



NRL/MR/6721--98-8164

MLTE An LTE Solver for Diatomic Molecules and Atoms above 6,000 °K

PAUL KEPPLER
JOHN GIULIANI
MARGARET MULBRANDON

*Radiation Hydrodynamics Branch
Plasma Physics Branch*

July 15, 1998

19980717 020

Approved for public release; distribution is unlimited.

DEMO QUALITY INSTRUMENTED 8

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE July 15, 1998	3. REPORT TYPE AND DATES COVERED		
4. TITLE AND SUBTITLE MLTE An LTE Solver for Diatomic Molecules and Atoms Above 6,000 °K		5. FUNDING NUMBERS		
6. AUTHOR(S) Paul Kepple, John Giuliani, and Margaret Mulbrandon				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Research Laboratory Washington, DC 20375-5320		8. PERFORMING ORGANIZATION REPORT NUMBER NRL/MR/6721--98-8164		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Ballistic Missile Defense Organization 7100 Defense Pentagon Washington, DC 20301-7100		10. SPONSORING/MONITORING AGENCY REPORT NUMBER		
11. SUPPLEMENTARY NOTES This research was sponsored by the Ballistic Missile Defense Organization under Job Order Title, "Material Plasma Processing."				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.		12b. DISTRIBUTION CODE		
13. ABSTRACT (Maximum 200 words) An algorithm is presented for calculating the equilibrium distribution of a mixture of atoms, ions, diatomic molecules, molecular ions, and electrons. The technique, which is a generalization of the Saha equation to include molecules, enables evaluation above 6,000 °K. This is the limit of the NASA Lewis data tables for thermodynamic functions. Finally, the heat capacity of various species are presented up to 20,000 °K which allows fits to extend standard tabular data.				
14. SUBJECT TERMS Molecular LTE C_p		15. NUMBER OF PAGES 61		16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

TABLE OF CONTENTS

	List of Figures	iii
I	Introduction	1
II	Theory of MLTE	3
IIa	Sum over States	9
III	Users Guide	11
IIIa	Examples	13
IV	Fortran Sample Driver for MLTE	29
V	Atomic and Molecular Data	34
VI	Specific Heat	40
	References	50
	Appendix A: Least Square Fit With a Constraint	52

LIST OF FIGURES

- Fig. 1 LTE distribution of species for equal parts carbon and oxygen at a pressure of 10 Torr. and temperatures between 3,500 and 8,000 °K.
- Fig. 2 Same as Fig. 1 but at 100 Torr.
- Fig. 3 Same as Fig. 1 but at 1,000 Torr.
- Fig. 4 Same as Fig. 1 but at 10,000 Torr.
- Fig. 5 Same as Fig. 1 but for temperatures between 8,000 and 12,500 °K.
- Fig. 6 Same as Fig. 5 but for 100 Torr.
- Fig. 7 Same as Fig. 5 but for 1,000 Torr.
- Fig. 8 Same as Fig. 5 but for 10,000 Torr.
- Fig. 9 LTE distribution of species for equal parts of nitrogen and oxygen at a pressure of 10 Torr and temperatures between 1,500 and 8,000 °K.
- Fig. 10 Same as Fig. 9 but for 100 Torr.
- Fig. 11 Same as Fig. 9 but for 1,000 Torr.
- Fig. 12 Same as Fig. 9 but for 10,000 Torr.
- Fig. 13 Same as Fig. 9 but for temperatures between 8,000 and 14,500 °K.
- Fig. 14 Same as Fig. 13 but for 100 Torr.
- Fig. 15 Same as Fig. 13 but for 1,000 Torr.
- Fig. 16 Same as Fig. 13 but for 10,000 Torr.
- Fig. 17 LTE distribution of species for nitrogen at a pressure of 7.5 Torr and temperatures between 2,000 and 14,000 °K.
- Fig. 18 Same as Fig. 17 but for 75 Torr.
- Fig. 19 Same as Fig. 17 but for 750 Torr.
- Fig. 20 Same as Fig. 17 but for 7,500 Torr.
- Fig. 21 LTE distribution of species for 10:1 argon to nitrogen at a pressure of 10 Torr and temperatures between 1,500 and 14,000 °K.
- Fig. 22 Same as Fig. 21 but for 100 Torr.
- Fig. 23 Same as Fig. 21 but for 1,000 Torr.
- Fig. 24 Same as Fig. 21 but for 10,000 Torr.
- Fig. 25 Cumulative C_p / R as contributions from successive electronic states are included for the case of O_2 .
- Fig. 26 C_p / R for O_2 . Solid line is from MLTE, dashed line from [Ref. 11].
- Fig. 27 Same as Fig. 25 for the case of O_2^+ .
- Fig. 28 Same as Fig. 26 for O_2^+ .
- Fig. 29 Same as Fig. 25 for the case of CO .
- Fig. 30 Same as Fig. 26 for CO .
- Fig. 31 Same as Fig. 25 for the case of CO^+ .
- Fig. 32 Same as Fig. 26 for CO^+ .

- Fig. 33 Same as Fig. 25 for the case of N_2 .
- Fig. 34 Same as Fig. 26 for N_2 .
- Fig. 35 Same as Fig. 25 for the case of N_2^+ .
- Fig. 36 Same as Fig. 26 for N_2^+ .
- Fig. 37 Same as Fig. 25 for the case of NO .
- Fig. 38 Same as Fig. 26 for NO .
- Fig. 39 Same as Fig. 25 for the case of NO^+ .
- Fig. 40 Same as Fig. 26 for NO^+ .
- Fig. 41 Same as Fig. 25 for the case of C_2 .
- Fig. 42 Same as Fig. 26 for C_2 .
- Fig. 43 Same as Fig. 26 for O .
- Fig. 44 Same as Fig. 26 for O^+ .
- Fig. 45 Same as Fig. 26 for N .
- Fig. 46 Same as Fig. 26 for N^+ .
- Fig. 47 Same as Fig. 26 for C .
- Fig. 48 Same as Fig. 26 for C^+ .
- Fig. A1 Φ_2 . Part of the (approximately) orthogonal basis set for least square fit.
- Fig. A2 Φ_3 “
- Fig. A3 Φ_4 “
- Fig. A4 Φ_5 “

MLTE AN LTE SOLVER FOR DIATOMIC MOLECULES AND ATOMS ABOVE 6,000 °K

I. Introduction

Consider an isolated system containing a mixture of various species. Under the condition of thermodynamic equilibrium the particle number of each species N_i , is that set of numbers which is most probable to be observed in the system subject to certain constraints. These constraints are conservation of two of the following three: mass, volume, or pressure. Thermodynamic equilibria is appropriate to systems at high pressure and for timescales long compared to $1/(\text{reaction rate})$. Such conditions generally hold in automobile combustion engines, waste treatment plants, and atmospheric arc torches. Even in systems which are not in thermodynamic equilibrium, such as low pressure inductively coupled plasma processing reactors, the rapid calculation of species composition assuming equilibria provides a first order guess to further kinetic calculations. Furthermore, one often finds that while an entire collection of species is not in thermodynamic equilibria, certain subsets of the species are, due to the rapid reaction rates among the members of the set. This state is often referred to as one in partial equilibria.

According to Besmann [Ref.1], there have been two basic approaches to solving for the most probable set of abundances $\{N_i\}$. Brinkley [Ref.2,3] in the 1940's determined chemical equilibrium constants for each chemical equation, while White, et al., [Ref.4] developed in 1958 an approach using Lagrange multipliers to directly minimize the free energy of the system and obtain thermodynamic equilibria. A good description of this technique from 1968 is presented by Zeleznik and Gordon [Ref.5]. Since this time several generations of "standard" equilibria solvers for use on mainframe computers have emerged, among them are SOLGASMIX [Ref.1] and CHEMKIN [Ref.6]. To evaluate the free energy of a system one needs the thermodynamic data for each species in the mixture. A standard reference for the required functions, such as specific heat capacity, is McBride, Gordon, and Reno [Ref.7]. Today there are readily available software packages for Personal Computers which come with their own data base for several thousand compounds.

A study of the tables in Ref.7 for the thermodynamic quantities used in minimizing the free energy reveals that the values for each compound listed in the table are limited to temperatures less than 6000 K. This upper limit is certainly reasonable for any research in combustion processes. However, there are conditions in reactors used for plasma processing which easily exceed this limit. For instance, Mehmetoglu and Gauvin [Ref.8] report peak temperatures of 18,000 °K in a transferred arc torch and Chen, et al., [Ref.9] have measured temperatures of 14,000 °K in a plasma jet (non-transferred) using absolute emissivities. The purpose of the present research is to describe a numerical code developed in Code 6720 which can solve for thermodynamic equilibria among a mixture of elemental atoms, diatomic molecules, associated ions, and electrons up to arbitrary temperatures. Of course at temperatures where all of the molecules have been dissociated

(~20,000 ° K) and all that remains are atoms, ions, and electrons, then the standard Saha equilibrium formula is valid. Below temperatures of complete dissociation it is also possible to include molecules within a Saha-like framework and solve a set of Saha equations which involve the partition function for each species considered [Ref.10]. In this approach to obtaining equilibrium, the proper set of abundances $\{N_i\}$ is the one which maximizes the system partition function.

In Section II the theoretical approach used in the equilibrium solver is developed with an example consisting of two elements: carbon and oxygen. The numerical code can treat an arbitrary number of elements, but again we note that it is limited to only diatomic molecules and their associated ions. Section III is a user's guide to MLTE, the numerical code, and presents results for four examples. MLTE is a subroutine and a listing of a sample driver, written in FORTRAN, is included in Section IV. One major component of the code is the calculation of the partition function of each species at an arbitrary temperature from a data base. The entries for the molecular and atomic data bases are described in Section V. Basically the molecular data for the energy levels in diatomic molecules is taken from Herzberg's classic book [Ref.11] or Huber and Herzberg [Ref.12]. Based on the data tables it is just as easy to calculate a specific heat as a partition function. If the specific heat for an individual species can be fit above 6000K by a quartic in temperature and the heat of formation is known, then the coefficients can be formulated in a manner similar to that in the NASA tables [Ref.7]. It is then possible to use standard solvers such as CHEMKIN to properly calculate the equilibrium abundances at temperatures above the existing table limits of 6000K. This approach is described in Section VI with several other examples.

II. MLTE

A Molecular (and Atomic) LTE Populations Code

The purpose of the MLTE code is to calculate the LTE population distribution over the set of allowed states, both molecular (diatomic only) and atomic. The sets of electronic states are defined in the input data files. The code calculates their LTE distribution at temperature T_e . For the molecules the distribution over the vibrational and rotational states in LTE at temperatures T_v and T_r , respectively, is also calculated.

The first, and by far the most complicated part, is the distribution over the electronic states. The MLTE subroutine is best described in terms of a specific example. Consider the following system: G_o oxygen atoms/cc and G_c carbon atoms/cc at temperature $T_e = T$ (°K). The molecular and atomic species that we will include in the calculation are:

1	O_2
2	O_2^+
3	CO
4	CO^+
5	C_2
6	O
7	O^+
8	C
9	C^+
10	e

Note 1: In this example and throughout, molecules (in arbitrary order) are first, then atoms and electrons are last.

Note 2: The set does not include C_2^+ . If there are any stable C_2^+ states, they are not listed in Herzberg and will be ignored here. Also CO_2 is not included since the code is limited to diatomic molecules.

Next let us define a descriptive array such that $S_{i,k}$ is the number of atoms of type k in the items in the above list. It is convenient to extend the k types to include electrons. The order in each group is not important, however the code is hard wired to have the molecular species first followed by the atomic species and then the electrons.

		$S_{l,k}$		
l		k=1	k=2	k=3
		C	O	e
1	O_2	0	2	0
2	O_2^+	0	2	-1
3	CO	1	1	0
4	CO^+	1	1	-1
5	C_2	2	0	0
6	O	0	1	0
7	O^+	0	1	-1
8	C	1	0	0
9	C^+	1	0	-1
10	e	0	0	1

We will follow the procedure (and to some extent the notation) of Chapter 5 of “Atomic Theory of Gas Dynamics” by Bond, Watson and Welch [Ref. 10]. The partition function (their equation 5-121) can be written as

$$Z = \sum \prod_l Z_l(N_l) \quad (1)$$

For the product, l runs over all the species included in the list, and the sum is over all the sets of N_l consistent with the following constraints:

$$\sum_{l=1}^{10} N_l S_{l,1} = G_C \quad \text{Conservation of carbon atoms} \quad (2)$$

$$\sum_{l=1}^{10} N_l S_{l,2} = G_O \quad \text{Conservation of oxygen atoms} \quad (3)$$

$$\sum_{l=1}^{10} N_l S_{l,3} = 0 \quad \text{Conservation of charge.} \quad (4)$$

The partition function is maximized when the set N_i has the “most probable” value, $\bar{N}_i$. That is

$$Z_{\max} = \prod_i Z_i(\bar{N}_i) = \Gamma. \quad (5)$$

The $Z_i(N_i)$ can be written

$$Z_i(N_i) = \frac{1}{N_i!} Q_i^{N_i}. \quad (6)$$

The Q_i are further defined as

$$Q_i = V \left(\frac{4 \pi M k t}{h^2} \right)^{3/2} Q_{i,i} \quad (7)$$

where the $Q_{i,i}$ are the sums over the internal states of the species, and will be discussed in subsection IIa. The energy levels and statistical weights of the internal states are part of the input files. Note that for electrons that sum is 2.

By using Sterling’s approximation we can write

$$\ln(\Gamma) = \sum_{i=1}^{10} \left\{ -N_i [\ln(N_i) - 1] + N_i \cdot \ln(Q_i) \right\}. \quad (8)$$

The change in $\ln(\Gamma)$ due to an arbitrary change in the set N_i is

$$\delta \ln(\Gamma) = \sum_{i=1}^{10} \delta N_i \ln(Q_i / N_i), \quad (9)$$

however we need to subject the variations δN_i to the constraints of eqs 2,3, and 4. Thus

$$\delta \left[\sum_{i=1}^{10} N_i S_{i,1} \right] = 0 \rightarrow \sum_{i=1}^{10} \delta N_i S_{i,1} = 0 \quad (10)$$

$$\delta \left[\sum_{i=1}^{10} N_i S_{i,2} \right] = 0 \rightarrow \sum_{i=1}^{10} \delta N_i S_{i,2} = 0 \quad (11)$$

$$\delta \left[\sum_{i=1}^{10} N_i S_{i,3} \right] = 0 \rightarrow \sum_{i=1}^{10} \delta N_i S_{i,3} = 0. \quad (12)$$

With the constraints in this form we can use the Lagrange multiplier technique to eliminate the constraints. That is

$$\delta \ln(\Gamma) = \sum_{l=1}^{10} \{ \delta N_l \cdot \ln(Q_l / N_l) \} + \sum_{k=1}^3 \lambda_k \left\{ \sum_{l=1}^{10} \delta N_l S_{l,k} \right\} \quad (13)$$

and now since the δN_l are arbitrary we can equate the coefficient of each to zero independently. Thus for each species in the list we can write

$$\ln(Q_l / N_l) + \sum_{k=1}^3 \lambda_k S_{l,k} = 0. \quad (14)$$

It is convenient to define $n_l = N_l / V$ and $q_l = Q_l / V$ then eq. 14 can be written

$$\ln(q_l / n_l) + \sum_{k=1}^3 \lambda_k S_{l,k} = 0. \quad (15)$$

To eliminate the Lagrange multipliers we note that there are three rows of the S matrix that have a single entry equal to 1. In the present example, these rows correspond to $l = 6, 8,$ and 10 i.e. to neutral carbon, oxygen, and the electrons. By inspection then

$$\begin{aligned} \lambda_1 &= -\ln(q_8 / n_8), \\ \lambda_2 &= -\ln(q_6 / n_6), \\ \lambda_3 &= -\ln(q_{10} / n_{10}). \end{aligned} \quad (16)$$

When these are substituted into eq. 15 we obtain

$$\ln(q_l / n_l) - \ln(q_6 / n_6) S_{l,2} - \ln(q_8 / n_8) S_{l,1} - \ln(q_{10} / n_{10}) S_{l,3} = 0 \quad (17)$$

or

$$\ln(q_l / n_l) = \ln \left\{ (q_6 / n_6)^{S_{l,2}} (q_8 / n_8)^{S_{l,1}} (q_{10} / n_{10})^{S_{l,3}} \right\}, \quad (18)$$

$$n_l = q_l (n_6 / q_6)^{S_{l,2}} (n_8 / q_8)^{S_{l,1}} (n_{10} / q_{10})^{S_{l,3}}. \quad (19)$$

Equation 19 can be used to solve for the number density for species 1, 2, ..., 9 in terms of two unknowns n_6 and n_8 . The electron density n_{10} can be calculated from

$$n_{10} = - \sum_{l=1}^9 n_l S_{l,3}, \quad (20)$$

where n_6 and n_8 are still unknown.

It is convenient to define

$$\begin{aligned}
y_1 &= R n_8 / q_8 \\
y_2 &= R n_6 / q_6 \\
y_3 &= R n_{10} / q_{10}
\end{aligned} \tag{21}$$

where R is a scaling parameter to be defined at the time of numerical solutions. In terms of these new variables equation 19 becomes:

$$n_i = q_i \prod_{k=1}^3 (y_k / R)^{S_{i,k}}, \tag{22}$$

where again y_3 is an implicit function of y_1 and y_2 .

To find the functional dependence of y_3 on y_1 and y_2 we write:

$$\begin{aligned}
n_{10} &= - \sum_{i=1}^9 n_i S_{i,3} \\
&= - \sum_{i=1}^9 q_i \prod_{k=1}^3 [y_k / R]^{S_{i,k}} S_{i,3} \\
&= - \sum_{i=1}^9 q_i \prod_{k=1}^2 [y_k / R]^{S_{i,k}} [y_3 / R]^{S_{i,3}} S_{i,3}
\end{aligned}$$

since $S_{i,3}$ is either -1 or 0, $[y_3 / R]^{S_{i,3}} S_{i,3} = \frac{1}{y_3 / R} S_{i,3}$ and we can write

$$\begin{aligned}
n_{10} &= \frac{- \sum_{i=1}^9 q_i \prod_{k=1}^2 [y_k / R]^{S_{i,k}} S_{i,3}}{y_3 / R} \text{ therefore} \\
y_3 = R n_{10} / q_{10} &= [R / q_{10}] \frac{- \sum_{i=1}^9 q_i \prod_{k=1}^2 [y_k / R]^{S_{i,k}} S_{i,3}}{y_3 / R} \text{ or} \\
y_3 &= R \left\{ - \sum_{i=1}^9 q_i \prod_{k=1}^2 [y_k / R]^{S_{i,k}} S_{i,3} / q_{10} \right\}^{1/2}. \tag{23}
\end{aligned}$$

We will consider two cases: constant number densities and constant pressure. For constant number density we can rewrite equations 2 and 3 as

$$\begin{aligned}\sum_{l=1}^9 n_l S_{l,1} - G_c &= 0 \\ \sum_{l=1}^9 n_l S_{l,2} - G_o &= 0.\end{aligned}\tag{24}$$

When equations 22 and 23 are substituted into equations 24 we obtain two equations in two unknowns.

For the case of constant pressure we need to insure that

$$k_B T \sum_{l=1}^{10} n_l - P_0 = 0.\tag{25}$$

One more equation is needed to close the system. The code is set up for the “no diffusion” approximation. In this approximation we can write

$$\sum_{l=1}^9 n_l S_{l,1} - F_{co} \sum_{l=1}^9 n_l S_{l,2} = 0 ,\tag{26}$$

where F_{co} is the constant ratio of oxygen to carbon nuclei.

It is not clear that either equation 25 or 26 is separately consistent with the constraints imposed on the variation procedure used to find the most probable distribution (eq. 2, 3, and 4). One can imagine an iterative scheme in which constant pressure is achieved by first solving for the LTE distribution under constant number density and then adjusting the number densities to both obtain the desired pressure (eq. 25) and number density ratio (eq. 26). In that sense eq. 25 and 26 together may satisfy the constraints of eq. 2, 3, and 4. The constant pressure version of the code was implemented replacing eqs. 24 by 25 and 26 in the numerical scheme. The results were checked against the constant number density version and found to be in remarkable agreement.

Equations 24 or 25 & 26 are nonlinear. On the other hand, the $S_{l,j}$ are integers and in fact are only 0, ∓ 1 , 2. The systems of equations are ideal for a numerical scheme such as the BROYDN subroutine described in Chapter 9 of “Numerical Recipes in Fortran” by Press, Teukolsky, Vetterling, and Flannery [Ref. 13] . For the special case of a single homonuclear molecule the system of equations reduces to a single equation and BRENT’s method is more appropriate than BROYDEN’s.

IIa. Sum over States

The first point to be addressed is the zero or reference point for bound energies. Chemists like to work with the Enthalpy and to have the ground state of certain molecules be the reference point. When this is done one needs to include the heat of formation in a consistent manner. We will adopt the simpler strategy of letting the ground state of atoms be the reference point of all (bound state) energies. With this reference system the ground state of atomic ions has an energy equal to the ionization potential, while the lowest state of a molecule has an energy equal to minus the dissociation (into the ground state of the atoms) potential. All other electronic states of a molecule are referenced to the lowest state.

We look first at the sum over states for the atoms and atomic ions. An isolated atom has an infinite number of states and the sum over them diverges! An atom in a gas or plasma is not isolated and, although we can not solve the $10^{11} \rightarrow 10^{16}$ body problem to prove it, the number of bound states is finite and not too large. For atoms in a plasma the problem has been addressed by Theimer and Kepple [Ref. 14] among others. Although similar arguments can be made for the case of an atom in a gas, we will adopt a much more pragmatic point of view and simply truncate the set of bound states. From experience we know that the results are not sensitive to the exact number of bound states that are retained. Suffice it to say that a state should not be retained if its mean radius is larger than that of the unit sphere. That is

$$\bar{r}_b < \left(\frac{3}{4\pi n_a} \right)^{1/3} \quad (27)$$

where n_a is the density (in cm^{-3}) and $\bar{r}_b$ is the mean radius of the bound state. This cutoff is density dependent and should not be considered an upper bound to the cutoff rather than one to use in a typical calculation.

The sum over states for atoms is

$$q_l = \sum_i g_i \exp(-E_i / kT), \quad (28)$$

where the sum is over the set of bound states for the l th atom, g_i is the statistical weight, and $E_i = 0$ for its ground state. For atomic ions the sum over states is similar

$$q_l = \sum_i g_i \exp(-(I_l + E_i) / kT) = \exp(-I_l / kT) \sum_i g_i \exp(-E_i / kT) \quad (30)$$

where I_l is the ionization potential of the atom of the l th species and again $E_i = 0$ for the ionic ground state.

The sum over states for molecules is similar except that the summation is over electronic, vibrational and rotational states.

$$q_l = \exp(+D_l / kT) \sum_{i,v,r} g_{i,v,r} \exp(-E_{i,v,r} / kT), \quad (31)$$

In the above equation D_l is the dissociation energy from the ground state of molecules of the l th species.

The basic data for molecules is not known to great precision. The potential curve for each electronic state is characterized by only a few parameters, and many of these are only valid near the bottom of the well. The energy, in wave numbers, is given by Herzberg [Ref. 11 equations IV, 9; IV, 10; and IV, 11]:

$$T = T_e + G + F, \quad (32)$$

$$G = \omega_e (v + 1/2) - \omega_e x_e (\omega + 1/2)^2 + \omega_e y_e (v + 1/2)^3 + \dots, \quad (33)$$

$$F = B_v J(J + 1) - D_v J^2(J + 1)^2 + \dots, \quad (34)$$

$$D_e = 4 B_e^3 / \omega_e^2,$$

$$\beta_e = D_e \left(\frac{8 \omega_e x_e}{\omega_e} - \frac{5 \alpha_e}{B_e} - \frac{\alpha_e^2 \omega_e}{24 B_e^3} \right),$$

$$D_v = D_e + \beta_e (v + 1/2),$$

$$B_v = B_e - \alpha_e (v + 1/2).$$

In the above, v is the vibrational quantum number and J is the rotational quantum number. In spite of the fact that the parameters were fitted to the bottom of the potential curve, we retained all terms such that the energy was less than the dissociation energy, G and F greater than zero, $D_v < 0.4 D_e$, and $1 - D_v J(J + 1) > 0.1$. These cutoffs in v and J are arbitrary but necessary. Blind use of equations 32 - 34 can, and often does, lead to energies that are negative with respect to the ground state zero point! There is a large body of literature on corrections and approximations for this molecule or that one. We embarked on this project to see just how far we could get with no more data than that found in [Ref. 11 and 12]. Thus our results will not always agree with those from CHEMKIN.

III. USER'S GUIDE

Subroutine MLTE calculates the LTE distribution over the electronic states. Once the distribution over the electronic states is known a call to ALTEDIST or MLTEDIST will calculate the distribution over the excited atomic states or the vibrational and rotational molecular states.

The input arguments passed to subroutine MLTE are the electron, vibrational and rotational temperatures all in degrees Kelvin. The LTE (electronic states) population distribution is returned in the array den. The populations are ordered as in the S array (specified in the input file "mlte.in").

The atomic and molecular data are read by MLTE on the first call only. In addition to the formal arguments in the MLTE call, information is passed to and from MLTE through the following common blocks: qp, cntrl, names, and logic. The use of the variables in these common blocks is illustrated in the sample MLTE driver. To make it easier to find these variables they are printed in bold face. Many of these variables are set via the namelist "mltein". Default values for most of the variables in the namelist are set in the sample driver.

Some important variables:

q	see eq. 15 and next 3 lines.
en	population...see eq. 22.
r	numerical scaling factor...see eq. 21.
p0	(constant) pressure.
tk	electron temperature in degree Kelvin.
fn	see F_{co} in eq. 26.
amass	mass of the atom or molecule.
fg	used to calculate a "first guess" of the atomic number densities.
dn1	total (nuclear) density of first atomic species.
xlow/xhigh	For the case of one homonuclear molecular species, xlow/xhigh are the min/max number density for the neutral atom .
ztol	Tolerance in the solution of the equation for the neutral atomic number density in the above case.
eps	Convergence tolerance for the case of more than one unknown.
imol	Number of molecular species. Specified in "mlte.in".
iatom	Number of atomic species. "
id	imol + iatom
idp	id + 1
maxloop	maximum number of iterations.

index	For internal use only!
is	The “S” array. Specified in “mlte.in”.
ns	Number of independent variables.
nsp	ns + 1
jvv	Number of vibrational levels included in the calculations.
jvr	Number of rotational levels “.
cp	Constant pressure?
latom	Calculate the distribution over the excited atomic states?
lmol	“ vibrational and rotational molecular states?
lbug	Print debug information?
lplot	Non-standard Plots?
lnp	Use xlow/xhigh?

The arguments to ALTEDIST are the electronic state distribution (output from MLTE) and the electron temperature. The arguments to MLTEDIST are the electronic state distribution, the electron, vibrational, and rotational temperatures. At present the outputs are files “mlte.atom”, and “mlte.mol”. The calls to ALTEDIST and MLTEDIST can be implemented by setting latom and lmol = .true. before the call to MLTE.

If the code returns a “Singular Jacobian” error, you need a better first guess. If the code returns an error for too many iterations in Broyden, you can continue, but if the error messages persist, you may need a better first guess for the number density. i.e. `den(i)`. In the sample driver, the input parameter `dn1` sets the overall density and can be set in the namelist.

The gradients in the partition function are so steep that unless you have a fair “first guess” the code can go into a “chaotic mode”. The Broyden package is quite sophisticated and we have not been able to modify it to avoid the fall into chaos. From eq. 21 and 22 we note that the y ’s determine all the n ’s but only the n ’s corresponding to the neutral atomic species determine the y ’s. Thus it is clear that it is best to start at some intermediate temperature where our intuition tells us that most of the populations will be neutral atoms and thus we can make a very good first guess of the y ’s. Once we have obtained a converged solution at any temperature, we can then “adiabatically” (the converged solution at the last temperature is a good first guess for the next) change the temperature up or down as needed. The same technique may be necessary for very small p_0 , small or large f_n (ratio of atomic species).

IIIa. EXAMPLES

The following four examples all use the sample driver listed on pages 29 - 33. For these examples the desired output are plots of the populations Vs temperature. The files *.com are UNIX scripts. Their purpose is to show explicitly the input and data files.

INPUT DATA

case1.com

cp /home/mydisk/lte/case1.in	mlte.in
cp /home/mydisk/lte/case1.nlst	mlte.nlst
cp /home/mydisk/data/molq/molqo2.in	mole1
cp /home/mydisk/data/molq/molqo2p.in	mole2
cp /home/mydisk/data/molq/molqco.in	mole3
cp /home/mydisk/data/molq/molqcop.in	mole4
cp /home/mydisk/data/molq/molqc2.in	mole5
cp /home/mydisk/data/molq/atqo.in	atom1
cp /home/mydisk/data/molq/atqop.in	atom2
cp /home/mydisk/data/molq/atqc.in	atom3
cp /home/mydisk/data/molq/atqcp.in	atom4
/home/mydisk/lte/mlte.x	

case1.nlst

```
$mltein
cp=.true.,
p0=1.3332d3,
fn(1)=1.0,
fn(2)=1.0,
itag=' C,O 1.0',
fg=0.001,
dn1=1.0e12,
r=1.0d16,
scale(1)=1.0,
scale(2)=100.0,
scale(3)=1.0,
eps=0.001,
maxloop=80,
npp=4,
t0=0.690,
delte = -0.000431,
ntp=850,
latom=.false.,
lmol=.false.,
lbug1=.false.,
lbug2=.false.,
lbug3=.false.,
```

```

ltrace=.false.,
jvv=65,
jvr=300,
tvv=-1.0,
trr=-1.0,
dshp(1)='$$$$$$$$$$$$$$$$$',
dshp(2)='$$$$$$$$$$$$$$$$$',
dshp(3)='$$$$"$"$"$"$"$"$'$,
dshp(4)='$$"$"$"$"$"$"$'$,
dshp(5)='$$"$"$"$"$"$"$'$,
dshp(6)='$$$$$$$$$$$$"$"$"$'$,
dshp(7)='$$"$"$"$"$"$"$'$,
dshp(8)='$$"$"$"$"$"$"$'$,
dshp(9)='$$"$"$"$"$"$"$'$,
nlb(1)=28,
nlb(2)=28,
nlb(3)=28,
nlb(4)=28,
nlb(5)=28,
nlb(6)=28,
nlb(7)=28,
nlb(8)=28,
nlb(9)=28,
$end

```

```

cp.....constant pressure?
p0.....if cp is true, then the pressure = p0
fn.....ratio of atoms (nuclei) of type n to type 1
fg.....used for first guess
r.....scale factor
scale.....more scale factors
eps.....convergence test
maxloop...max iterations for convergence
laton.....print out atomic population dist ?
lmol.....print out molecular population dist ?
lbug.....print out debug info ?
jvv.....number of vibrational levels
jvr.....number of rotational levels
tvv.....vibrational temp ( °K) if negative tvv = tk
trr.....rotational temp ( °K) if negative trr = tk

```

case1.in

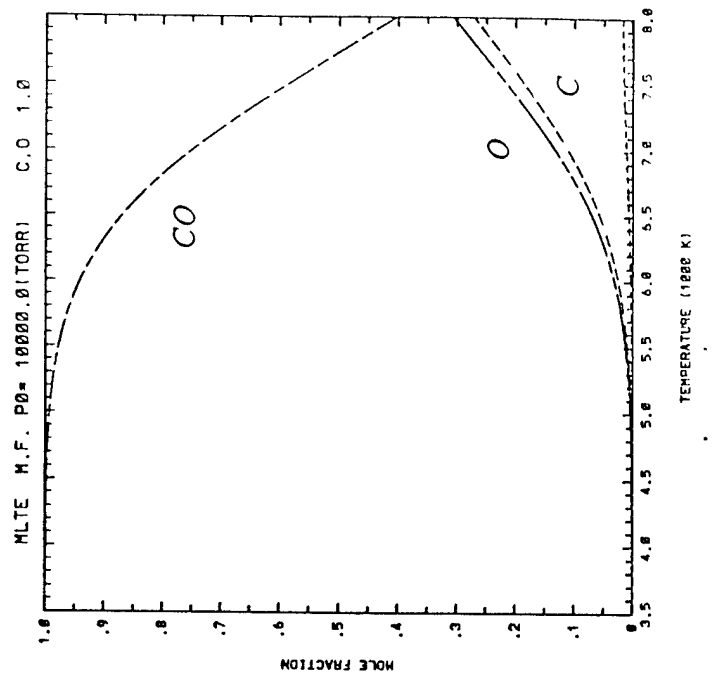
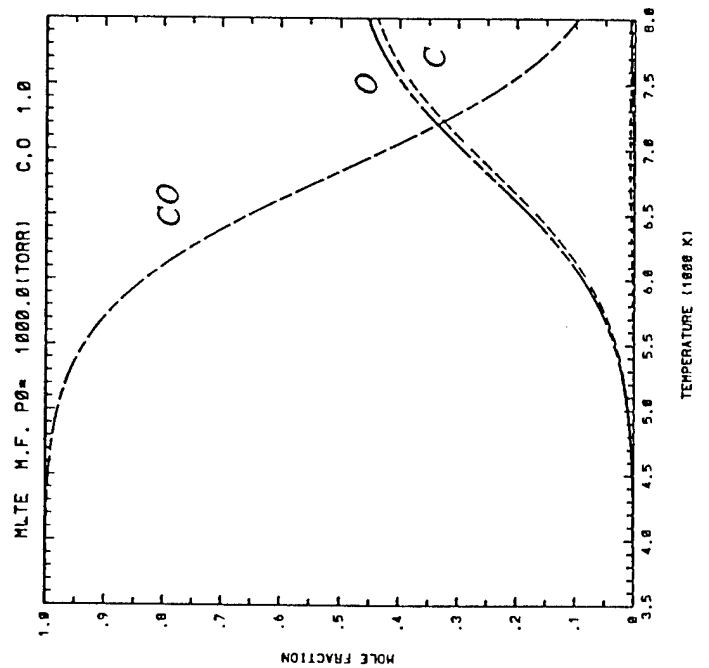
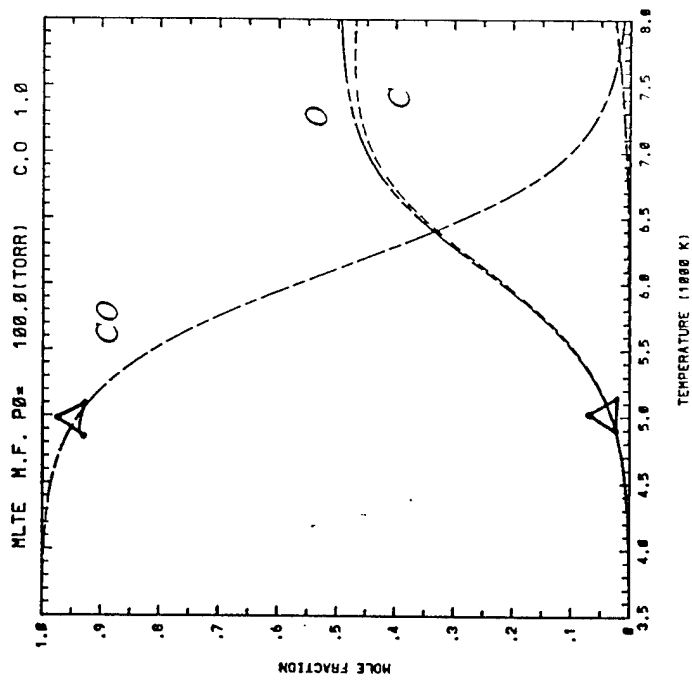
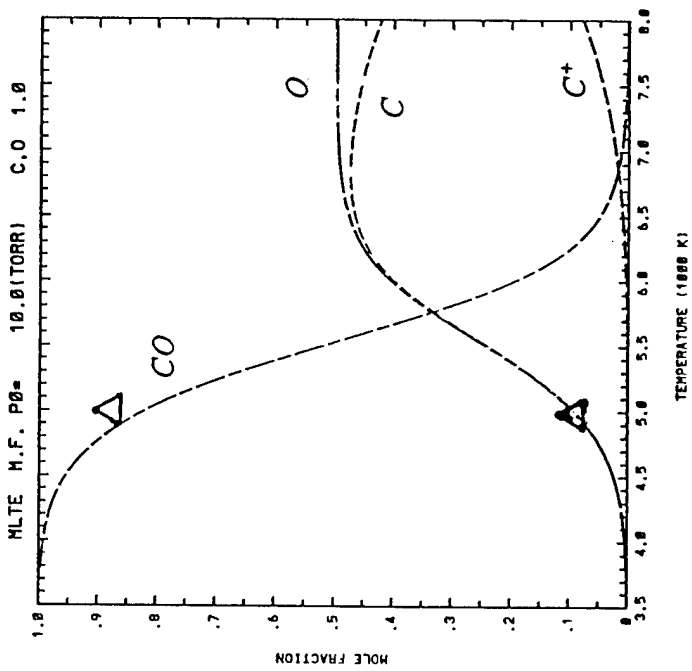
```

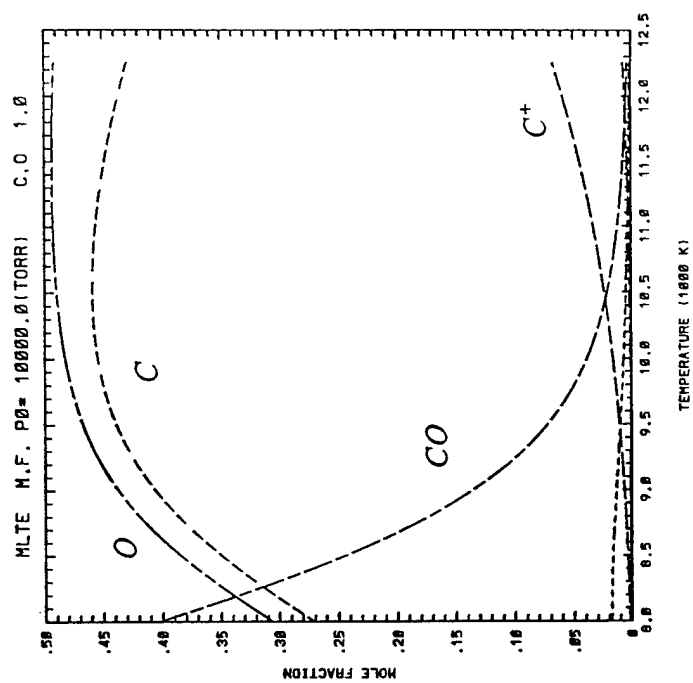
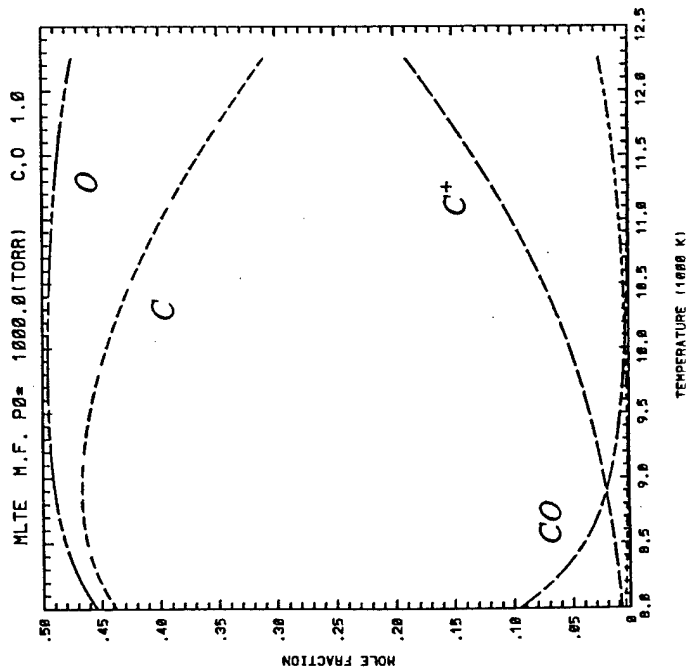
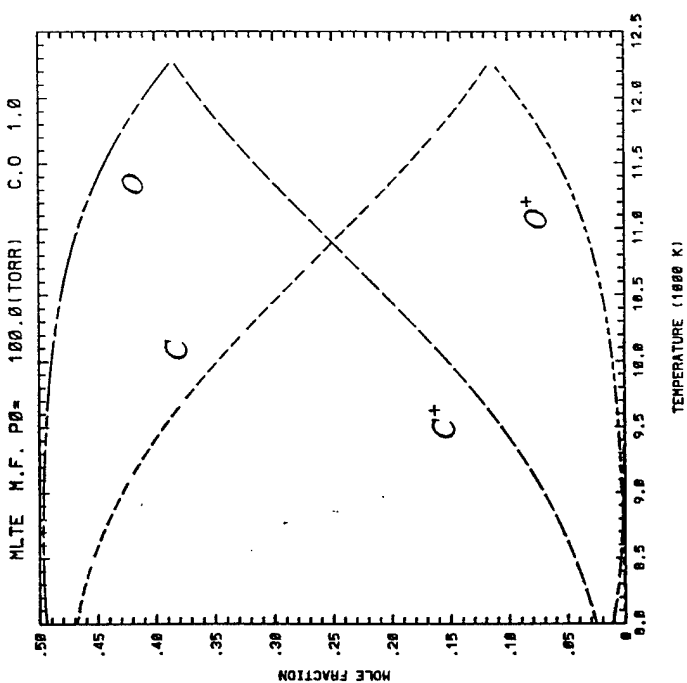
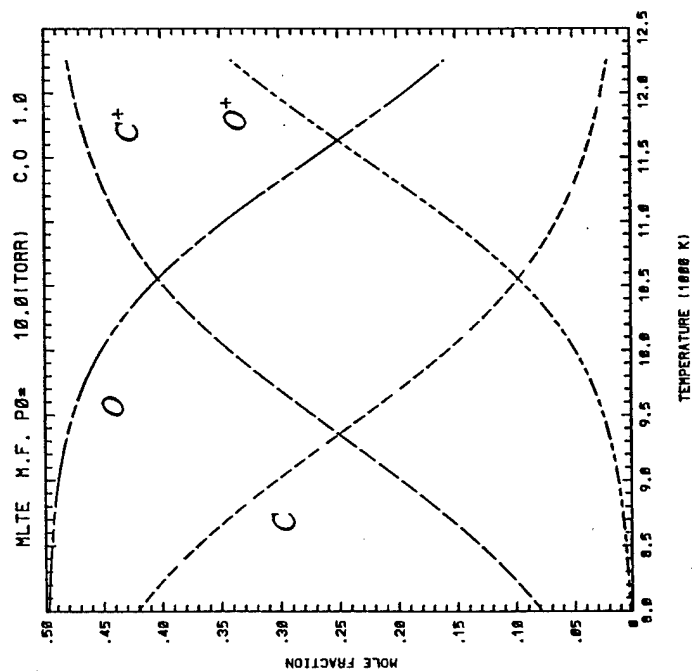
5      4
      S
      k=1 2 3
      C 0 e
1 O2      0 2 0
2 O2+     0 2 -1
3 CO       1 1 0
4 CO+      1 1 -1
5 C2       2 0 0
6 O        0 1 0
7 O+       0 1 -1
8 C        1 0 0
9 C+       1 0 -1
10 e       0 0 1

```

In this example it was necessary to start at an intermediate temperature (8000 °K) and step down to lower temperatures and then make a second run to work up to higher temperatures. The namelist for the second run was the same as for the first except for the sign of delte.

In the following figures the Δs are from CHEMKIN and indicate the agreement between MLTE and CHEMKIN. Note: The CHEMKIN data base is only valid for temperatures below 6000 °K.





case2.com

```
cp /home/mydisk/lte/case2.in      mlte.in
cp /home/mydisk/lte/case2.nlst    mlte.nlst
cp /home/mydisk/data/molq/molqo2.in  mole1
cp /home/mydisk/data/molq/molqo2p.in mole2
cp /home/mydisk/data/molq/molqno.in  mole3
cp /home/mydisk/data/molq/molqnop.in mole4
cp /home/mydisk/data/molq/molqn2.in  mole5
cp /home/mydisk/data/molq/molqn2p.in mole6
cp /home/mydisk/data/molq/atqo.in    atom1
cp /home/mydisk/data/molq/atqop.in   atom2
cp /home/mydisk/data/molq/atqn.in    atom3
cp /home/mydisk/data/molq/atqnp.in   atom4
/home/mydisk/lte/mlte.x
```

case2.nlst

```
$mltein
cp=.true.,
p0=1.3332d4,
fn(1)=1.0,
fn(2)=1.00,
itag=' N,O 1.00',
fg=0.001,
dn1=1.0e12,
r=1.0d16,
scale(1)=1.0,
scale(2)=100.0,
scale(3)=1.0,
eps=0.001,
maxloop=80,
npp=4,
t0=0.069,
delte = -0.000650,
ntp=850,
latom=.false.,
lmol=.false.,
lbug1=.false.,
lbug2=.false.,
lbug3=.false.,
ltrace=.false.,
lplot1=.false.,
lplot2=.false.,
x2low=3.80d9,
dlx2=1.05,
tpl=+8.0,
dtp=0.04,
spl=5.0,
dsp=0.02,
```

```

jvv=65,
jvr=300,
tvv=-1.0,
trr=-1.0,
dshp(1)='$$$$$$$$$$$$$$$$$$',
dshp(2)='$$$$$$$$$$$$$$$$$$',
dshp(3)='$$$"$$$"$$$"$$$"$',
dshp(4)='$$$"$$$"$$$"$$$"$',
dshp(5)='$$$"$$$"$$$"$$$"$',
dshp(6)='$$$$$$$$$$$$$"$$$"$',
dshp(7)='$$$"$$$"$$$"$$$"$',
dshp(8)='$$$"$$$"$$$"$$$"$',
dshp(9)='$$$"$$$"$$$"$$$"$',
nlb(1)=28,
nlb(2)=28,
nlb(3)=28,
nlb(4)=28,
nlb(5)=28,
nlb(6)=28,
nlb(7)=28,
nlb(8)=28,
nlb(9)=28,
$end

```

```

cp.....constant pressure?
p0.....if cp is true, then the pressure = p0
fn.....ratio of atoms (nuclei) of type n to type 1
fg.....used for first guess
r.....scale factor
scale.....more scale factors
eps.....convergence test
maxloop...max iterations for convergence
laton.....print out atomic population dist ?
lmol.....print out molecular population dist ?
lbug.....print out debug info ?
jvv.....number of vibrational levels
jvr.....number of rotational levels
tvv.....vibrational temp ( °K) if negative tvv = tk
trr.....rotational temp ( °K) if negative trr = tk

```

case2.in

6	4			
			S	
		k=1	2	3
		N	O	e
1	O2	0	2	0
2	O2+	0	2	-1
3	NO	1	1	0
4	NO+	1	1	-1
5	N2	2	0	0
6	N2+	2	0	-1
7	O	0	1	0
8	O+	0	1	-1
9	N	1	0	0
10	N+	1	0	-1
11	e	0	0	1

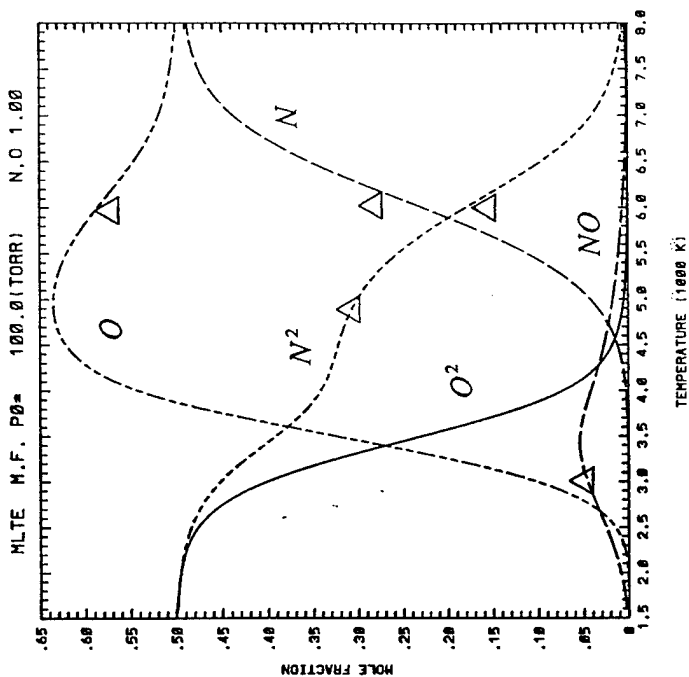


Fig. 9

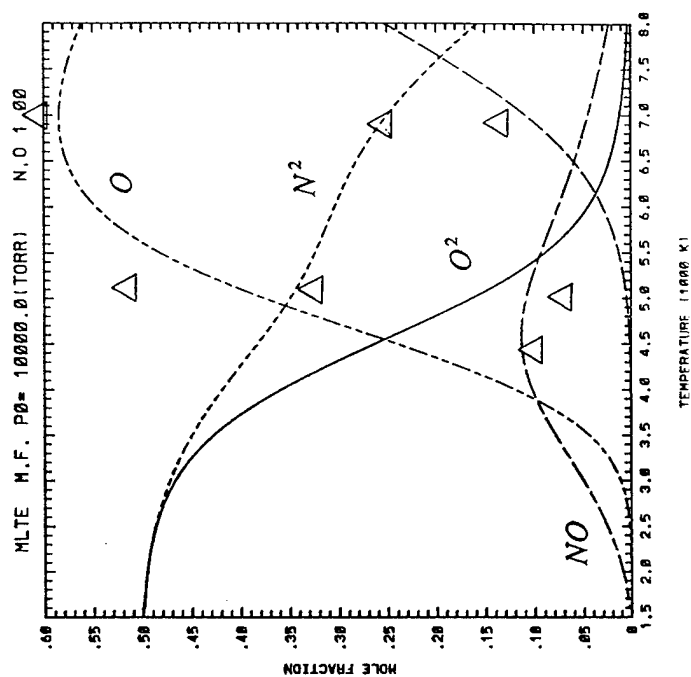


Fig. 10

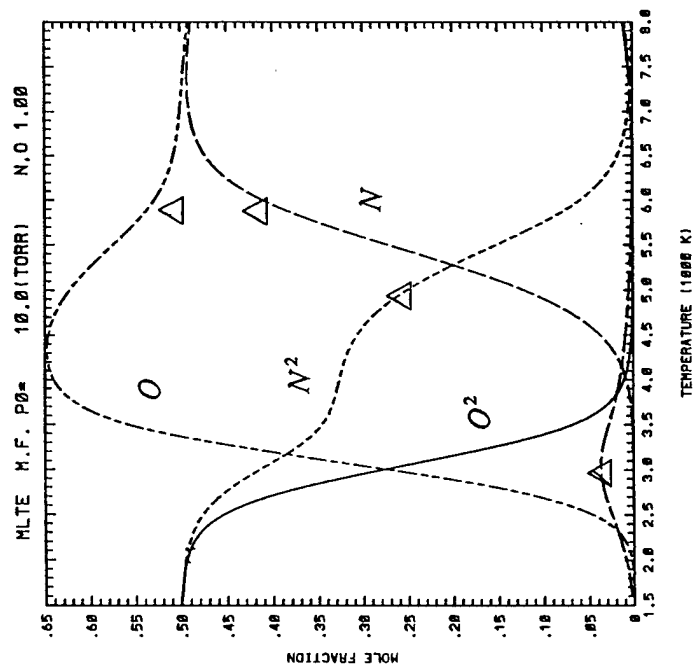


Fig. 11

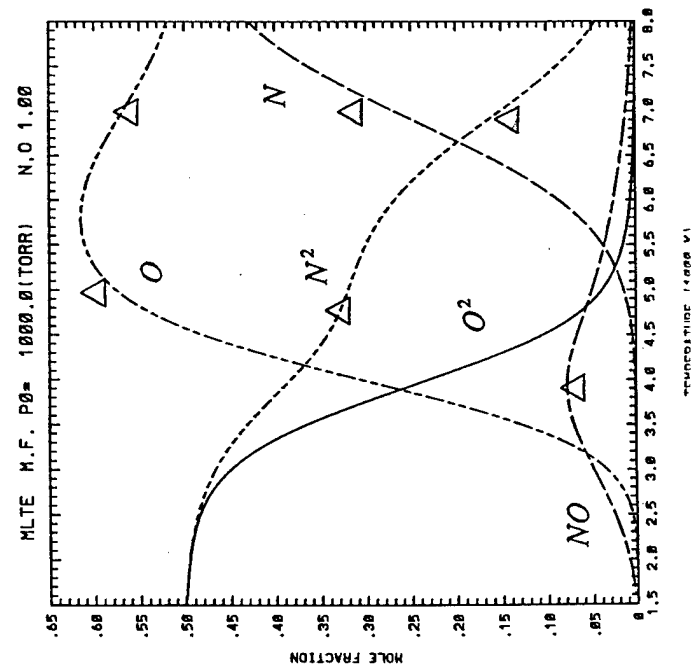


Fig. 12

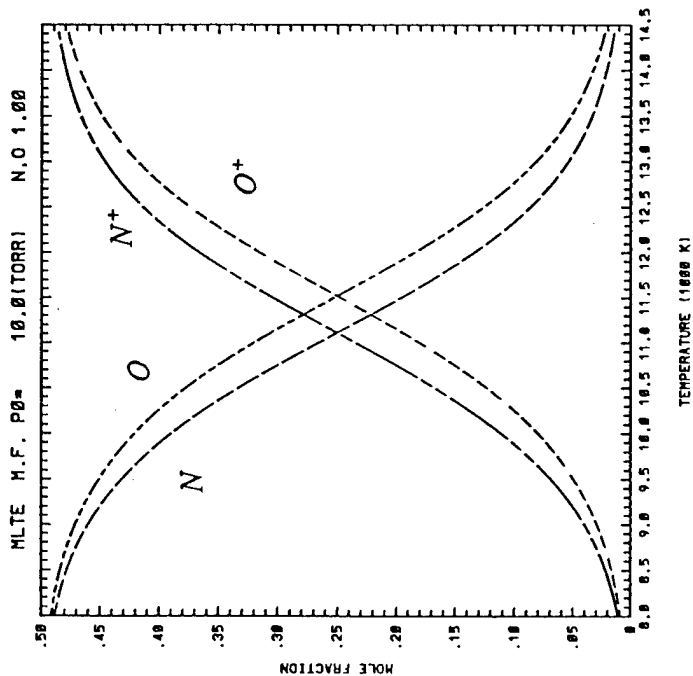


Fig. 13 ←

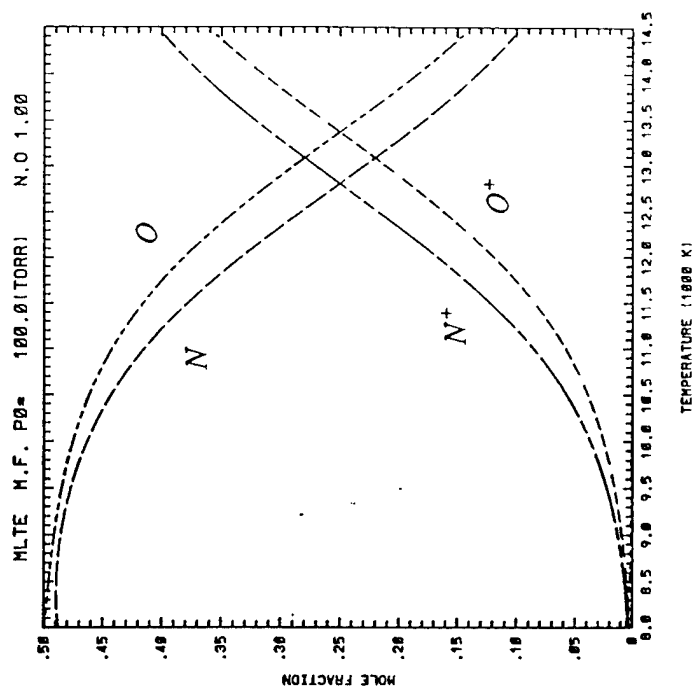


Fig. 14 →

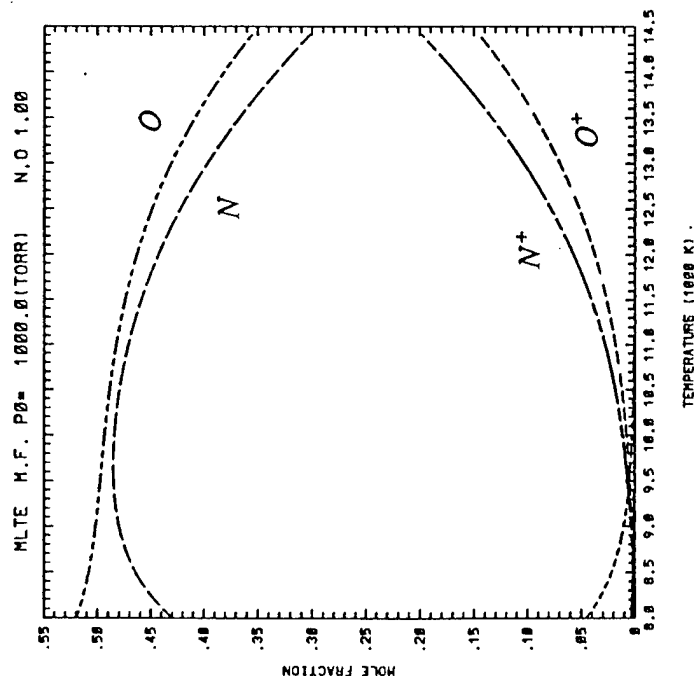


Fig. 15 ←

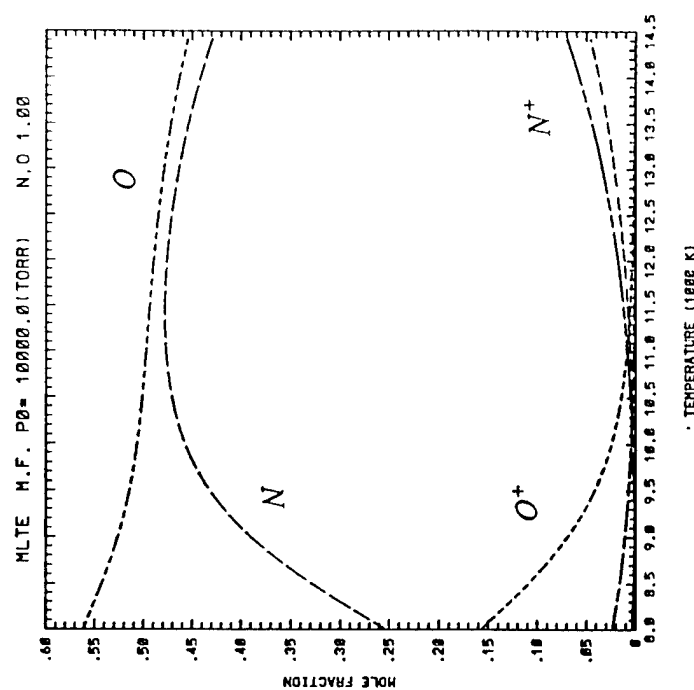


Fig. 16 →

case3.com

```
cp /home/mydisk/lte/case3.in      mlte.in
cp /home/mydisk/lte/case3.nlst    mlte.nlst
cp /home/mydisk/data/molq/molqn2.in mole1
cp /home/mydisk/data/molq/molqn2p.in mole2
cp /home/mydisk/data/molq/atqn.in atom1
cp /home/mydisk/data/molq/atqnp.in atom2
/home/mydisk/lte/mlte.x
```

case3.nlst

```
$mltein
cp=.true.,
p0=1.0000e3,
fn(1)=1.0,
fn(2)=1.00,
itag=' N2, N ',
fg=0.00001,
dn1=1.0e16,
r=1.0d16,
scale(1)=1.0,
scale(2)=100.0,
scale(3)=1.0,
eps=0.001,
maxloop=80,
npp=4,
t0=1.207,
delte = -0.000202,
ntp=501,
xlow=1.0e-36,
xhigh=1.0e+16,
latom=.false.,
lmol=.false.,
lbug1=.true.,
lbug2=.false.,
lbug3=.false.,
ltrace=.true.,
lplot1=.false.,
lplot2=.false.,
tpl=-9.0,
dtp=0.02,
jvv=35,
jvr=150,
tvv=-1.0,
trr=-1.0,
dshp(1)='$$$$$$$$$$$$$$$$$$$$$',
dshp(2)='$$$$$$$$$$$$$$$$$$$$$'
```

```

dshp(3)='$$$$'$$$$$$$$'$',
dshp(4)='$$$'$$$$$'$',
dshp(5)='$$$'$$$$$'$',
dshp(6)='$$$$$$$$$$$$'$',
dshp(7)='$$$'$$$$$'$',
dshp(8)='$$$'$$$$$'$',
dshp(9)='$$$'$$$$$'$',
nlb(1)=28,
nlb(2)=28,
nlb(3)=28,
nlb(4)=28,
nlb(5)=28,
nlb(6)=28,
nlb(7)=28,
nlb(8)=28,
nlb(9)=28,
$end

```

cp.....constant pressure?
 p0.....if cp is true, then the pressure = p0
 fn.....ratio of atoms (nuclei) of type n to type 1
 fg.....used for first guess
 r.....scale factor
 scale.....more scale factors
 eps.....convergence test
 maxloop...max iterations for convergence
 laton.....print out atomic population dist ?
 lmol.....print out molecular population dist ?
 lbug.....print out debug info ?
 jvv.....number of vibrational levels
 jvr.....number of rotational levels
 tvv.....vibrational temp (°K) if negative tvv = tk
 trr.....rotational temp (°K) if negative trr = tk

case3.in

```

2      2
          S
          k=1 2
          N   e
5 N2      2   0
6 N2+     2  -1
9 N        1   0
10 N+      1  -1
11 e        0   1

```

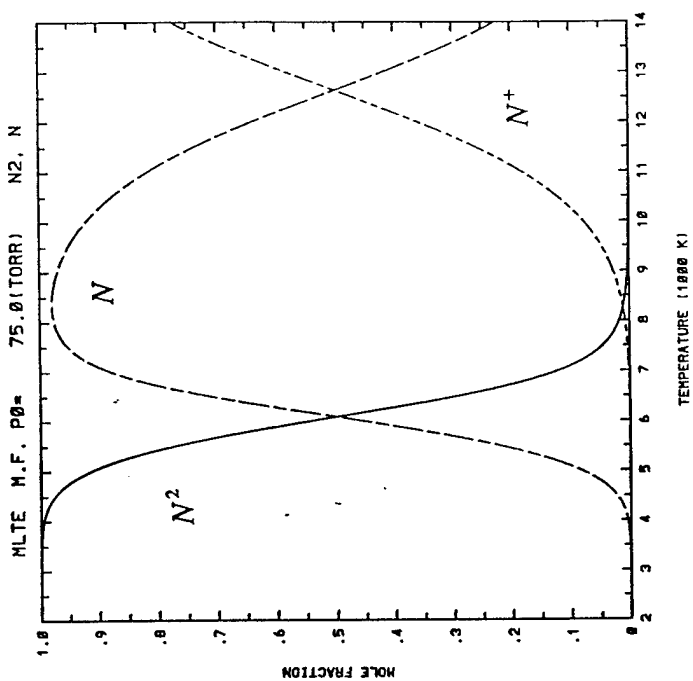


Fig. 18

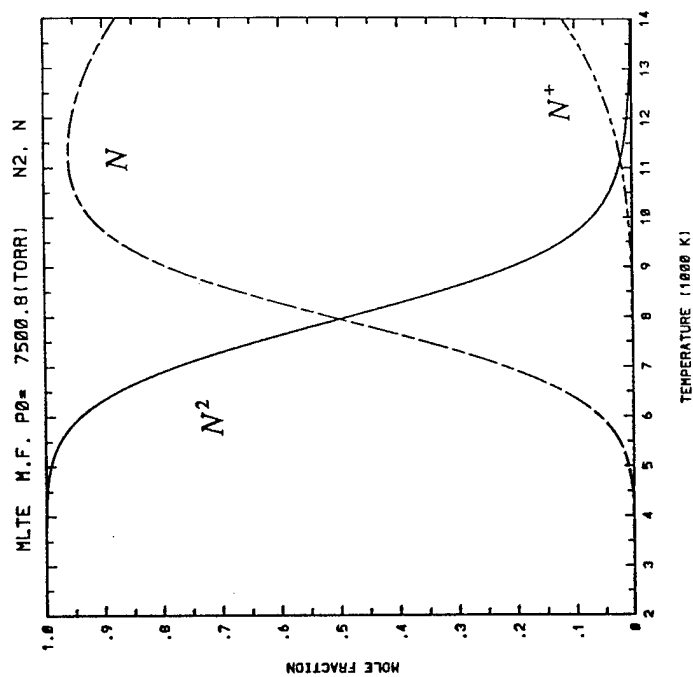


Fig. 20

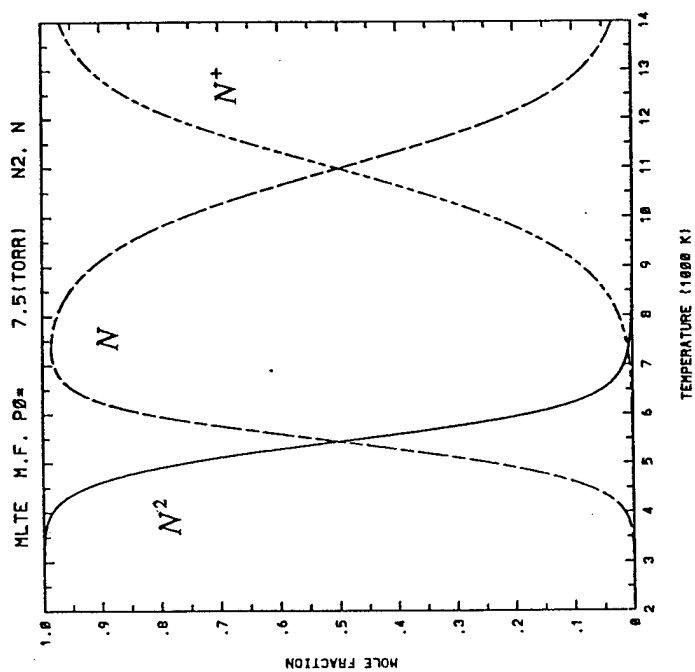


Fig. 17

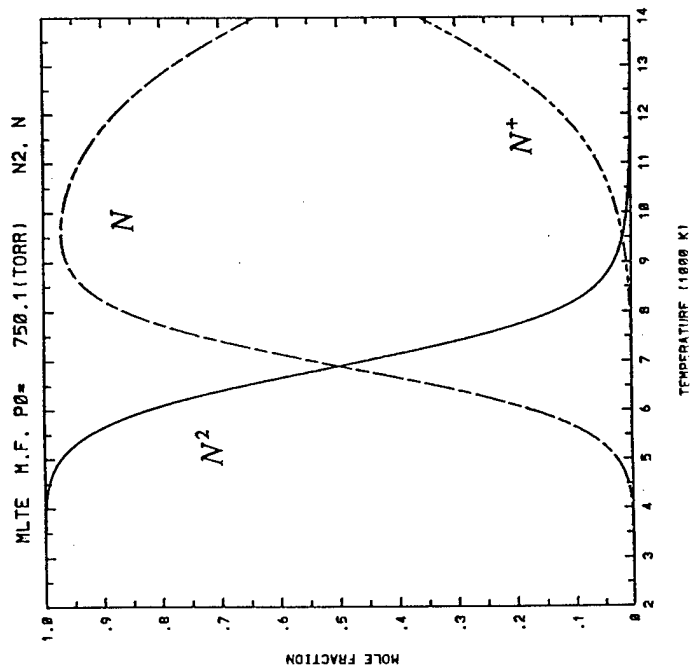


Fig. 19

case4.com

```
cp /home/mydisk/lte/case4.in      mlte.in
cp /home/mydisk/lte/case4.nlst    mlte.nlst
cp /home/mydisk/data/molq/molqn2.in mole1
cp /home/mydisk/data/molq/molqn2p.in mole2
cp /home/mydisk/data/molq/atqn.in atom1
cp /home/mydisk/data/molq/atqnp.in atom2
cp /home/mydisk/data/molq/atqar.in atom3
cp /home/mydisk/data/molq/atqarp.in atom4
/home/mydisk/lte/mlte.x
```

case4.nlst

```
$mltein
cp=.true.,
p0=1.3333e3,
fn(1)=1.0,
fn(2)=3.0,
itag=' N2 N Ar ',
fg=0.00001,
dn1=1.0e2,
r=1.0d16,
scale(1)=1.0,
scale(2)=1000.0,
scale(3)=1.0,
eps=0.001,
maxloop=80,
npp=4,
t0=1.2000,
delte = -0.0030,
ntp=350,
xlow=1.0e-16,
xhigh=1.0e+16,
latom=.false.,
lmol=.false.,
lbug1=.true.,
lbug2=.false.,
lbug3=.false.,
ltrace=.true.,
lplot1=.false.,
lplot2=.false.,
tpl=-9.0,
dtp=0.02,
jvv=35,
jvr=150,
tvv=-1.0,
trr=-1.0,
```

```

dshp(1)='$$$$$$$$$$$$$$$$$',
dshp(2)='$$$$$$$$$$$$$$$$$',
dshp(3)='$$$$"$$$$$"$$',
dshp(4)='$$"$$$$$"$$',
dshp(5)='"$"$"$"$"$$',
dshp(6)='$$$$$$$$$"$"$$',
dshp(7)='"$"$"$"$"$$',
dshp(8)='"$"$"$"$"$$',
dshp(9)='$$"$$$$$"$"$$',
nlb(1)=28,
nlb(2)=28,
nlb(3)=28,
nlb(4)=28,
nlb(5)=28,
nlb(6)=28,
nlb(7)=28,
nlb(8)=28,
nlb(9)=28,
$end

```

```

cp.....constant pressure?
p0.....if cp is true, then the pressure = p0
fn.....ratio of atoms (nuclei) of type n to type 1
fg.....used for first guess
r.....scale factor
scale.....more scale factors
eps.....convergence test
maxloop...max iterations for convergence
laton.....print out atomic population dist ?
lmol.....print out molecular population dist ?
lbug.....print out debug info ?
jvv.....number of vibrational levels
jvr.....number of rotational levels
tvv.....vibrational temp ( °K) if negative tvv = tk
trr.....rotational temp ( °K) if negative trr = tk

```

case4.in

```

2      4
      S
      k=1 2 3
      N Ar e
1 N2      2 0 0
2 N2+     2 0 -1
3 N       1 0 0
4 N+      1 0 -1
5 Ar      0 1 0
6 Ar+     0 1 -1
7 e       0 0 1

```

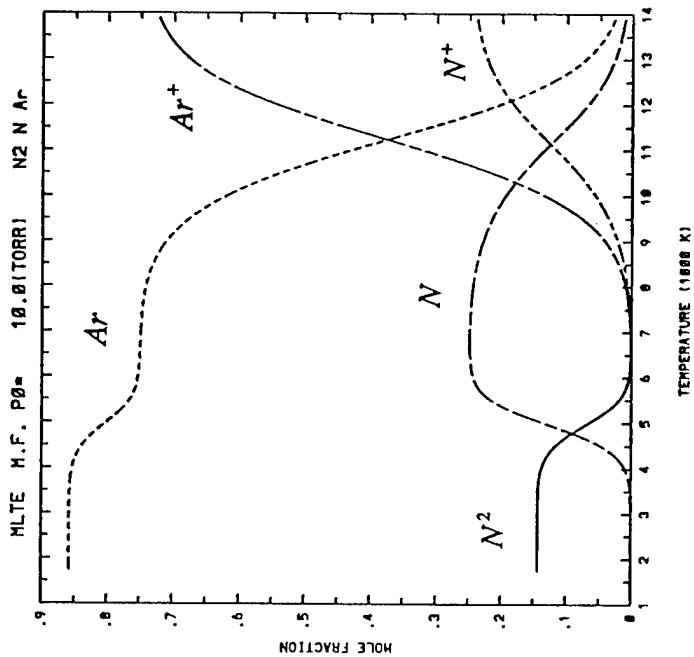


Fig. 21 ←

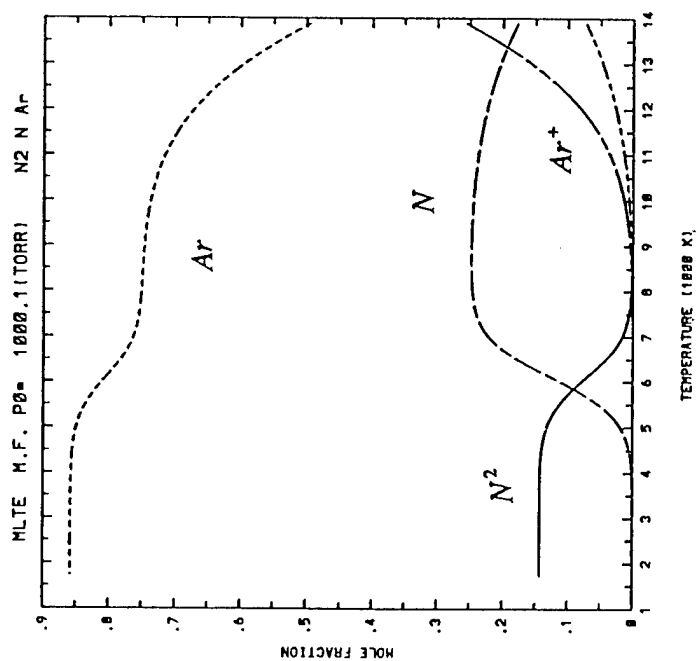


Fig. 23 ←

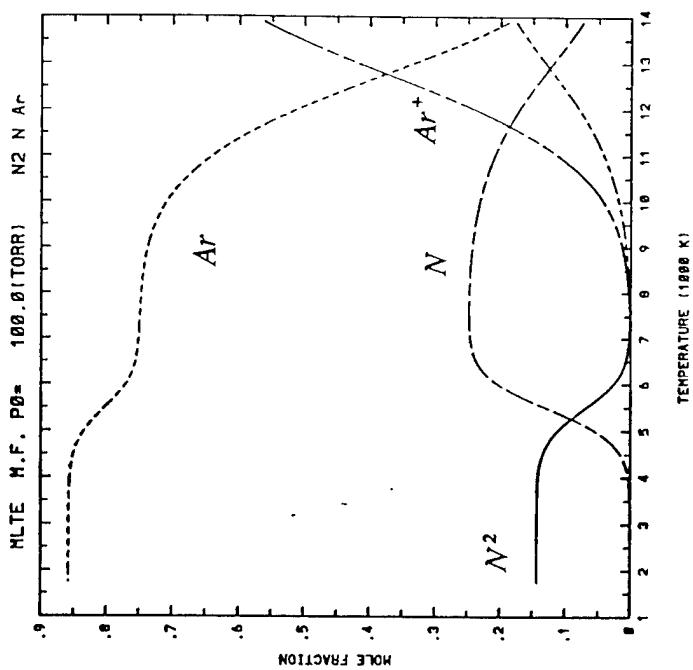


Fig. 22 →

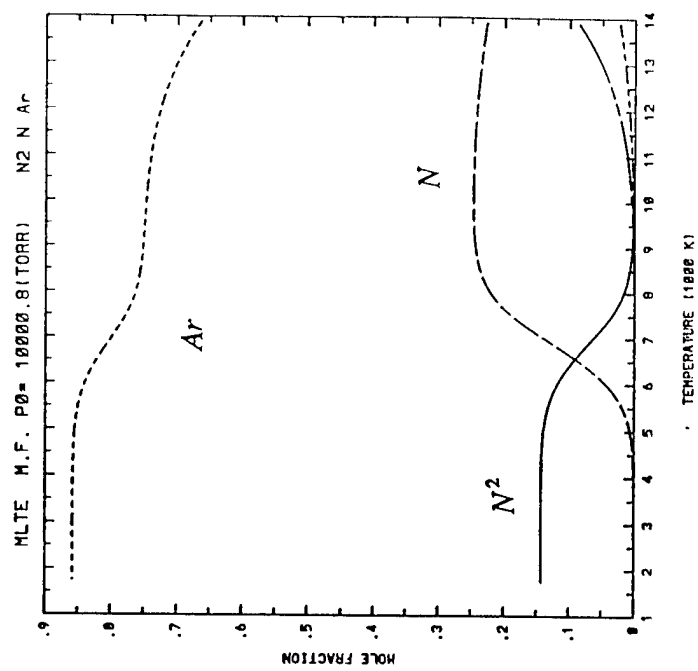


Fig. 24 →

IV. SAMPLE DRIVER

```

cdeck tmlte    <-----<<<
      PROGRAM tmlte
c
c  driver for subroutine mlte...molecular lte distribution
c
c      implicit real*8 (a-h, o-z)
c
c  from MLTE5.MTR  September 2, 1997
c
c      integer va, vd, vm, vam, vamp, ve, vap, vdv2, vv, vr
c      PARAMETER(va=10, vap=va+1, vd=20, vm=10, vam=va+vm,
c      . vamp=vam+1, ve=30, vdv2=vd*vm*vm, vv = 50, vr=100)
c      logical cp, latom, lmol, lbug1, lbug2, lbug3, lplot1, lplot2,
c      . lnp
c      character*6 sname
c      dimension index(va), is(vam, vap), sname(vamp)
c      common /cntrl/ eps, imol, iatom, id, idp, maxloop, index,
c      . is, ns, nsp, jvv, jvr
c      common /names/ sname
c      save /cntrl/, /names/
c      common /logic/ cp, latom, lmol, lbug1, lbug2, lbug3, lplot1,
c      . lplot2, lnp
c      save /logic/
c      dimension q(vamp), en(vamp), scale(va), fn(va), amass(vamp)
c      common /qp/ q, en, r, p0, tk, scale, fn, amass, fg, dn1,
c      . xlow, xhigh, ztol, tpl, dtp, spl, dsp, x2low, dlx2
c      save /qp/
c
c      common /bugs/ ltrace
c
c      logical ltrace
c
c
c      parameter (NPPTS=4, NTPTS=1000, NSM=20)
c
c      character*51 titl
c      character*9 itag
c      character*28 dshp(NSM)
c      character*16 agdshn
c
c      real*4 temp, dens

```

```

dimension temp(NTPTS), dens(NTPTS, NSM), nlb(NSM)
data titl/MLTE M.F. P0=      (TORR)      $'/
dimension den(vamp), sden(vamp, 500), stk(500)

c
namelist /mltein/ cp, p0, dn1, fn, fg, r, scale, eps, latom,
. lmol, lbug1, lbug2, lbug3, jvv, jvr, dn1, tvv, trr,
. maxloop, ntp, npp, xlow, xhigh, ztol, lplot1, lplot2, tpl, dtp,
. t0, delte, p0, itag, dshp, nlb, ltrace, x2low, dlx2, spl, dsp

c
c
c
call opngks

c
c default values for namelist variables
c
cp = .true.
latom = .false.
lmol = .false.
lbug1 = .false.
lbug2 = .false.
lbug3 = .false.
xlow = 1.0e-16
xhigh = 1.0e+16
ztol = 1.0e-06
ltrace = .false.
lplot1 = .false.
lplot2 = .false.
r = 1.0e16
eps = 1.0e-4
fg = 0.90
tvv = -1.0
trr = -1.0
jvv = 20
jvr = 50
ntp = NTPTS
npp = NPTPS
t0 = 0.01
delte = 0.002
p0 = 1.0d4
maxloop = 70
x2low = 1.0e-16
dlx2 = 3.0
spl = -5.0
dsp = 0.02

```

```

do 3 i = 1, va
  index(i) = 0
  scale(i) = 1.0
3  fn(i) = 0.0
c
  open(unit=4,file='mlte.nlst',status='old')
  open(unit=6, file='mlte.out', status='unknown')
  open(unit=7,file='mlte.t1',status='unknown')
  open(unit=8,file='mlte.t2',status='unknown')
  open(unit=9,file='mlte.b1',status='unknown')
  open(unit=22, file='mlte.tbl', status='unknown')
c
c  read namelist
c  read(4, mltein)
c  write(6, mltein)
c
c  if ((.not. lplot1) .and. (.not. lplot2)) then
c
c    call agseti('DASH/SELECTOR.',9)
c    call agseti('DASH/LENGTJ.', 28)
c    call agsetf('DASH/DOLLAR-QUOTE.',0.004)
c    do 801 i = 1, 9
c      call agsetc(agdshn(i), dshp(i))
801  continue
c
c    call agsetc('LABEL/NAME.', 'L')
c    call agseti('LINE/NUMBER.', 100)
c    call agsetc('LINE/TEXT.','MOLE FRACTION$')
c
c    call agsetc('LABEL/NAME.', 'B')
c    call agseti('LINE/NUMBER.',-100)
c    call agsetc('LINE/TEXT.','$')
c    call agseti('LINE/NUMBER.',-120)
c    call agsetc('LINE/TEXT.','TEMPERATURE (1000 K)$')
c
c  endif
c
c
c
c  do 200 np = 1, npp
c    lnp = .true.
c    p0 = 10.0*p0
c  write(6, 913) p0
913  format('  p0 = ', lpe12.3/

```

```

. '   temp      O2      O2+      CO      CO+',
. '      C2/'    O      O+      C      C+')
titl(32:40) = itag
p0t = p0/1.3332d3
write(titl(16:22), 99) p0t
99  format(f7.1)
    write(6, 92) p0
92  format(' Pressure =', 1pe12.3)
c
c
c
    te = t0
    do 100 nt = 1, ntp
    te = te + delte
    tk = te*11600.0
    temp(nt) = tk/1000.0
    tvv = tv
    if (tvv .le. 0.0) tvv = tk
    trr = tr
    if (trr .le. 0.0) trr = tk
c
c
    call mlte( tk, tvv, trr, den)
c
    if (nt .eq. 1) call mlte( tk, tvv, trr, den)
    if (nt .eq. 1) call mlte( tk, tvv, trr, den)
c
c
    write(6, 90) te, tk, (den(i), i = 1, id)
90  format(' te = ', f8.4, ' (eV) ', ' tk = ', f8.0, ' (K)'/
.    ('den =', 1p4e15.4))
c
c
    stk(nt) = tk
    do 23 i = 1, idp
23  sden(i, nt) = den(i)
c
    sumden = 0.0d0
    do 13 i = 1, id
13  sumden = sumden + den(i)

    do 12 i = 1, id
c    pden(i) = den(i)/sumden
12  dens(nt, i) = den(i)/sumden
c

```


V. DATA Molecular Data

molqc2.in

C2	Name	Te	24.0	0.0	-50087.0	wx	wy	wz	Be	alph	re	D0
x3	1 S+g	0.0	1854.71	13.34	-0.172			+0.00000	1.8198	0.01765	1.2425	29036.
a	3 P u	716.24	1641.35	11.67	+			+0.00000	1.6324	0.01661	1.3119	29036.
b	3 S-g	6434.27	1470.45	11.19	+0.028			+0.00000	1.4985	0.01634	1.3692	29036.
A	1 P u	8391.00	1608.35	12.07	-0.010			+0.00000	1.61634	0.01686	1.3184	29036.
c	3 S+u	13312.1	1961.6	13.7				+0.00000	1.87	0.02	1.23	29036.
d	3 P g	20022.50	1788.22	16.440	-0.5067			+0.00000	1.7527	0.01608	1.2661	29036.
C	1 P g	34261.3	1809.1	15.81	+			+0.00000	1.7834	0.0180	1.2552	50000.
e	3 P g	40796.65	1106.56	39.260	+2.8050			-0.1271	1.1922	0.0242	1.5351	50000.
D	1 S+u	43239.44	1829.57	13.940	+			+0.00000	1.8332	0.0196	1.2360	50000.
E	1 S+u	55034.7	1671.5	40.020	+0.248			+0.00000	1.7897	0.0387	1.2529	61000.

molqco.in

CO	Name	Te	28.0	0.0	-89462.5	wx	wy	wz	Be	alph	re	D0
X	1 S+	0.0	2169.81	13.281	-0.01051			+0.000054	1.6912	0.01904	1.2057	75816.
a	3 P	48686.70	1743.41	14.361	+0.045			+0.00250	1.6912	0.01904	1.20574	75816.
a'	3 S+	55825.5	1218.60	10.468	-0.0091			+0.00259	1.3446	0.0189	1.352	75816.
d	3 D	61120.1	1171.94	10.635	-0.0785			-0.001634	1.3108	0.0178	1.3696	75816.
e	3 S-	64230.24	1117.72	10.686	-0.1174			-0.00000	1.2836	0.01753	1.3840	75816.
A	1 P	65075.8	1518.24	19.40	-0.766			+0.00000	1.6115	0.02325	1.2353	75816.
I	1 S-	65084.40	1092.22	10.704	-0.0554			+0.00000	1.275	0.01848	1.3911	75816.
D	1 D	65928.	.1094.	10.20	+0.0000			+0.00000	1.257	0.017	1.399	75816.

molqcop.in

CO+	28.0	112917.0	-89462.5						
Name	Te	w	wx	wy	wz	Be	alph	re	D0
X 2 S+	0.0	2214.24	15.164	-0.0007	+0.00000	1.9772	0.01896	1.1150	79848.
A 2 P	20733.3	1562.06	13.53	+0.0131	+0.00000	1.5894	0.01942	1.2436	79848.
B 2 S+	45876.7	1734.18	27.927	+0.3283	+0.00000	1.7999	0.03025	1.1686	79848.
C 2 D	63012.	1144.	33.3	+0.0000	+0.00000	1.357	0.024	1.346	79848.

molqn2.in

N2	28.0	0.0	-78714.4						
Name	Te	w	wx	wy	wz	Be	alph	re	D0
X 1 S+g	0.0	2358.57	14.324	-0.00226	-0.00024	1.99824	0.017318	1.09768	78687.
A 3 S+u	50203.63	1460.64	13.872	+0.0103	-0.00197	1.4546	0.0180	1.2866	78687.
B 3 P g	59619.35	1733.39	14.122	-0.0569	+0.00361	1.63745	0.01791	1.2126	97592.
B' 3 S-u	66272.47	1516.88	12.181	+0.04186	-0.00732	1.4733	0.01666	1.2784	107271.
a' 1 S-u	68152.66	1530.254	12.0747	+0.04129	-0.00029	1.4799	0.01657	1.2755	119369.
a 1 P g	69283.06	1694.208	13.9491	+0.007935	+0.000291	1.6169	0.01793	1.2203	119369.
w 1 D u	72097.4	1559.26	11.63	-0.0000	+0.0000	1.498	0.0166	1.268	119369.
G 3 D g	87900.	742.49	11.85	-0.0000	+0.0000	0.928	0.0161	1.6107	119369.
C 3 P u	89136.88	2047.178	28.445	+2.08833	-0.5350	1.82473	0.01868	1.14869	97592.
E 3 S+g	95858.	2185.			+0.0000	1.9273	0.02	1.117	97592.
C' 3 P u	98351.	791.	33.5		+0.0000	1.0496	0.02	1.5146	99000.

molqn2p.in

N2+	28.0	125666.0	-78714.4						
Name	Te	w	wx	wy	wz	Be	alph	re	D0
X 2 S+g	0.0	2207.00	16.10	-0.040	+0.00000	1.932	0.01881	1.11642	70363.
A 2 P u	9166.95	1903.70	15.02	-0.0000	+0.00000	1.744	0.01883	1.1749	70363.
B 2 S+u	25461.5	2419.84	23.19	-0.5375	-0.0495	2.0746	0.024	1.0742	70363.
a 4 S+u	25467.	2398.	14.	-0.0000	+0.00000	2.071	0.014	1.07	70363.
D 2 P g	52318.2	907.71	11.91	-0.016	+0.00000	1.113	0.020	1.471	70363.
C 2 S+u	64608.1	2071.5	9.29	-0.4300	+0.00000	1.5098	0.05	1.2628	90333.

molqno.in

NO	Name	Te	30.0	0.0	-52399.9	w	wx	wy	wz	Be	alph	re	D0
X	2 P												
A	2 S+	0.0	1904.03	14.075	+0.0110				+0.00000	1.7046	0.0178	1.1508	52264.
B	2 P+	43965.7	2374.31	16.106	-0.0465				+0.00000	1.9965	0.01915	1.0634	64520.
C	2 P	45942.6	1039.8	8.3	+0.17				+0.00000	1.152	0.012	1.416	64520.
D	2 S+	52126.0	2395.	15.	+				+0.00000	2.000	0.030	1.062	64520.
E	2 S+	53084.7	2323.90	22.885	+0.75				-0.22000	2.0026	0.02175	1.0618	68557.
F	2 S+	60628.8	2375.3	16.43	+				+0.00000	1.9863	0.0182	1.0661	70976.
F	2 D	61800.	2394.	20.	+				+0.00000	1.982	0.023	1.067	70976.

molqnop.in

NO+	Name	Te	30.0	74203.0	-52399.9	w	wx	wy	wz	Be	alph	re	D0
X	1 S+												
a	3 S+	0.0	2376.42	16.262	+0.0				+0.00000	1.997	0.01889	1.06322	87913.
b	3 P	52190.0	1293.0	15.1	+0.0				+0.00000	1.369	0.0192	1.284	87913.
w	3 D	59240.0	1710.0	14.0	+0.0				+0.00000	1.634	0.0184	1.175	94366.
b'	3 S-	61880.0	1313.0	10.6	+0.0				+0.00000	1.377	0.0192	1.280	94366.
A'	1 S+	67720.0	1284.0	10.7	+0.0				+0.00000	1.357	0.0192	1.290	94366.
W	1 D	69540.0	1283.0	13.5	+0.0				+0.00000	1.363	0.0192	1.287	94366.
A	1 P	71450.0	1278.0	16.0	+0.0				+0.00000	1.361	0.0192	1.288	94366.
		73471.72	1601.93	20.207	-0.2683				+0.00000	1.586	0.0245	1.1931	94366.

molqo2.in

O2	Name	Te	32.0	0.0	-41259.9	w	wx	wy	wz	Be	alph	re	D0
X	3 S-g												
a	1 D g	0.0	1580.19	11.98	+0.0474				-0.00127	1.44566	0.01593	1.20752	40972.
b	1 S+g	7918.1	1483.5	12.9	-				+0.00000	1.4264	0.0171	1.2155	40972.
c	1 S-u	13195.1	1432.77	14.00	-				+0.00000	1.40037	0.01820	1.22675	40972.
A'	3 D u	33057.3	794.29	12.736	-0.2444				+0.00055	0.915	0.01391	1.517	40972.
A	3 S+g	34690.	850.	20.	-0.550				+0.00000	0.96	0.026	1.48	40972.
B	3 S+g	35397.8	799.07	12.16	-0.139				+0.00000	0.9106	0.01416	1.5215	40972.
D	3 S+u	49793.28	709.31	10.65	-0.000				+0.00000	0.819	0.01206	1.604	56458.
f	1 S+u	75260.	1957.	19.7	-0.000				+0.00000	1.739	0.025	1.14	75300.
		76091.	1927.	19.0	-0.000				+0.00000	1.703	0.020	1.113	76200.

Atomic Data

atgar.in

1	19	Ar	1	39.948	0.0	0.0
15.76						
.0000	1.0	3s2	3p6	1S(0)		
11.5480	5.0	3s2	3p5	4s1	3P(2)	
11.6240	3.0	3s2	3p5	4s1	3P(1)	
11.7230	1.0	3s2	3p5	4s1	3P(0)	
11.8280	3.0	3s2	3p5	4s1	1P(1)	
12.9070	3.0	3s2	3p5	4p1	3S(1)	
13.0760	7.0	3s2	3p5	4p1	3D(3)	
13.0930	5.0	3s2	3p5	4p1	3D(2)	
13.1530	3.0	3s2	3p5	4p1	3D(1)	
13.1720	5.0	3s2	3p5	4p1	3P(2)	
13.2730	1.0	3s2	3p5	4p1	3P(0)	
13.2830	3.0	3s2	3p5	4p1	3D(1)*	
13.3020	5.0	3s2	3p5	4p1	3P(2)*	
13.3280	3.0	3s2	3p5	4p1	3P(1)	
13.4800	1.0	3s2	3p5	4p1	1S(0)	
14.3162	60.0	3s2	3p5	3d1		
15.1662	144.0	3s2	3p5	4l1		
15.3763	300.0	3s2	3p5	5l1		
15.6366	432.0	3s2	3p5	6l1		

atqarp.in

2	5	Ar	2	39.948	15.76	0.0
27.63						
.0000	6.0	3s2	3p5	2P		
15.0409	2.0	3s1	3p6	2S		
18.8387	64.0	3s2	3p4	3d	4L	
22.0976	86.0	3s2	3p4	3d	2L	
24.9560	480.0	3s2	3p4	4l1		

atqc.in

1	12	C	I	12.0	0.0	0.0
11.260						
0.0		9.0	2P(2),3P	G		
1.264		5.0	2P(2),1D			
2.684		1.0	2P(2),1S			
4.183		5.0	2P(3),5S			
7.485		9.0	2P3S,3P			
7.685		3.0	2P3S,1P			
7.946		15.0	2P(3),3D			
8.727		27.0	2P3P,3SPD			
8.866		9.0	2P3P,1SPD			
9.330		9.0	2P(3),3P			
9.706		15.0	2P3D,1PDF			
9.729		45.0	2P3D,3PDF			

atqcp.in

2	10	C	II	12.0	11.26	0.0
24.383						
0.0		6.0	2S(2)2P,2P	G		
5.336		12.0	2P(2),4P			
9.290		10.0	2P(2),2D			
11.963		2.0	2P(2),2S			
13.719		6.0	2P(2),2P			
17.075		18.0	3SPD,DOUB			
17.609		4.0	2P(3),4S			
18.655		10.0	2P(3),2D			
20.676		32.0	4SPDF,DOUB			
20.921		6.0	2P(3),2P			

atqn.in

1 9 N 1 14.0 0.0 0.0
14.53

0.0000	4.0	1s2	2s2	2p3, 4S
2.2435	10.0	1s2	2s2	2p3, 2D
3.1185	6.0	1s2	2s2	2p3, 2P
10.2499	12.0	1s2	2s2	2p2 3s1, 4L
11.6727	36.0	1s2	2s2	2p2 3p1, 4L
11.8199	18.0	1s2	2s2	2p2 3s1, 2L
12.8907	60.0	1s2	2s2	2p2 3d1, 4L
13.2904	54.0	1s2	2s2	2p2 3p1, 2L
13.9866	80.0	1s2	2s2	2p2 3d1, 2L

atqnp.in

2 12 N 2 14.0 14.53 0.0
29.60

0.0000	9.0	1s2	2s2	2p2, 3P
1.9571	5.0	1s2	2s2	2p2, 1D
4.8372	1.0	1s2	2s2	2p2, 1S
5.1401	5.0	1s2	2s1	2p3, 5S
11.4279	15.0	1s2	2s1	2p3, 3D
13.3628	9.0	1s2	2s1	2p3, 3P
18.7515	9.0	1s2	2s2	2p1 3s1, 3P
18.9172	3.0	1s2	2s2	2p1 3s1, 1P
21.1874	27.0	1s2	2s2	2p1 3p1, 3L
21.7307	9.0	1s2	2s2	2p1 3s1, 1L
23.4969	45.0	1s2	2s2	2p1 3d1, 3L
23.6856	15.0	1s2	2s2	2p1 3d1, 1L

atqo.in

1 15 O 1 16.0 0.0 0.0
13.618

0.0000	9.0	1s2	2s2	2p4, 3P
1.7453	5.0	1s2	2s2	2p4, 1D
3.4190	1.0	1s2	2s2	2p4, 1S
8.8402	5.0	1s2	2s2	2p3 3s1, 5S
9.1286	3.0	1s2	2s2	2p3 3s1, 3S
10.4459	15.0	1s2	2s2	2p3 3p1, 5P
10.7228	9.0	1s2	2s2	2p3 3p1, 3P
11.8222	25.0	1s2	2s2	2p3 3d1, 5D
11.8302	15.0	1s2	2s2	2p3 3d1, 3D
11.9014	15.0	1s2	2s2	2p3 3s1, 3D
12.0451	5.0	1s2	2s2	2p3 3s1, 1D
13.4454	15.0	1s2	2s2	2p3 3p1, 3D
13.4496	3.0	1s2	2s2	2p3 3p1, 1P
13.4798	21.0	1s2	2s2	2p3 3p1, 3F
13.5079	7.0	1s2	2s2	2p3 3p1, 1F

atqop.in

2 10 O 2 16.0 13.618 0.0
35.118

0.0000	4.0	1s2	2s2	2p3, 4S
3.0056	10.0	1s2	2s2	2p3, 2D
4.2706	6.0	1s2	2s2	2p3, 2P
15.4930	12.0	1s2	2s1	2p4, 4P
22.8462	12.0	1s2	2s2	2p2 3s1, 4P
24.9628	18.0	1s2	2s2	2p2 3s1, 2L
25.6736	36.0	1s2	2s2	2p2 3p1, 4L
27.9275	54.0	1s2	2s2	2p2 3p1, 2L
28.6766	60.0	1s2	2s2	2p2 3d1, 4L
30.6400	90.0	1s2	2s2	2p2 3d1, 2L

VI. SPECIFIC HEAT

The input to CHEMKIN is the Enthalpy and its derivative with respect to temperature, C_p . We start by calculating the excitation energy:

$$U_l = \frac{\sum_i E_i g_i \exp(-E_i / kT)}{\sum_i g_i \exp(-E_i / kT)}. \quad (35)$$

In Eq. 35 the summation is over bound states for atoms and over electronic, vibrational and rotational states for molecules. Ionization and dissociation potentials are not needed since they cancel in the numerator and denominator. At this point we also define:

$$W_l = \frac{\sum_i E_i^2 g_i \exp(-E_i / kT)}{\sum_i g_i \exp(-E_i / kT)}. \quad (36)$$

From Eq. 35 we can calculate

$$\frac{\partial U_l}{\partial T} = (W_l - U_l^2) / kT^2. \quad (37)$$

By definition:

$$H_l = H_l^0 + \frac{5}{2} kT + U_l, \quad (38)$$

thus

$$C_p = \frac{\partial H_l}{\partial T} = k \left[\frac{5}{2} + (W_l - U_l^2) / (kT)^2 \right]. \quad (39)$$

The results of these calculations are presented in Figures 25 - 48. These results are presented first as the cumulative function as each electronic state is added. In the next figure the MLTE results (solid line) are compared to those of [Ref. 7] (dashed line). Recall that the data of [Ref. 7] is only valid up to 6000 °K. For all atoms and all molecules other than C_2 and O_2 , the MLTE results are a little higher, but very close (up to 6000 °K) to those of [Ref. 7]. Also note that for all molecules other than C_2 , the cumulative C_p is a monotonic nondecreasing function of electronic states. That is, each electronic states has a positive contribution to the total C_p for the molecule. For C_2 , the addition of the “a $^3\Pi_u$ ” first excited electronic state to the ground state contribution, significantly decreases C_p . Each additional electronic state contributes positively to C_p .

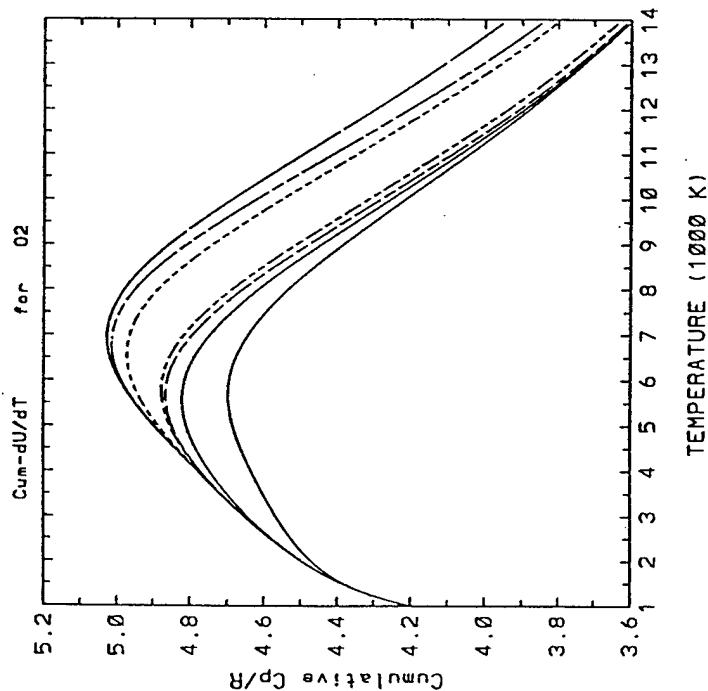


Fig. 25 ←

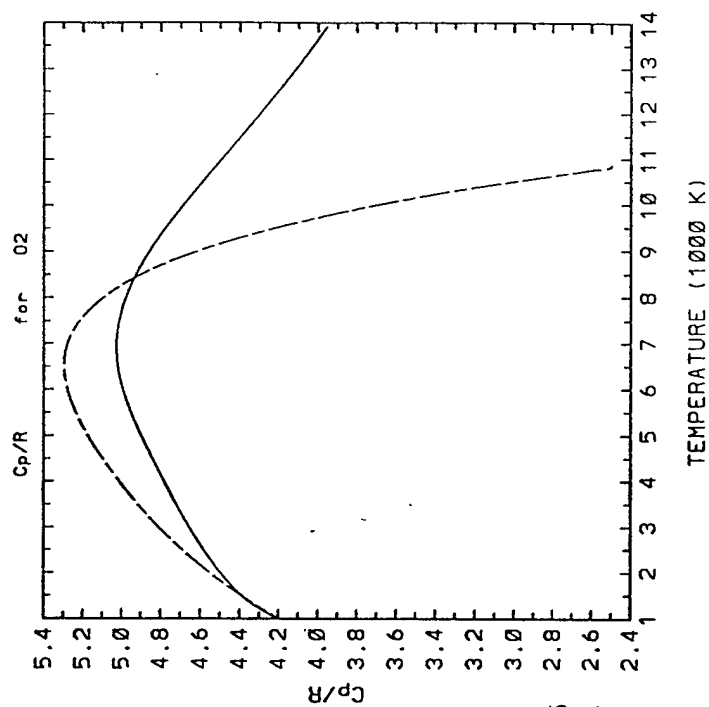


Fig. 26 →

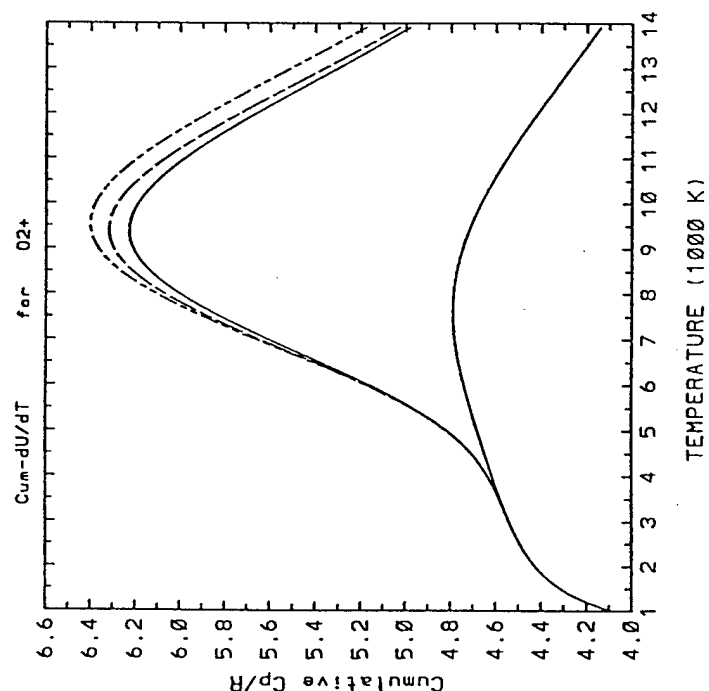


Fig. 27 ←

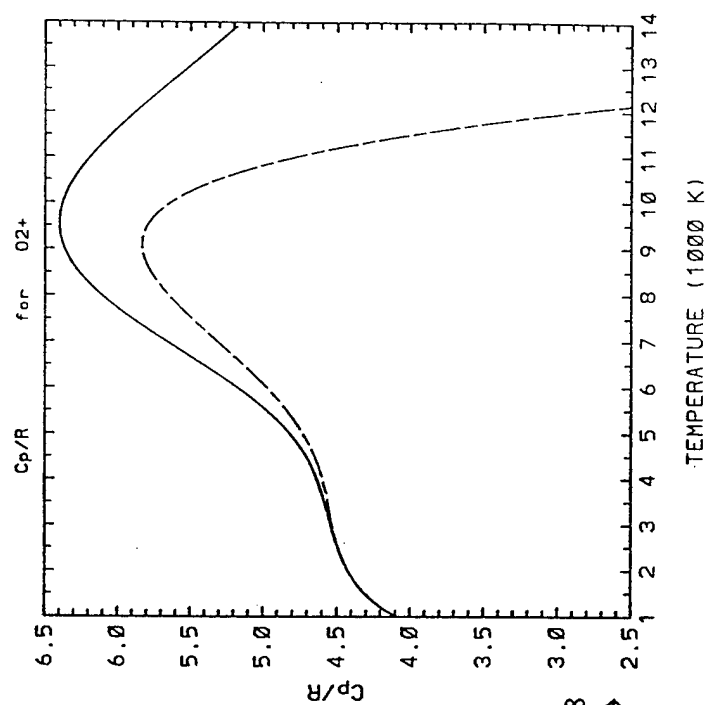


Fig. 28 →

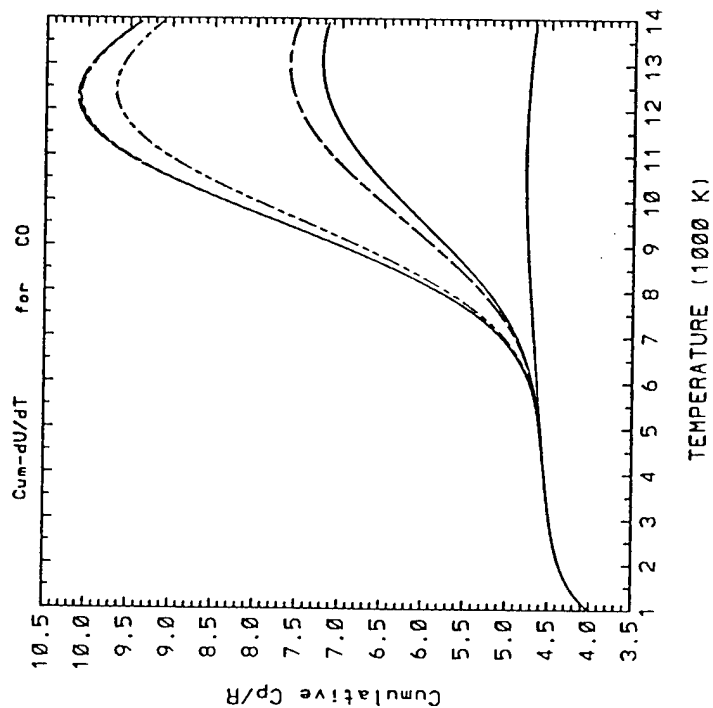


Fig. 29 ←

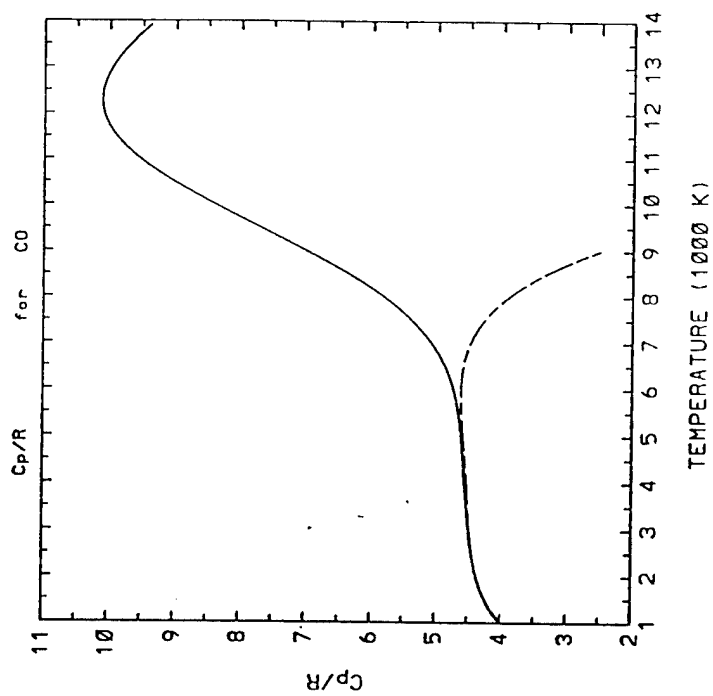


Fig. 30 →

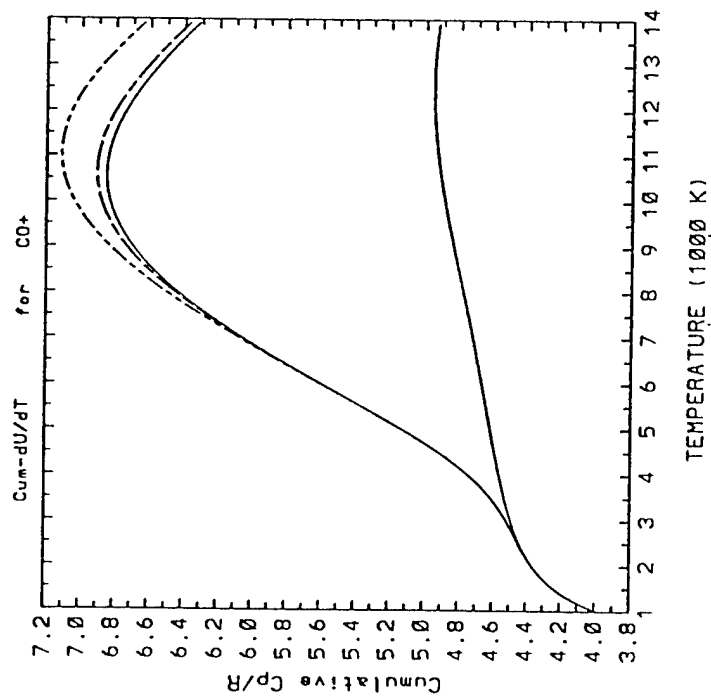


Fig. 31 ←

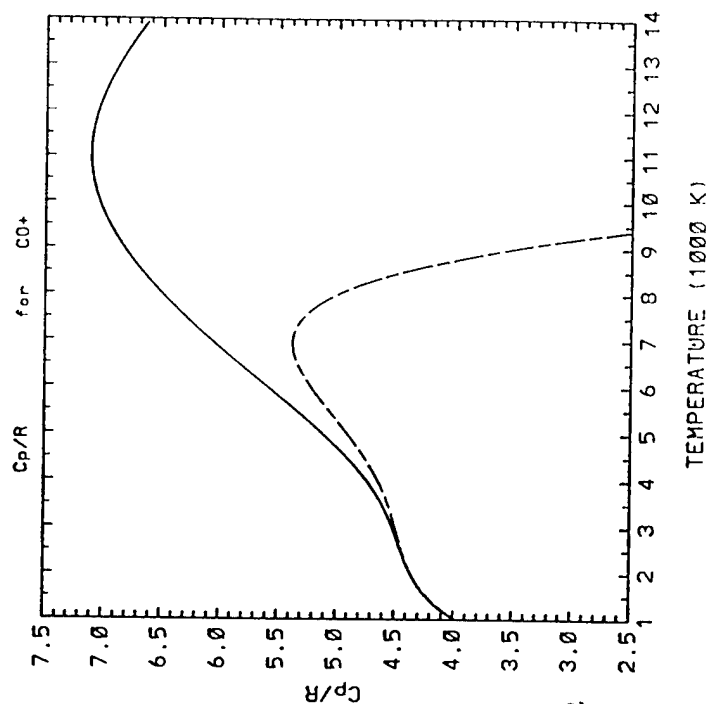


Fig. 32 →

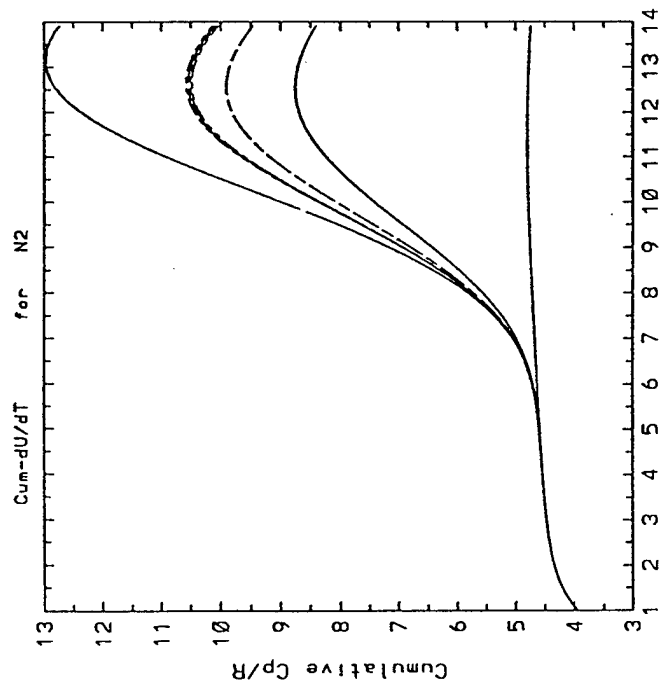


Fig. 33 ←

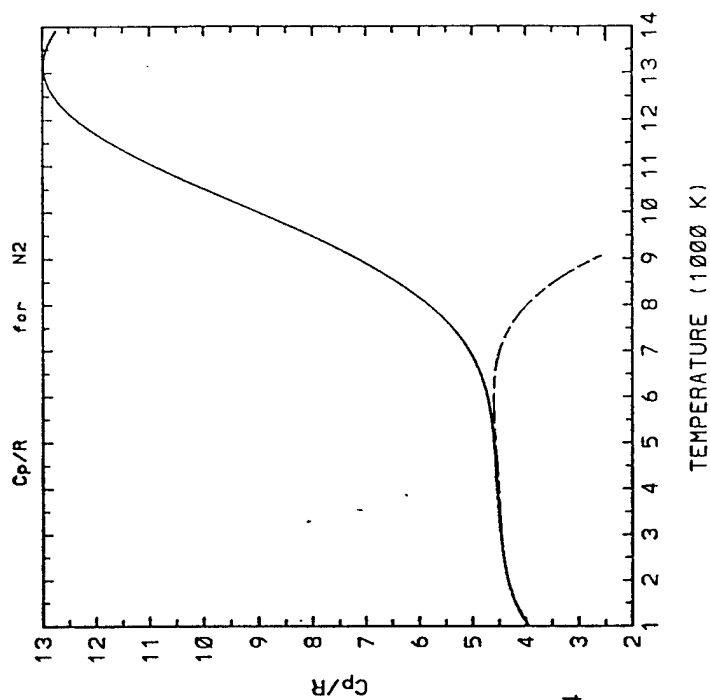


Fig. 34 →

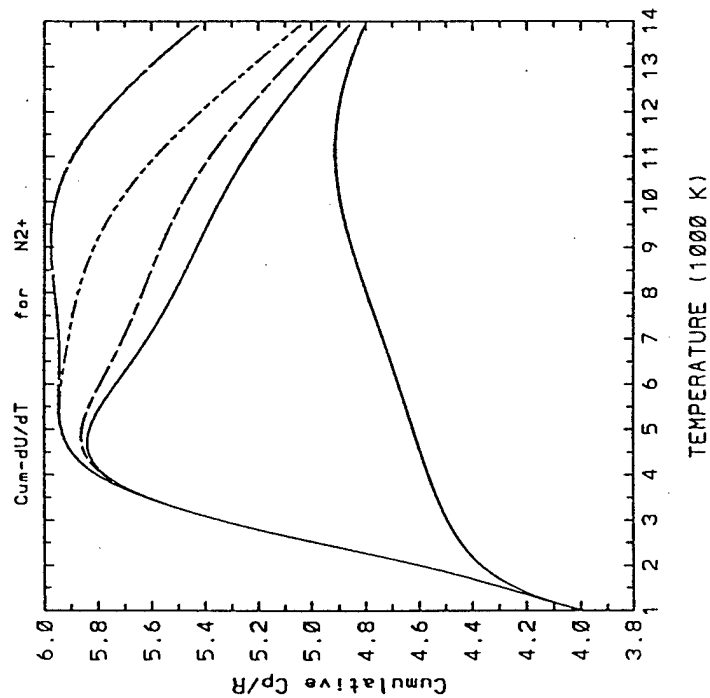


Fig. 35 ←

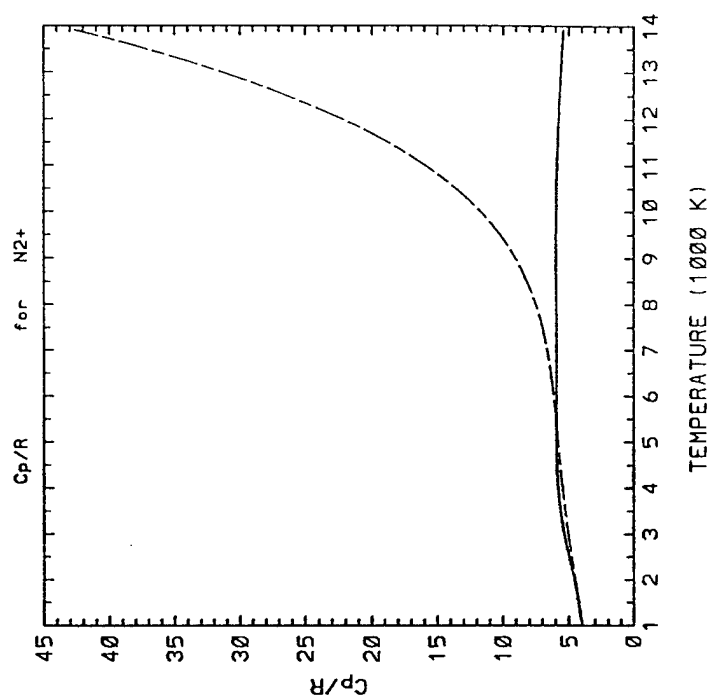


Fig. 36 →

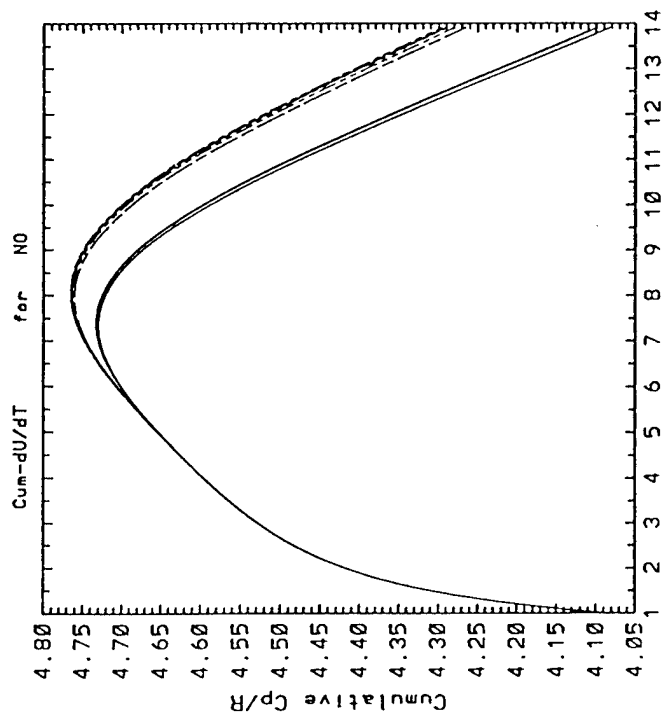


Fig. 37 ←

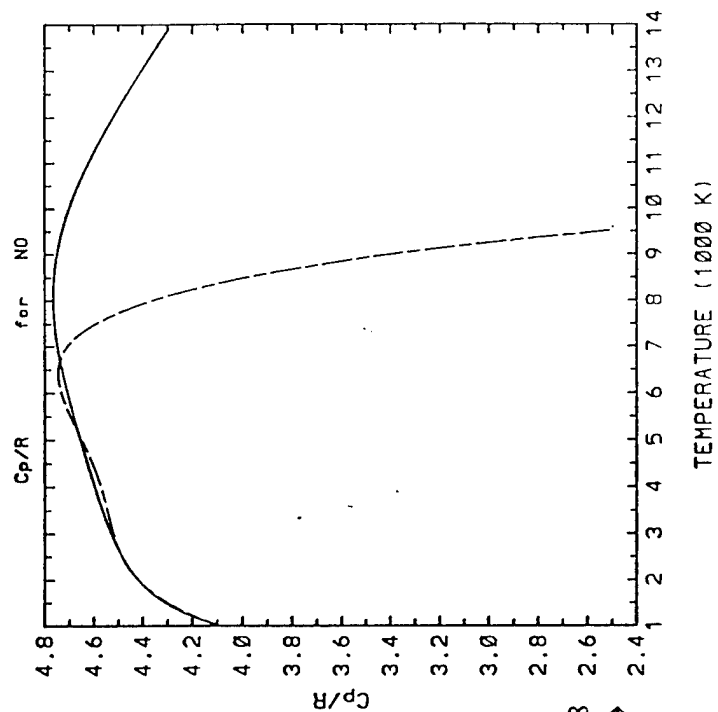


Fig. 38 →

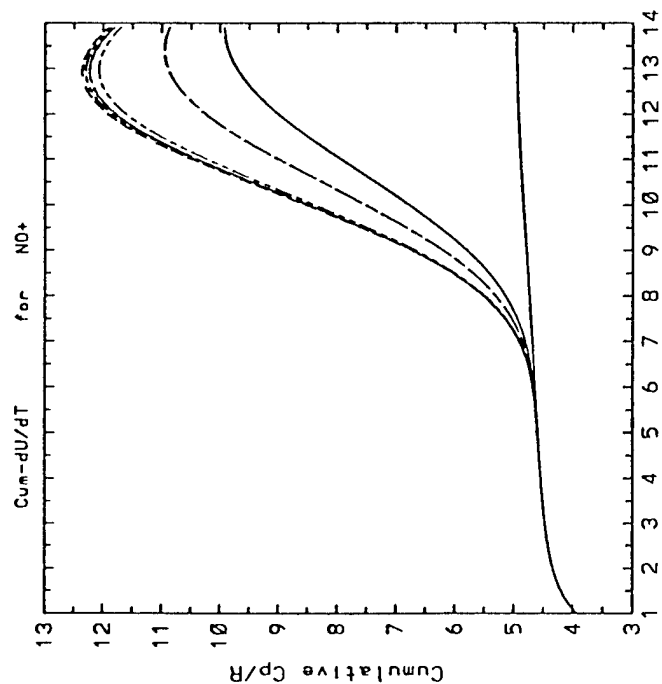


Fig. 39 ←

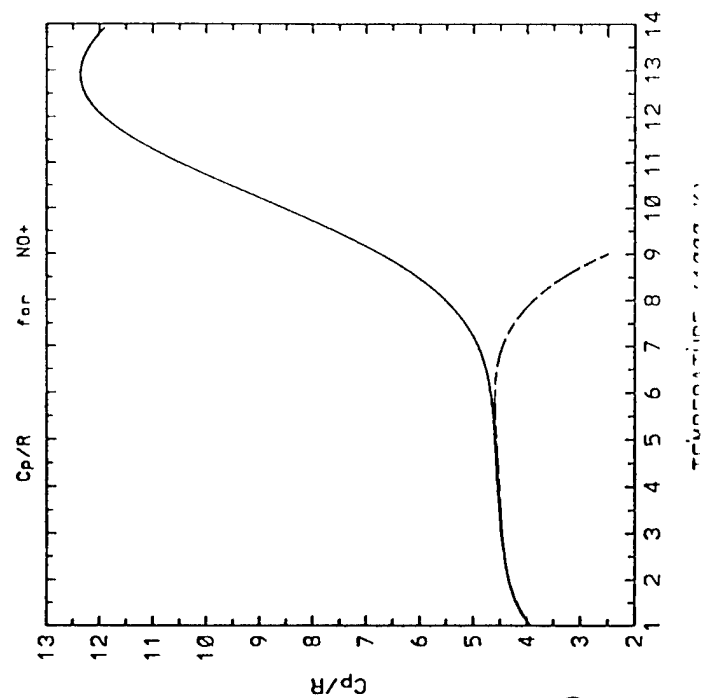


Fig. 40 →

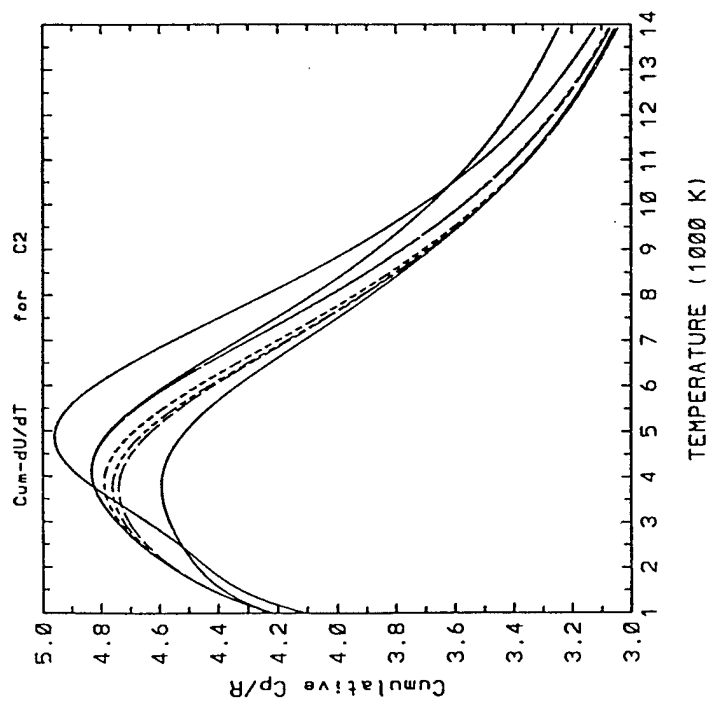


Fig. 41 ←

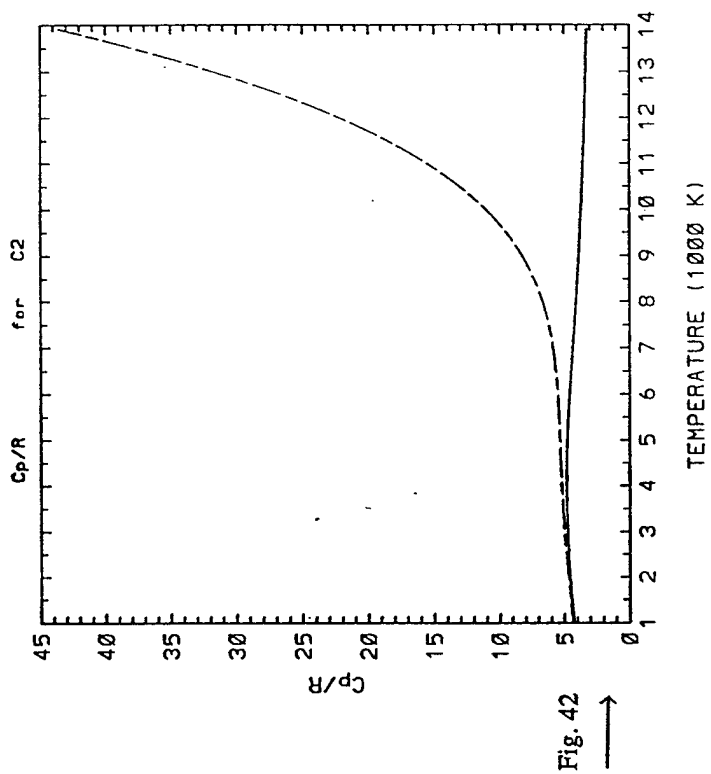


Fig. 42 →

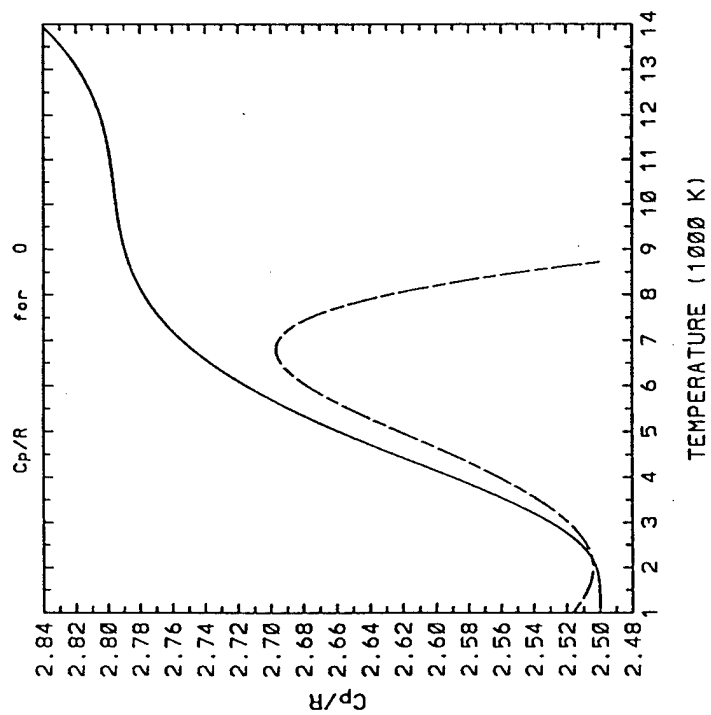


Fig. 43 ←

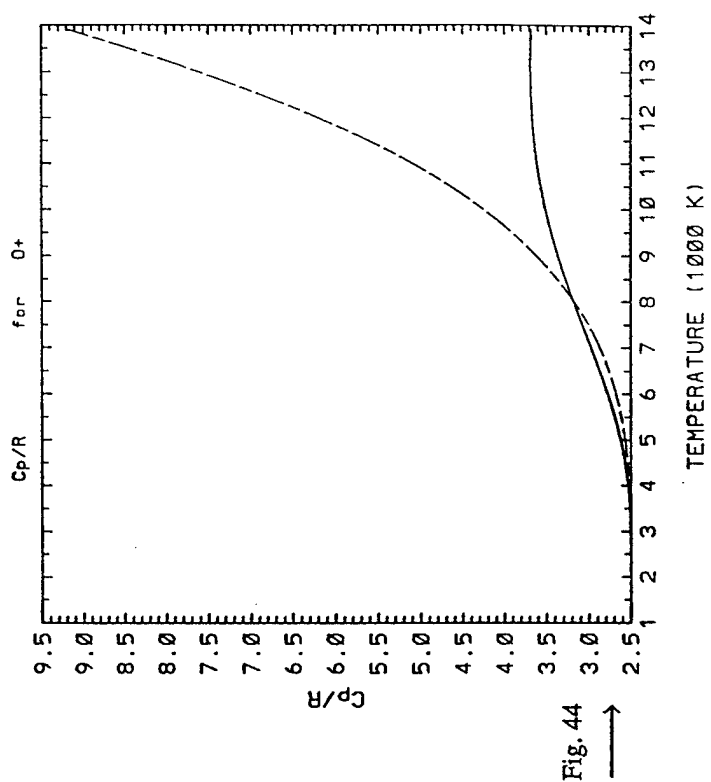


Fig. 44 →

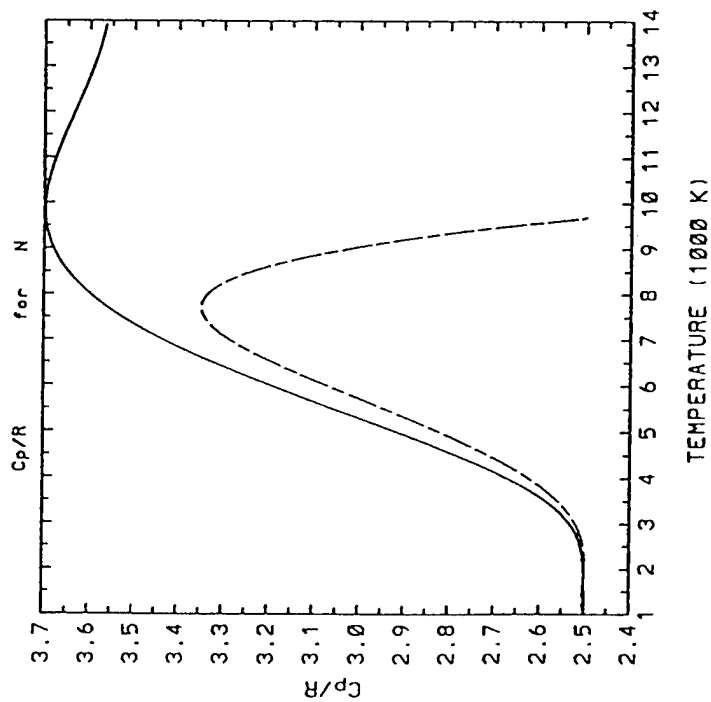


Fig. 45

←

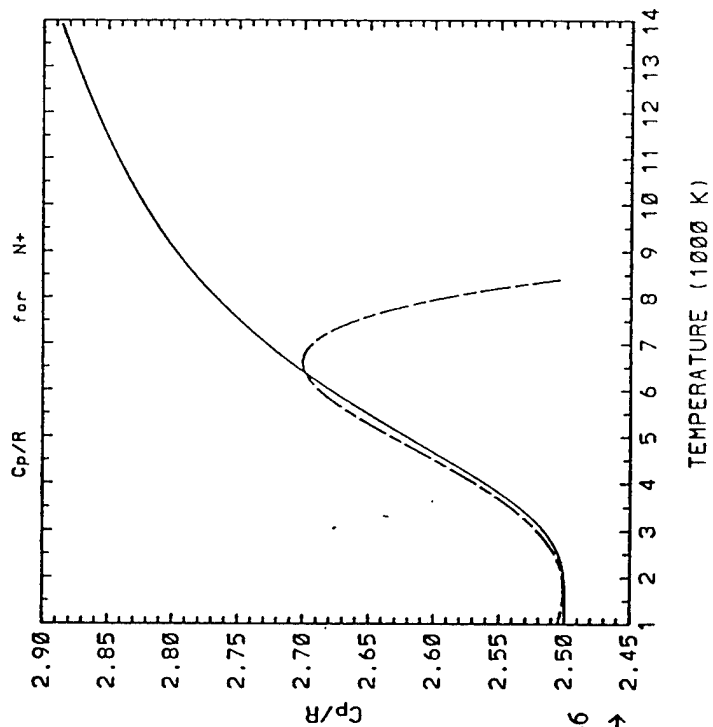


Fig. 46

→

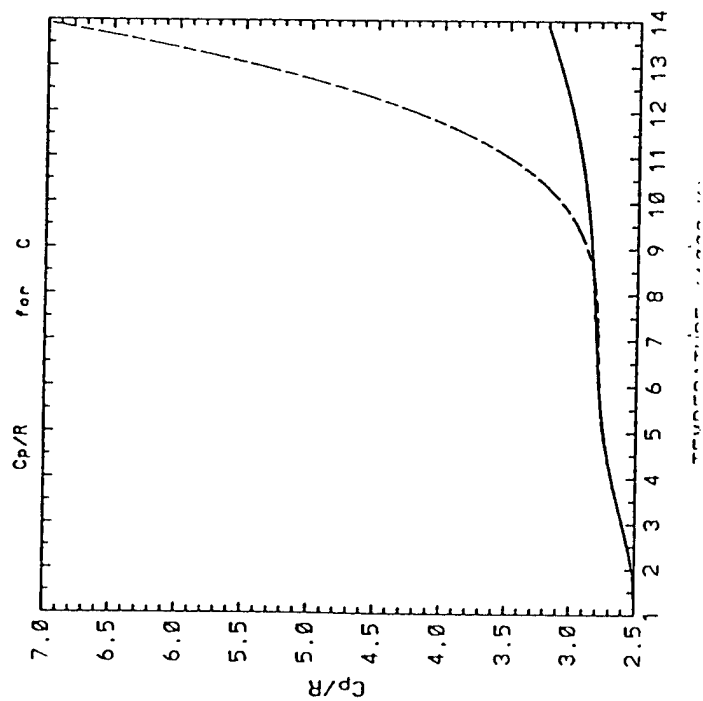


Fig. 47

←

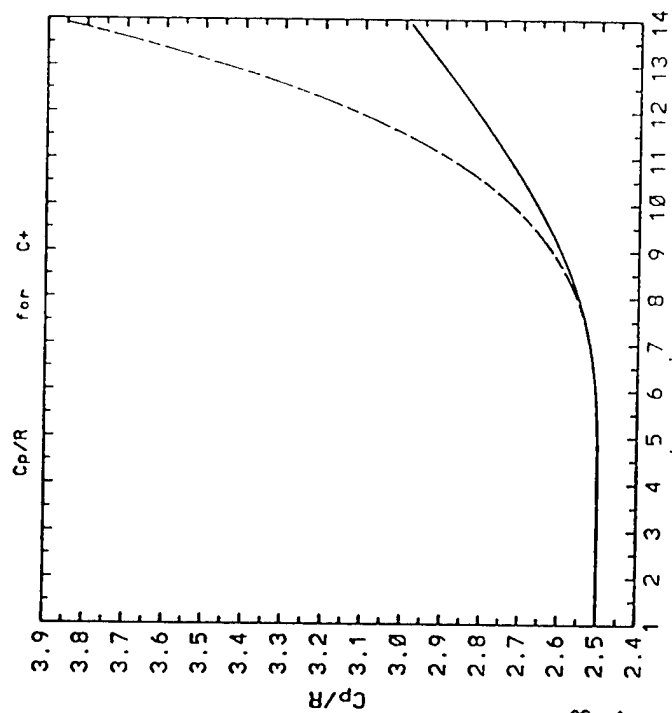


Fig. 48

→

With the technique described in Appendix A the heat capacity data can not only be put into the convenient and compact form of a fourth order polynomial in T , it can also be made to be continuous to the data of [Ref. 7] at 1000 °K. In this form the heat capacity data is shown in Table I and is valid over a temperature range from 1,000 °K to 20,000 °K.

TABLE I

Coefficients for the expansion: $C_p = \sum_{i=0}^4 X_i T^i$

1,000°K $\leq T \leq$ 20,000°K

	X_0	X_1	X_2	X_3	X_4
O2	3.84564E+00	3.84552E-04	-3.49093E-08	2.39363E-13	2.41539E-17
O2+	4.30658E+00	-3.54956E-04	1.65847E-07	-1.48660E-11	3.70888E-16
NO	3.81893E+00	2.95232E-04	-2.66967E-08	5.76425E-13	-2.66467E-19
NO+	4.95483E+00	-1.31580E-03	3.03675E-07	-1.43951E-11	1.15619E-16
N2	4.92123E+00	-1.26664E-03	2.88429E-07	-1.23135E-11	5.11080E-17
N2+	3.04607E+00	1.05458E-03	-1.24073E-07	5.81310E-12	-1.05991E-16
CO	4.99731E+00	-1.33305E-03	3.45703E-07	-2.18446E-11	3.87341E-16
CO+	4.07416E+00	-2.32754E-04	1.48826E-07	-1.25583E-11	2.89677E-16
C2	3.85173E+00	6.16230E-04	-1.24031E-07	7.67680E-12	-1.56928E-16
O	2.51193E+00	-5.80432E-06	1.16068E-08	-1.16638E-12	3.51696E-17
O+	2.69824E+00	-2.68845E-04	7.56701E-08	-5.18263E-12	1.08000E-16
N	2.61047E+00	-1.85537E-04	8.19666E-08	-7.08549E-12	1.82018E-16
N+	2.51903E+00	-2.69791E-05	1.43068E-08	-1.12102E-12	2.66681E-17
C	2.38164E+00	1.38052E-04	-2.03215E-08	1.44489E-12	-2.78334E-17
C+	2.49994E+00	6.28564E-06	-5.11023E-09	8.67188E-13	-2.55265E-17

From equation 39 we see that the Enthalpy can be calculated from the heat capacity data by integration. Thus

$$H(T) = \int_{T_0}^T C_p dT = \int_{T_0}^T C_p dT + \int_{T_1}^T C_p dT, \quad (40)$$

$$H(T) = H^0(T_1) + \int_{T_1}^T C_p dT \quad (41)$$

In [Ref. 7] the reference temperature, T_0 , is taken to be 298.15 °K. We will use this reference and their data for C_p up to $T_1 = 1,000^\circ\text{K}$ to evaluate

$$H^0(T_1) = \int_{T_0}^{T_1} C_p dT. \quad (42)$$

In the notation of [Ref. 7]:

$$H^0(T_1) = R \left[b_1 + \sum_{i=0}^{i=4} \frac{a_i T_1^{i+1}}{i+1} \right], \quad (43)$$

The coefficients b_1 and a_i are tabulated in [Ref. 7].

CHEMKIN is set up to use the data in the same format as that of [Ref. 7]. Thus we need to calculate y_1 , such that we can have

$$H(T) = R \left[y_1 + \sum_{i=0}^{i=4} \frac{X_i T^{i+1}}{i+1} \right]. \quad (44)$$

It is clear that

$$y_1 = b_1 + \sum_{i=0}^4 \left[\frac{a_i - X_i}{i+1} \right] (1,000)^{i+1}, \quad (45)$$

and are tabulated in Table II.

TABLE II

	y_1		y_1
O_2	-1.295342E+03	O_2^+	1.394564E+05
N_2	1.795155E+05	N_2^+	1.708964E+04
CO	-1.512556E+04	CO^+	1.486719E+05
NO	9.693373E+03	NO^+	1.173511E+05
C_2	9.902393E+04		
O	2.924312E+04	O^+	1.878475E+05
N	5.606134E+04	N^+	2.256235E+05
C	8.550795E+04	C^+	2.168825E+05

We can now use the data from Tables I and II to replace the CHEMKIN data for $T > 1,000$ °K, and thus extend the validity of the CHEMKIN results up to 20,000 °K.

REFERENCES

- [1.] T. M. Bresmann, "SOLGASMIX-PV, A Computer Program to Calculate Equilibrium Relationships in Complex Chemical Systems", Oak Ridge National Laboratory Report ORNL/TM-5775 (1977).
- [2.] S. R. Brinkley, Jr., J. Chem. Phys., Vol.14, p.563 (1946).
- [3.] S. R. Brinkley, Jr., J. Chem. Phys., Vol.15, p.107 (1947).
- [4.] W. B. White, W. M. Johnson, and G. B. Dantzig, J. Chem. Phys., Vol.28, p.751 (1958).
- [5.] F. J. Zeleznik and S. Gordon, "Calculation fo Complex Chemical Equilibria", Industrial Eng. Chem., Vol.60, p.27 (1968).
- [6.] P. Glarborg, R. J. Kee, J. F. Grcar, and J. A. Miller, SANDIA REPORT SAND86-8209 UC-4 (1993).
- [7.] B. J. McBride, S. Gordon, and M. A. Reno, "Coefficients for Calculating Thermodynamic and Transport Properties of Individual Species," NASA Technical Memorandum 4513 (1993).
- [8.] M. T. Mehmetoglu and W. H. Gauvin, "Characteristics of a Transferred-Arc Plasma," American Institute of Chemical Engineering, Vol.29, p.207 (1983).
- [9.] W. L. T. Chen, J. Heberlein, and E. Pfender, "Diagnostics of a Thermal Plasma Jet by Optical Emission Spectroscopy and Enthalpy Probe Measurments," Plasma Chemistry and Plasma Proicessing, Vol.14, p.317 (1994).
- [10.] J. W. Bond, Jr., K. M. Watson, and J. A. Welch, Jr., "Atomic Theory of Gas Dynamics," Addison-Wesley (Reading, MA), Chapter 5, (1965).
- [11.] G. Herzberg "Molecular Spectra and Molecular Structure: I. Spectra of Diatomic Molecules," 2nd Edition, Krieger Publishing (Malabar, FL), Table 39 (1950).
- [12.] K. P. Huber and G. Herzberg, "Molecular Spectra and Molecular Structure: IV. Constants of Diatomic Molecules", Van Nostrand Reinhold (New York), (1979).
- [13.] Press, Teukolsky, Vetterling, and Flannery, "Numerical Recipes in Fortran" Second Edition, Cambridge University Press (New York), (1992).

[14.] O. Theimer and P. Kepple, *Physical Review*, 165, 168 (1968) and *Physical Review A*, 1, 957 (1970).

[15.] B. J. McBride, S. Gordon, and M. A. Reno, "Thermodynamic Data for Fifty Reference Elements", NASA Technical Paper 3287 (1993).

APPENDIX A

LEAST SQUARE FIT WITH A CONSTRAINT

We wish to fit a set of data $\{x_i, f_i\}$ to a polynomial in x . It is known that, if the range of x is large, the usual algorithm results in a nearly singular set of linear equations. To avoid this problem we introduce a set of basis polynomials Φ_i . Ideally this set of polynomials are orthogonal over the range of x_i . That is

$$\int_a^b \Phi_i \Phi_j dx = \delta_{i,j},$$

where $a \leq x_i \leq b$. For simplicity we will use a set of polynomials:

$$\begin{aligned}\Phi_1 &= a_{1,1} \\ \Phi_2 &= a_{2,1} + a_{2,2} x \\ \Phi_3 &= a_{3,1} + a_{3,2} x + a_{3,3} x^2 \\ \Phi_4 &= a_{4,1} + a_{4,2} x + a_{4,3} x^2 + a_{4,4} x^3 \\ \Phi_5 &= a_{5,1} + a_{5,2} x + a_{5,3} x^2 + a_{5,4} x^3 + a_{5,5} x^4\end{aligned}\tag{A1}$$

and select the coefficients for approximate orthogonalization. The set of coefficients is listed in Table I and plots of $\Phi_2 - \Phi_5$ are shown in figures A1 - A4. It follows that if

$$F(x) = \sum_{i=1}^5 B_i \Phi_i(x) = \sum_{i=1}^5 C_i x^{i-1}, \text{ then}$$

A

$$\begin{aligned}C_5 &= B_5 a_{5,5} \\ C_4 &= B_5 a_{5,4} + B_4 a_{4,4} \\ C_3 &= B_5 a_{5,3} + B_4 a_{4,3} + B_3 a_{3,3} \\ C_2 &= B_5 a_{5,2} + B_4 a_{4,2} + B_3 a_{3,2} + B_2 a_{2,2} \\ C_1 &= B_5 a_{5,1} + B_4 a_{4,1} + B_3 a_{3,1} + B_2 a_{2,1} + B_1 a_{1,1}.\end{aligned}\tag{A2}$$

In the (unconstrained) least square algorithm one would minimize the residual

$$R = \sum_k \left[\sum_{i=1}^5 B_i \Phi_i(x_k) - f_k \right]^2.\tag{A3}$$

If, in addition to minimizing the above residual, one needs to impose the constraint

$$F(x^*) = \sum_{i=1}^5 B_i \Phi_i(x^*) = f^*, \quad (\text{A4})$$

then, by the technique of Lagrange multipliers, one can minimize the new function:

$$R = \sum_k \left[\sum_{i=1}^5 B_i \Phi_i(x_k) - f_k \right]^2 + 2\lambda \left[\sum_{i=1}^5 B_i \Phi_i(x^*) - f^* \right]. \quad (\text{A5})$$

This minimization is accomplished by setting the derivatives of R with respect to the coefficients B_i to zero. Thus

$$\begin{aligned} \frac{\partial R}{\partial B_j} &= 2 \sum_k \left(\sum_{i=1}^5 B_i \Phi_i(x_k) - f_k \right) \Phi_j(x_k) + 2\lambda \Phi_j(x^*) = 0, \\ \sum_{i=1}^5 B_i \sum_k \Phi_i(x_k) \Phi_j(x_k) - \sum_k f_k \Phi_j(x_k) + \lambda \Phi_j(x^*) &= 0, \end{aligned} \quad (\text{A6})$$

for $j = 1, 2, \dots, 5$. Next the equation for $j = 1$ can be solved for λ :

$$\lambda = \frac{\sum_k f_k \Phi_1(x_k) - \sum_{i=1}^5 B_i \left[\sum_k \Phi_i(x_k) \Phi_1(x_k) \right]}{\Phi_1(x^*)}. \quad (\text{A7})$$

On substitution of equation A7 into equation A6, one obtains

$$\sum_{i=1}^5 B_i \sum_k \Phi_i(x_k) \left[\Phi_j(x_k) - \frac{\Phi_j(x^*)}{\Phi_1(x^*)} \Phi_1(x_k) \right] - \sum_k f_k \left[\Phi_j(x_k) - \frac{\Phi_j(x^*)}{\Phi_1(x^*)} \Phi_1(x_k) \right] = 0, \quad (\text{A8})$$

But Φ_1 is a constant, therefore equation A8 simplifies to

$$\sum_{i=1}^5 B_i \sum_k \Phi_i(x_k) [\Phi_j(x_k) - \Phi_j(x^*)] - \sum_k f_k [\Phi_j(x_k) - \Phi_j(x^*)] \quad (\text{A9})$$

for $j = 2, 3, 4$, and 5 . These four equations plus equation 4 constitute a linear set of five equations in the five unknown B_i . If the Φ s are near enough to orthogonality, the linear equations will be well behaved.

The linear set can be written $VB = W$, where $V_{i,1} = \Phi_i(x^*)$, $W_1 = f^*$,

$$V_{i,j} = \sum_k \Phi_i(x_k) [\Phi_j(x_k) - \Phi_j(x^*)], \text{ and} \quad (\text{A10})$$

$$W_j = \sum_k f_k [\Phi_j(x_k) - \Phi_j(x^*)], \quad (\text{A11})$$

for $j = 2, \dots, 5$.

TABLE AI

Coefficients of Φ_i

$$\begin{aligned} a_{1,1} &= +1.0 \\ a_{2,1} &= -1.0 \\ a_{2,2} &= +1.0\text{e}4 \\ a_{3,1} &= +1.0 \\ a_{3,2} &= -2.67\text{e-}4 \\ a_{3,3} &= +1.30\text{e-}8 \\ a_{4,1} &= -1.0 \\ a_{4,2} &= +4.13\text{e-}4 \\ a_{4,3} &= -4.68\text{e-}8 \\ a_{4,4} &= +1.56\text{e-}12 \\ a_{5,1} &= +1.0 \\ a_{5,2} &= -4.88\text{e-}4 \\ a_{5,3} &= +7.76\text{e-}8 \\ a_{5,4} &= -4.96\text{e-}12 \\ a_{5,5} &= +1.10\text{e-}16 \end{aligned}$$

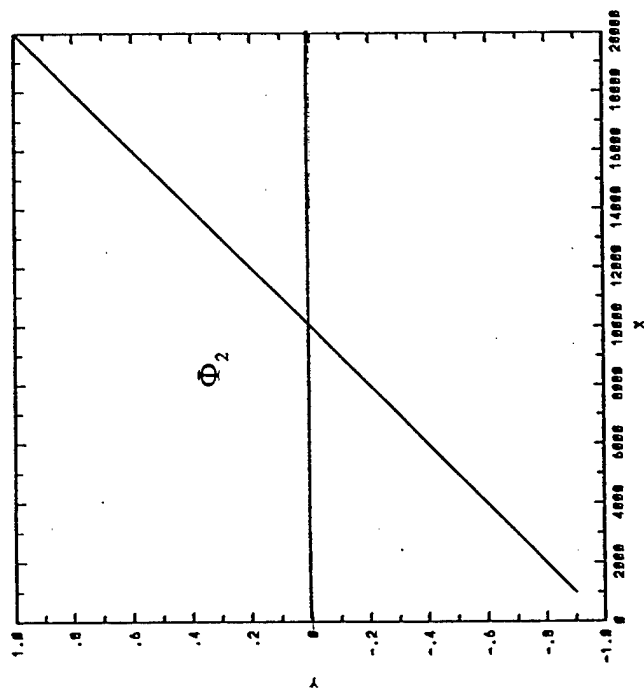


Fig. A1 ←

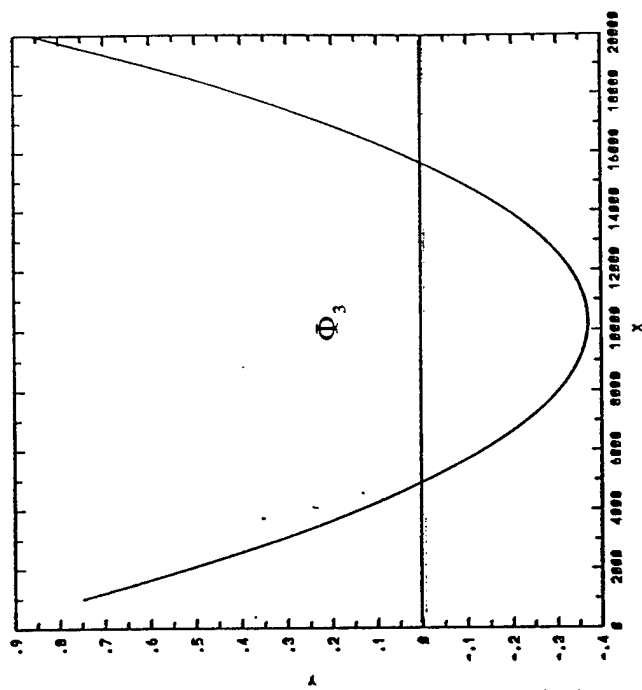


Fig. A2 →

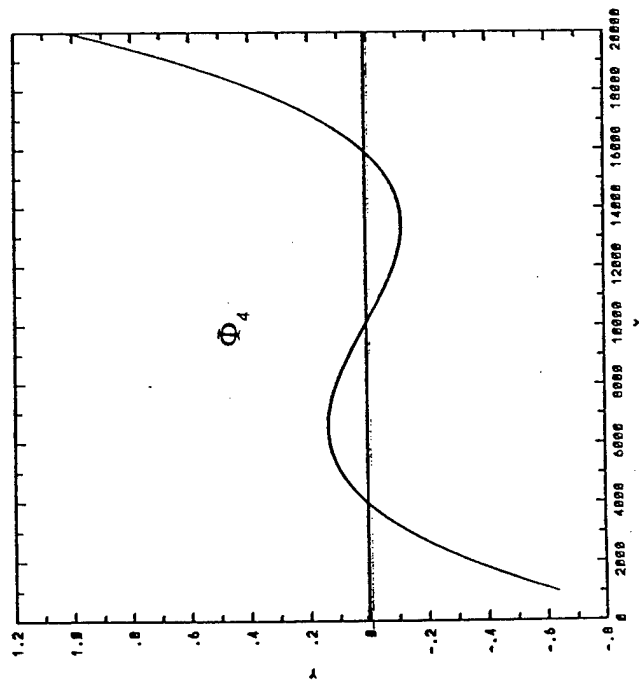


Fig. A3 ←

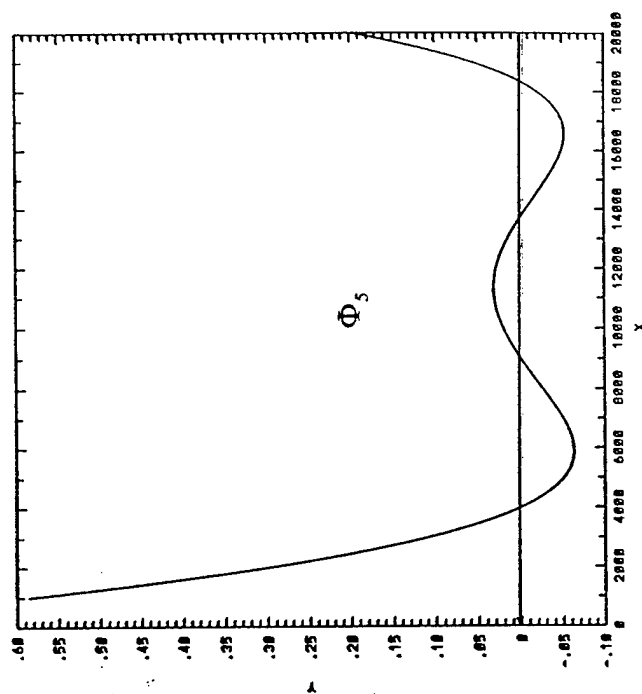


Fig. A4 →

ACKNOWLEDGMENT

This work was sponsored by the BMDO/IST.