and the major products. In the third figure, plot the concentrations of the minor intermediates and products. (See instructions on how to transfer the data to a PC or MAC).

5. Modify the mechanism file and keyword input file in problem 3 to calculate the flame speed and species concentrations for stoichiometric CO in O₂ flame a) with and b) without H₂O (mole fraction H₂O = 0.0322). Is water or hydrogen a necessary part of the CO combustion mechanism? What effect does hydrogen/water have on the flame speed? The CO combustion mechanism can be found in Fossil Fuel Combustion, edited by Bartok and Sarofim, 1991, pages 222-3. Pay attention to the units. The units on the Arrhenius constants were calculated with concentration units on the third bodies.

6. **Expert Problem:** Calculate the hydrogen in air flame speeds for equivalence ratios 1, 1.3, 1.6 and 2. Hint: Because this problem has trouble converging, you may need to begin with a converged solution from problem 5 and use the CNTN function to step (use small steps) to the desired reactant concentrations. How do these results compare with those shown in Figure 6, page 344 of Bartok and Sarofim. (Bring something else to do when you run this program. It can take over an hour to solve!)

Perfectly Stirred Reactor Calculations

Data collected from perfectly stirred reactors (PSR) can be interpreted by modeling the elementary chemical rate processes.* The Chemkin code assumes a small thermally insulated chamber with both a feed stream and a product stream. After the fuel and oxidizer reach the reactor, the feed is infinitely mixed, instantaneously. The walls of the chamber are non-catalytic and the flow through the reactor is characterized by a nominal residence time.

For the Chemkin code, the user may specify the reactor volume and mass flow rate or the reactor volume and residence time. The heat loss can be specified or a fixed reactor temperature can be used. The program uses equilibrium conditions for the initial guess of the solution. The use of the PSR code is very similar to that of the equilibrium and laminar premixed flame codes.

Setting up a PSR problem

1. Login to an HP workstation.
2. Open the Text Editor.
3. Under the File pull-down menu, select Open...
4. Open psr.sh
5. Scroll down to the line below “cat << EOF > mech”.
6. After the word “ELEMENTS” enter the symbols of every element that will be in any species that should be considered in the equilibrium calculation. Isotopes can be considered if their atomic weight follows their symbol in slashes “/”. Lines beginning with and exclamation mark are ignored. Conclude the elements section with the word “END”. Example:

```
ELEMENTS
N
H
! Define Deuterium
D /2.014/
O
END
```

Also Acceptable is:

```
ELEMENTS N H D /2.014/ O END
```

7. After the word “SPECIES”, enter the symbols of every species that should be considered. The format is similar to the elements section. Example:

```
SPECIES H2 O2 H O OH HO2 N2 N NO END
```


If species which are not contained in the thermodynamic data base need to be considered, check out the manual entitled Chemkin-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics. This manual will provide more detail on extra features of the Chemkin software package. Thermal dynamic data for numerous hydrocarbon combustion species are available in the correct format for the database in Combustion Chemistry, edited by William C. Gardiner or from the NASA-LEWIS data file. Transport property data may also need to be added to the transport property data base.

a. This section must begin with the word REACTIONS. The default units for the Arrhenius rate coefficients are cal/mole for the activation energy and the units of A are in terms of cm, sec, K, and moles. Equation 1 is shown here again:

$$k_f = AT^\beta \exp\left(\frac{E}{RT}\right) \quad (1)$$

These default units of the activation energy can be changed by adding the words CAL/MOLE, KCAL/MOLE, JOULES/MOLE or KELVINS after the word REACTIONS. The default units for A can be changed by adding MOLECULES so that the units of A are in terms of cm, sec, K, and molecules.

b. The information of the reaction is fairly free from format but certain conventions must be observed. At the beginning of the line the reaction mechanism must be symbolically written. No more than 3 molecules can be on either side of the equality. All coefficients of reactants must be integers. A plus sign (+) is used to separate the reactants from the reactants and the products from the products. There are two symbols used to separate the reactants from the products. The equals sign (=) is used to represent a reversible reaction. When this symbol is used, the reverse reaction is considered and calculated using the equilibrium constant. When the symbol => is used, the reaction is assumed to be irreversible and the reverse reaction is not considered. Third body reactions can be considered by using the symbol M as both a reactant and product. The M species does not need to be declared in the SPECIES section. On both sides of the equation, M must be the last species. The default third body efficiency is one, but this can be modified as shown in the example elementary step. The three Arrhenius coefficients (in order of A, β , and E) are on the same line as the reaction and must be separated by at least one space. (Multiple spaces or tabs are acceptable for separating the data on a line.) Exclamation marks (!) signify that information following the ! is to be ignored. An example elementary step is:



H2O/18.6/ H2/2.86/ N2/1.26/ ! 3rd body efficiencies

Refer to the manual, Chemkin-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics for details on handling photo-chemical reactions, pressure-dependent fall-off reactions and Landau-Teller formulation of rate expressions.

c. End the mechanism description with the word END.

8. Scroll down to the line: "cat << EOF > inp"

9. The shell file generates a file named *inp* that contains the keyword inputs to specify the problem for the perfectly stirred reactor program. In the Appendix are several pages from the manual and explain all the possible keyword inputs. Some of the most important keywords will be discussed here (a complete listing is included in the appendix).

TGIV - The reactor is operating at a user specified temperature.

ENRG - The reactor temperature is calculated from the energy equations.

Either TGIV or ENRG must be specified.

TEMP - Either the actual reactor temperature (TGIV case) or the initial guess of the reactor temperature (ENRG case). (Temperature units = K) Format: TEMP 1900

PRES - The pressure of the reactor in atmospheres. Format: PRES 2.0

TAU - The nominal residence time in the reactor in seconds. Either TAU or FLRT must be specified. Format: TAU 3.0

FLRT - Mass flow rate through the reactor in g/sec. Format: FLRT 2.6

VOL - The volume of the reactor in cm³. Required input. Format: VOL 1.0

QLOS - The heat loss from the reactor in cal/sec. This is a required input for ENRG problems. Format: QLOS 15000.0

TINL - The inlet temperature in K. Required input for ENRG problems. Format: TINL 300.0

REAC - mole fraction values of the unburned reactants. One REAC keyword must appear for each reactant species. Format: REAC C2H2 0.45.

EQUI - The fuel/air equivalence ratio. This is used if the keywords FUEL/OXID/PROD is used instead of REAC. Format: EQUI 1.25

FUEL - Like the keyword REAC, but used to specify the composition of the fuel. The following example means that ethane is the only fuel. Format: FUEL C2H6 1.0

OXID - Like the keyword REAC, but used to specify the composition of the oxidizer. For air the following keywords would be used:

OXID O2 0.21

OXID N2 0.79

PROD - Used to specify the product species. This is required so that the program can calculate the feed concentration from EQUI. Format: PROD CO2

ADD - Used to specify inert(s) mole fraction if OXID/FUEL/EQUI keywords are used.

PRNT - Specifies the detail of the results. There are three levels: 0, 1, and 2. Level 0 prints the answers only and Level 2 prints all the intermediate iterations. Format: PRNT 2

CNTN - (continue) Used to solve another problem using the solution of the first. This is very useful when trying to solve a problem involving a small reactor and tau. Once a problem has converged for a large tau, this keyword can be used to step away from the converged problem to a problem of a shorter tau.

ASEN - The sensitivity coefficients with respect to the rate constant for all species and temperature. The data will be written to the out put file and written to *save* for post processing. (See *senpsre* below.)

AROP - Will calculate the rate of production for each species by each reaction and write it to the output file and to the save file for post processing. (See *senpsre* below.)

11. If you would like to customize the output file name, you must edit the file name on the second to last line in the shell file.

Example:

Before:

```
make psre; rm save recover; psre < inp > psr.out
```

After:

```
make psre; rm save recover; psre < inp > filename.out
```

12. Under the File pull-down menu, select the Save.

13. Open a text window by clicking on the X-term button on the tool bar.

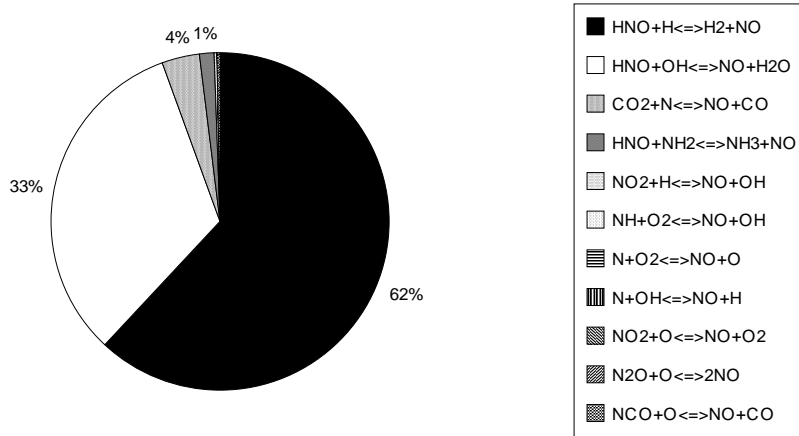
14. Change the directory to the directory where the Chemkin files are located using the **cd** command.

15. Run the shell program by typing: **sh psr.sh** or **sh psr.sh &** to run the program in the background.

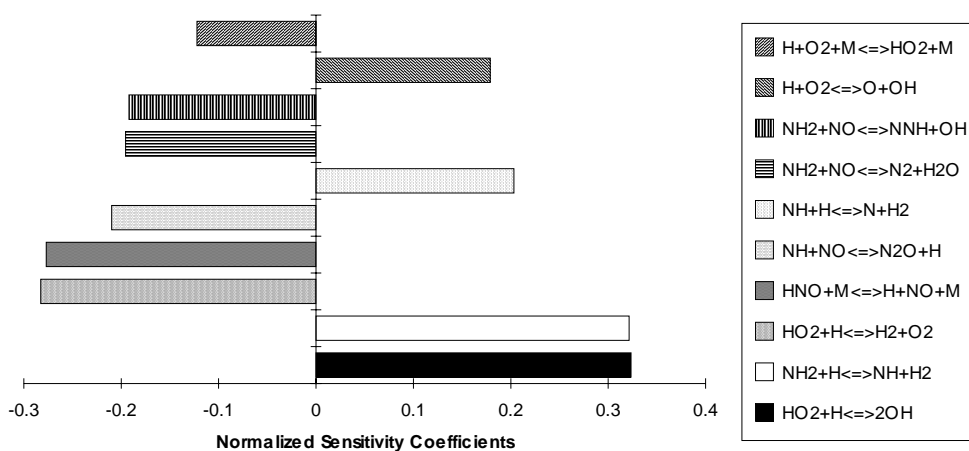
16. When the program is completed, the output file can be viewed using the text editor and printed via the text editor or using the lp command.

17. The output can also be analyzed using *senpsre*. This program will read the *save* file (which includes the solution, sensitivity coefficients (if ASEN) and rate of production (if AROP) for each case (see CNTN above)). As *senpsre* reads *save* it will ask the user for output file names each time new data is found and write the data in semicolon separated text. These text file can then be read into a graphing program or spread sheet program for post processing. Sample post processing for NH₃/NO_x reactions could include:

Fractional Contribution to the Formation of NO (from AROP)



Top 10 rxns Affecting NO (from ASEN)



Example Problems for the Perfectly Stirred Reactor

1. Determine the product composition and temperature of a hydrogen flame in air. The volume of the reactor equals 67.4 cm^3 . The residence time equals $3.0\text{E-}5$ seconds. The reactor operates adiabatically. a) Equivalence ratio = 1.0, b) equivalence ratio = 1.2. (Note: The mechanism and input files have already been entered into the shell file. All you need to do is run the psr.sh file and examine the results.)

2. Using the mechanism and reactor description of problem 1, determine the residence time required for 90% conversion of hydrogen. Equivalence ratio = 1.0. What is the residence time for 100 percent conversion?

Extracting Transport Data from Chemkin

With the Chemkin thermodynamic data base and subroutines, there is a transport property data base. The program *tranprop.f* was written to extract the transport properties of pure gases and of gas mixtures. It is rather straight forward to use and is similar to the other Chemkin programs like the equilibrium, premix, and psr codes. The transport data base was developed for the temperature range 300 to 5000 K, but most values between 273 and 300 K are still valid.

Instructions for Extracting Transport Properties

1. Login to an HP workstation.
2. Open the Text Editor.
3. Under the File pull-down menu, select Open...
4. Open tranprop.sh
5. Scroll down to the line below “cat << EOF > mech”.
6. After the word “ELEMENTS” enter the symbols of every element that will be in any species that should be considered in the equilibrium calculation. Isotopes can be considered if their atomic weight follows their symbol in slashes “/”. Lines beginning with an exclamation mark are ignored. Conclude the elements section with the word “END”. Example:

```
ELEMENTS
N
H
! Define Deuterium
D /2.014/
O
END
```

Also Acceptable is:

```
ELEMENTS N H D /2.014/ O END
```

7. After the word “SPECIES”, enter the symbols of every species that should be considered. The format is similar to the elements section. Example:

```
SPECIES H2 O2 H O OH HO2 N2 N NO END
```

If species which are not contained in the thermodynamic data base need to be considered, check out the manual entitled Chemkin-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics. This manual will provide more detail on extra features of the Chemkin software package. Thermal dynamic data for numerous hydrocarbon combustion species are available in the correct format for the

thermodynamic database in Combustion Chemistry, edited by William C. Gardiner or from a NASA LEWIS equilibrium code database.

8. Scroll past the driver program to the line: “cat << EOF > inp”

9. The shell file generates a file named *inp* that contains the keyword inputs to specify the conditions of the gas mixture.

10. The following keyword inputs are allowed in any order:

REAC - Specifies the reactants and their molar quantity or mole fraction.

If the sum of the mole fractions is not equal to one then the program will normalize the mole fractions.

Format: REAC H2 2

TINT - Specifies the initial temperature. *Tranprop.f* will calculate the transport properties for a temperature range by using TEND and TINC. Format: TINT 300

TEND - Specifies the final temperature of the temperature range. *Tranprop.f* will calculate the transport properties for a temperature range by using TINT and TINC. If only one temperature is required set TEND to the same temperature as TINT. Format: TEND 300

TINC - Specifies the increment to be used for the temperature range from TINT to TEND. For example, if transport properties are desired at 300, 400 and 500 K, the temperature keywords would be:

TINT 300

TEND 500

TINC 100

PRES - Specifies the pressure of the problem in atmospheres. Format: PRES 1.5

VISP - Will print the pure component viscosities.

VISM - Will print the mixture viscosity.

CNDP - Will print the pure component thermal conductivity.

CNDM - Will print the mixture thermal conductivity.

DIFB- Will print the binary diffusivities for all the gases in the mixture.

DIFM - Will print the mixture-averaged diffusion coefficients.

DIFC - Will print the ordinary multi-component diffusion coefficients.

TDFL - Will print the thermal diffusion ratios for the light species in the mixture (MM<=5.0)

TDFM - Will print the thermal diffusion coefficients and mixture thermal conductivity.

CPSP Will print the pure component constant pressure heat capacity.
CPSM - Will print the mixture constant pressure heat capacity.
CVSP - Will print the pure component constant volume heat capacity.
CVSM - Will print the mixture constant volume heat capacity.
PRNS - Will print the pure component Prandtl number.
PRNM - Will print the mixture Prandtl number.
END - End of keyword input.

An example of a input file *inp* follows these instructions.

11. If you would like to customize the output file name, you must edit the file name on the second to last line in the shell file.

Example:

Before:

```
make tranprope; tranprope < inp > tranprop.out
```

After:

```
make tranprope; tranprope < inp > filename
```

12. Under the File pull-down menu of the text editor, select the Save or Save as.. command.

13. Open a text interface window by clicking on the X-term button on the tool bar.

14. Change the directory to the directory where the Chemkin files are located using the **cd** command.

15. Run the shell program by typing: sh tranprop.sh

16. When the cursor reappears, the program is completed and the output file can be viewed and printed via the text editor.

An example *inp* file is found in the Appendix.

Transferring data from the HP workstations to a PC

Data files generated by the Chemkin programs can be transferred to the Caedm network so that they can be accessed by either a PC or MAC. Spread Sheet programs like EXCEL can then be used to efficiently plot your results. To transfer your data from a text file on the STATES to Caedm, follow these steps:

1. Login to one of the States.
2. Use the **cp** command to make a copy of the file you wish to transfer.
3. Use the text editor to erase all the unnecessary data from the file to be transferred.
4. On a X-term screen enter **ftp** at the cursor.
5. In the ftp program open the caedm network by entering:
open caedm
6. Type **cd{path}username** to move to the directory where you will deposit your file. The commands **DIR** or **ls** can be used to see where you are.
7. Enter **ascii** to tell the computer to generate an ascii file.
8. Type **put filename** to send your data file to caedm.
9. Exit *ftp* by typing **bye**.
9. Logout from the STATE, and login to Caedm at a MAC or PC terminal.
10. Run EXCEL and open your file using the semicolon separated text format option.
11. Plot your data and remember to save the file in the normal format if you want to keep your plots with the file.

Programming using Chemkin-II

The core to Chemkin-II consists of a thermodynamic database and a library of FORTRAN-77 subroutines for calculating gas phase properties and reaction rates. In addition to the thermodynamic calculations, a second database and subroutine library have been developed for determining the multicomponent transport properties of gas phase mixtures. These subroutines can be easily incorporated into programs which can solve a variety of gas-phase problems. There are several important steps in using the Chemkin subroutines. The purpose of this section is only to provide an introduction and summary to the necessary steps in using the Chemkin subroutines. The programmer will need to refer to Chemkin-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics and A FORTRAN Computer Code Package for the Evaluation of Gas-Phase Multicomponent Transport Properties when writing the actual code. The relationship between the various files and programs is shown in Figure 1.

Before any of the Chemkin or transport subroutines can be called, the Chemkin Interpreter (*ckinterpe*) must be executed. This is a separate program which reads the *mech* file (renamed as *ftn15*) and extracts the necessary information for the specific problem from the thermodynamic database. The format for the *mech* file is as in this example:

```
ELEMENTS H O END
SPECIES H2 O2 H2O H O OH H2O2 END
REACTIONS
H2+O2=2OH          1.7E13      0.0   47780.0
2OH=O+H2O         6.0E8       1.3   0.0
/ COMMENT DENOTED BY A "/"
/ OTHER REACTIONS AS NECESSARY
END
```

Each element and species must be specified. Each of the elementary step reactions (if your program uses Chemkin subroutines which require reaction data) must be listed. (Refer to Chemkin-II... for details).

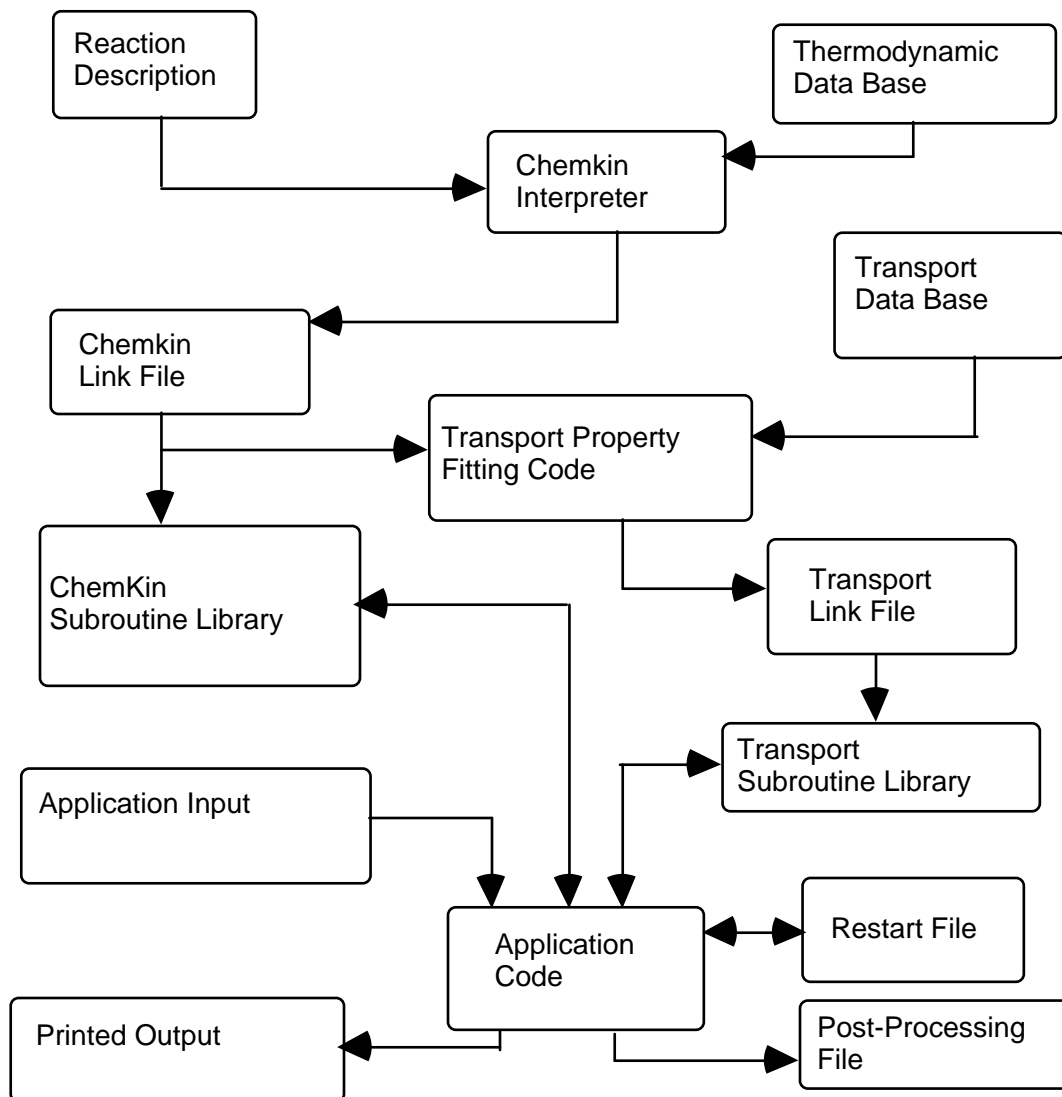


Figure 1: Relationship between Chemkin files. (adapted from Kee, 1992).

The interpreter checks for possible errors in the mechanism and creates a link file (unit 25) which is used by the transport property fitting code and the Chemkin initialization subroutine. If transport subroutines are going to be used in the application code, the user must then execute the transport property fitting code (*tranfite*). *tranfite* extracts the necessary information from the transport database and writes the transport link file (unit 35). The transport link file is used by the transport initialization subroutine (called in the user's application code).

After *ckinterpe* and *tranfite* have been executed, the user may execute the application code. There is a general format which must be followed at the beginning of the application code. For the Chemkin subroutines, there are three arrays which must be

dimensioned (RCKWRK, ICKWRK, and CCKWRK). If the transport subroutines are going to be used then arrays RMCWRK and IMCWRK must also be dimensioned. (RMCWRK and RCKWRK are real arrays; IMCWRK and ICKWRK are integer arrays; and CCKWRK is a character array.) The minimum length of these arrays depends on the size of the problems as defined in the *mech* file. If the arrays are dimensioned improperly then the program will stop and a error message will be displayed in either *ckout* or *tpout*.

The Chemkin link file and the transport link file need to be opened and then the two initialization subroutines CKINIT and MCINIT need to be called. Because the format is fairly general, the following program header can be used for most applications. The specific application code is contained in the user-defined subroutine APPLICATION.

```

PROGRAM SAMPLE
C23456789012345678901234567890123456789012345678901234567890123456789012
C SAMPLE PROGRAM HEADER WHICH DEMONSTRATES THE INITIALIZATION
C SUBROUTINES FOR CHEMKIN AND THE TRANSPORT DATA
C
C IMPLICIT REAL*8 (A-H,O-Z), INTEGER (I-N)
C
C LENICK IS THE LENGTH OF THE CHEMKIN INTEGER WORKING ARRAY
C LENRCK IS THE LENGTH OF THE CHEMKIN REAL WORKING ARRAY
C LENCCK IS THE LENGTH OF THE CHEMKIN CHARACTER ARRAY
C LENRMC IS THE LENGTH OF THE TRANSPORT REAL ARRAY
C LENIMC IS THE LENGTH OF THE TRANSPORT INTEGER ARRAY
C LINKCK IS THE FILE NUMBER OF THE CHEMKIN LINK FILE
C LINKMC IS THE FILE NUMBER OF THE TRANSPORT LINK FILE
C KDIM IS THE MAXIMUM NUMBER OF SPECIES
C MDIM IS THE MAXIMUM NUMBER OF ELEMENTS
C LOUT IS THE STANDARD OUTPUT FILE NUMBER
C
C PARAMETER (LENICK=5000, LENRCK=20000, LENCCK=100, LENRMC=20000,
1 LENIMC=5000, LINKCK=25, LINKMC=35, KDIM=30, MDIM=20,
2 LOUT=6)
C
C KSYM IS A CHARACTER ARRAY OF THE SPECIES NAMES
C ATOM IS A CHARACTER ARRAY OF THE ELEMENT NAMES
C CCKWRK IS THE CHEMKIN CHARACTER WORKING ARRAY
C
C CHARACTER*16 KSYM(KDIM), ATOM(MDIM), CCKWRK(LENCCK)
C
C ICKWRK IS THE CHEMKIN INTEGER WORKING ARRAY
C RCKWRK IS THE CHEMKIN REAL WORKING ARRAY
C IMCWRK IS THE TRANSPORT INTEGER WORKING ARRAY
C RMCWRK IS THE TRANSPORT REAL WORKING ARRAY
C
C DIMENSION ICKWRK(LENICK), RCKWRK(LENRCK), IMCWRK(LENIMC),
1 RMCWRK(LENRMC)
C
C OPEN(LINKCK, FORM='UNFORMATTED', FILE='cklink')
C OPEN(LINKMC, FORM='UNFORMATTED', FILE='tplink')
C
C INITIALIZE CHEMKIN SUBROUTINES
C CALL CKINIT(LENICK, LENRCK, LENCCK, LINKCK, LOUT, ICKWRK,
1 RCKWRK, CCKWRK)
C

```

```

C      INITIALIZE TRANSPORT SUBROUTINES
C      CALL MCINIT(LINKMC, LOUT, LENIMC, LENRMC, IMCWRK, RMCWRK)
C
C      MM IS THE TOTAL NUMBER OF ELEMENTS IN MECHANISM
C      KK IS THE TOTAL NUMBER OF SPECIES IN MECHANISM
C      II IS THE TOTAL NUMBER OF REACTIONS IN MECHANISM
C      NFIT IS THE NUMBER OF COEFFICIENTS IN FITS TO THERMODYNAMIC DATA
C
C      CKINDX RETURNS A GROUP OF INDICES DEFINING THE SIZE OF THE
C      PARTICULAR REACTION MECHANISM
C
C      CALL CKINDX(ICKWRK, RCKWRK, MM, KK, II, NFIT)
C
C      CHECK TO SEE IF PROGRAM IS DIMENSIONED PROPERLY
C      IF (KDIM.LT.KK) THEN
C          WRITE(LOUT,*) 'KDIM MUST BE AT LEAST ',KK
C          STOP
C      ENDIF
C      IF (MDIM.LT.MM) THEN
C          WRITE(LOUT,*) 'MDIM MUST BE AT LEAST ',MM
C          STOP
C      ENDIF
C
C      RUN SPECIFIC PROBLEM AS DEFINED BY USER, IE SUBROUTINE
C      APPLICATION MUST BE WRITTEN BY THE PROGRAMMER
C
C      CALL APPLICATION(ICKWRK, RCKWRK, CCKWRK, IMCWRK, RMCWRK, LOUT,
1          KK, KDIM, KSYM, MM, MDIM, ATOM, II)
C
C      END OF PROGRAM
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
END

```

In the subroutine APPLICATION, the programmer can call numerous subroutines from either the Chemkin library or the transport property library. The specific information for each of the subroutines is contained in the manuals listed above. All of the variables used by these subroutines are in cgs units. (dynes,ergs,cm,K,sec)

When compiling the application code, the following libraries must be linked:

cklib.f, *tranlib.f*, *twopnt.f* and *dmath.f*.

In summary, to use the Chemkin and transport subroutines, the programmer must:

1. Build the *mech* file and name it *fn15*.
2. Execute *ckinterpe*.
3. Execute *tranfite*.
4. Set up the application code with a header similar to that shown in

SAMPLE.

5. Compile application code with links to the necessary subroutine

libraries.

Calculations Using Global Mechanisms

Global mechanisms generally involve fewer species and fewer reaction steps than elementary step mechanisms. Calculation time can be reduced considerably by using a four-step global mechanism as compared to an 80-step elementary mechanism. In general, global mechanisms are written for narrow temperature and pressure ranges and specific feed concentrations. Deviations from the conditions for which the global mechanism was developed lead to erroneous results or cause the numerical calculations to fail.

In general, the Chemkin subroutines do not handle global mechanisms very well, and must be fooled into thinking that the global mechanisms are elementary reactions. Several different approaches are available. The method described here was developed by Jeff Davidson, a PhD student in Chemical Engineering (1994-1996).

Premix was modified to calculate the reaction rates from a subroutine **GETRATES** rather than from the Chemkin interpreter. The source code is in *premix2.f*. Error checking must also be included in case a negative concentration is sent to the file. This program is very sensitive and will only converge for reaction conditions and mechanisms which immediately exhibit a high change in temperature. This program should be used only in narrow temperature ranges. There is an example mechanism located in the **GETRATES** subroutine which can be used as a pattern for programming other mechanisms. The feed conditions and all the other program inputs (discussed in the premixed flame program section) can be modified in the shell file *premixglobal.sh*. This example case in this shell file does not converge. Trial and error is required to get this to work but it should.

Installing the Chemkin Programs into your Directory

The Chemkin files which you will need to access are located on Dr. Fletcher's hard drive. They can be loaded to your area by following these instructions.

1. Login to one of the *states*.
2. At a text interface (X-Term or HP-Term if at a work station) type: **pwd**. Write down your working path directory.

3. Type:

mkdir chemkin [Enter]

cd /cheme/tom/chemkin [Enter]

4. Type: **cp * /userpath/username/chemkin** where the user path and user name are the result from the **pwd** command in step 2.

5. Type **cd /userpath/username/chemkin**

6. Type **ls** to make sure the files have been transferred.

References

Bartok, William and Adel F. Sarofim. Fossil Fuel Combustion: A Source Book John Wiley & Sons, Inc. 1991.

Glarborg, P., Kee, Grcar, and Miller. PSR: A Fortran Program for Modeling Well-Stirred Reactors. *Gov Pub*. SAND86-8209. Feb 1992.

Kee, R. J., F. M. Rumpely, and J.A. Miller. Chemkin-II: A Fortran Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics *Gov. Pub*. SAND89-9009B. Sep. 1989.

Kee, R. J., Dixon-Lewis, Warnatz, Coltrin, and Miller. A Fortran Computer Code Package for the Evaluation of Gas-Phase Multicomponent Transport Properties *Gov. Pub*. Sand86-8246. Jul 1992.

Kee, R. J., Grcar, Smooke, Miller. A Fortran Program for Modeling Steady Laminar One-Dimensional Premixed Flame. *Gov Pub*. SAND85-8240. Apr 1992.

Kee, R. J., Rumpely and Miller. The Chemkin Thermodynamic Data Base *Gov Pub*. SAND-8215B. Oct 1992.

Kiehne, Thomas M., Matthews, Wilson . "An Eight-Step Kinetics Mechanism for High Temperature Propane Flames" *Combust. Sci. and Tech*. 1987, Vol. 54. pp 1-23

Appendix

Key Words for Premixed Flame Code (Premix).....	A-F
Key Words for Perfectly Stirred Reactor (PSR).....	G-M
Premixed Flame Sample Problem 1 <i>inp</i> and <i>mech</i> files	N
Premixed Flame Sample Problem 2 <i>inp</i> and <i>mech</i> files	O-P
Sample Keyword input file for Tranprop.sh.....	Q
Answer Key to Sample Problems and selected problems from Bird Stewart and Lightfoot	R→