LSENS, The NASA Lewis Kinetics and Sensitivity Analysis Code

K. Radhakrishnan
Institute for Computational Mechanics in Propulsion, Cleveland, Ohio
Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized data bases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at http://www.sti.nasa.gov

- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA Access Help Desk at (301) 621-0134

- Telephone the NASA Access Help Desk at (301) 621-0390

- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076
LSENS, The NASA Lewis Kinetics and Sensitivity Analysis Code

K. Radhakrishnan
Institute for Computational Mechanics in Propulsion, Cleveland, Ohio

Prepared for the
35th Joint Propulsion Conference and Exhibit
cosponsored by the AIAA, ASME, SAE, and ASEE
Los Angeles, California, June 20–24, 1999

Prepared under Cooperative Agreement NCC3–622

National Aeronautics and
Space Administration

Glenn Research Center

February 2000
Acknowledgments

This work was supported by the NASA Glenn Research Center through Grant NCC3–622.

Note that at the time of research, the NASA Lewis Research Center was undergoing a name change to the NASA John H. Glenn Research Center at Lewis Field. Both names appear in these proceedings.

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076
Price Code: A03

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100
Price Code: A03
LSENS, The NASA Lewis Kinetics and Sensitivity Analysis Code

K. Radhakrishnan
Institute for Computational Mechanics in Propulsion
Cleveland, OH 44135

Abstract

A general chemical kinetics and sensitivity analysis code for complex, homogeneous, gas-phase reactions is described. The main features of the code, LSENS (the NASA Lewis kinetics and sensitivity analysis code), are its flexibility, efficiency and convenience in treating many different chemical reaction models. The models include: static system; steady, one-dimensional, inviscid flow; incident-shock initiated reaction in a shock tube; and a perfectly stirred reactor. In addition, equilibrium computations can be performed for several assigned states. An implicit numerical integration method (LSODE, the Livermore Solver for Ordinary Differential Equations), which works efficiently for the extremes of very fast and very slow reactions, is used to solve the "stiff" ordinary differential equation systems that arise in chemical kinetics. For static reactions, the code uses the decoupled direct method to calculate sensitivity coefficients of the dependent variables and their temporal derivatives with respect to the initial values of dependent variables and/or the rate coefficient parameters. Solution methods for the equilibrium and post-shock conditions and for perfectly stirred reactor problems are either adapted from or based on the procedures built into the NASA code CEA (Chemical Equilibrium and Applications).

Introduction

This paper describes a FORTRAN code, LSENS [1-3], the NASA Lewis chemical kinetics and sensitivity analysis code, which has been developed for solving complex, homogeneous, gas-phase, chemical kinetics problems. The motivation for this work is the continuing interest in developing detailed chemical reaction mechanisms for complex reactions such as the combustion of fuels and pollutant formation and destruction, and reduced mechanisms for specific applications (e.g., computational fluid dynamic simulations).

Mathematical descriptions of chemical kinetics problems constitute sets of coupled, nonlinear, first order ordinary differential equations (ODEs) [4]. The number of ODEs can be very large, because of the numerous chemical species involved in the reaction mechanism. Further complicating the situation are the many simultaneous reactions needed to describe the chemical kinetics of practical fuels. For example, the mechanism describing the oxidation of the simplest hydrocarbon fuel, methane, involves over 25 species participating in nearly 100 elementary reaction steps [5].

Validating a chemical reaction mechanism requires repetitive solutions of the governing ODEs for a variety of reaction conditions. Analytical solutions to the systems of ODEs describing chemistry are not possible, except for the simplest of cases, which are of little or no practical value. Consequently, there is a need for fast and reliable numerical solution techniques for chemical kinetics problems.

In addition to solving the ODEs describing chemical kinetics, it is often necessary to know what effects variations of either initial condition values or chemical reaction mechanism parameters have on the solution. Such a need arises in the development of reaction mechanisms from experimental data [6]. The rate coefficients are often not known with great precision and in general, the experimental data are not sufficiently detailed to determine accurately the rate coefficient parameters. The development of a reaction mechanism is facilitated by a systematic sensitivity analysis, which provides the relationships between the predictions of a kinetics model and the input parameters of the problem [7-9].

The LSENS code is designed for flexibility, convenience and computational efficiency. A variety of chemical reaction models can be considered. The models include: (1) static system; (2) steady, one-dimensional, inviscid flow; (3) reaction behind an incident shock wave, including boundary layer correction; and (4) the perfectly stirred (highly back-mixed) reactor. In addition, computations of equilibrium properties can be performed for the following assigned states: (1) enthalpy and pressure; (2) temperature and pressure; (3) internal energy and volume; and (4) temperature and volume. For static problems the code computes sensitivity coefficients with respect to the initial values of the dependent variables and/or the three rate coefficient parameters of the chemical reactions. A unique feature of the code is the capability of solving multiple (essentially limitless) problems in a single computer run.

LSENS is the product of an ongoing effort at NASA Glenn to develop and upgrade algorithms and computer codes for chemical kinetics applications. The first NASA Glenn kinetics code, GCKP (General Chemical Kinetics Program [10]), was developed by Bittker and Scullin in 1972. This code uses the implicit integration method of Tyson [11], with some modifications due to Scullin [10]. Since then, many integration methods and computer codes have been developed for the efficient integration of ODEs in general and chemical kinetics equations in particular (see...
OS. It has also been successfully executed on the following systems (OS) and the Amdahl 5870 computer using the UTS Center's IBM 370/3033 computer using the TSS operating system. For flow problems, the independent variable may be either time or axial distance. In addition, the pressure, temperature, and volume profiles may be specified as functions of time or distance, irrespective of the independent variable.

This method has shown greater efficiency and stability, with equal or better accuracy, than other methods of sensitivity analysis [1,12,23,24]. Subsequently, many improvements and new options were incorporated into GCKP86, and the present version, LSENS, replaces the previous NASA general chemical kinetics codes GCKP [10], GCKP84 [13], and GCKP86 [22].

**Description of Code and Capabilities**

The LSENS code has been designed for the following reaction models and computations: (1) Static reaction at constant density or with pressure assigned either as a polynomial function of up to third degree in time or in tabular form; (2) One-dimensional flow reaction with an assigned pressure or area profile, either as (a) a polynomial function of up to third degree in time or axial distance or (b) in tabular form as a function of time or distance; (3) Static or flow reaction with an assigned temperature profile, either as (a) a polynomial function of up to third degree in time or distance or (b) in tabular form as a function of time or distance; (4) Sensitivity analysis for a static reaction; (5) Equilibrium computations for the following assigned states: (a) enthalpy and pressure, (b) temperature and pressure, (c) internal energy and volume, and (d) temperature and volume; (6) Reaction initiated by an incident shock wave, including computation of the frozen and equilibrium post-shock states; and (7) Reaction in a perfectly stirred reactor, with either an assigned mass flow rate or reactor temperature. This problem may be followed by an one-dimensional flow reaction (i.e., in a plug flow reactor).

Any static or one-dimensional flow kinetic reaction problem for which temperature is not assigned may either be adiabatic or have a prescribed rate of heat exchange with its environment. For flow problems, the independent variable can be either time or axial distance. In addition, the pressure, area, and temperature profiles may be specified as functions of time or distance, irrespective of the independent variable.

LSSENS was developed on the NASA Lewis Research Center's IBM 370/3033 computer using the TSS operating system (OS) and the Amdahl 5870 computer using the UTS OS. It has also been successfully executed on the following computer systems: NASA Lewis Research Center's Amdahl 5870 using the VM/CMS OS, Cray-X/MP/2/4 using the COS and UNICOS operating systems and the CFT and CFT77 compilers, Cray-Y/MP/8/6128 using UNICOS 6.0 and CFT77, the Alliant FX/S, the Convex C220 minicomputer using the Convex 8.0 OS, and the VAX 11/750, 11/780, 11/785, 6320, 8650, 8800, and 9410; NASA Ames Research Center's Cray-2 and Cray-Y/MP, using UNICOS and CFT77 and Cray-C90, using UNICOS and the F90 compiler; the SUN Sparcstation 1 using the Sun 4.1 OS; several IRIS workstations using the IRIX 4.0.1 OS and F77 compiler; the IBM RISC System/6000 using the AIX 3.1 OS and the XLF and F77 compilers; the Pentium I, using MS Windows 95 OS and the Digital Visual Fortran 90; and the Pentium II running the Linux OS and the Fortran compiler included therein.

The code consists of a MAIN program, 59 subprograms and a BLOCK DATA module. Five of these subprograms are function routines; all others are subroutines. The code uses the following intrinsic and external routines: ALOG, DABS, DBLE, DEXP, DFLOAT, DLOG, DMAXI, DMIN1, DSIGN, DSQRT, EXP, FLOAT, IABS, IFIX, MAX0, MIN0, MOD, READ, SNGL, and WRITE. Namelists and statement functions are also used. Finally, the code calls the system clock to obtain the total CPU time used since job initiation.

The different subprograms that comprise the LSENS package are arranged in three different groups. The first group contains the MAIN program and those related to thermodynamic, transport and kinetics computations. This group also includes the numerical procedures for the equilibrium and post-incident shock states and perfectly stirred reactor (PSR) problems. These solution methods were either adapted from or based on the NASA Chemical Equilibrium and Applications code CEA [25]. The second group includes the subroutines required for sensitivity analysis; several of these routines were adapted from the code CHEMDDM [26]. The last group contains the subprograms included in the code LSODE [20,21], which is used to solve the governing ODEs. The BLOCK DATA module is located at the end of the code. A detailed flowchart of the MAIN program is given in Fig. 1, which essentially illustrates the structure of LSENS. The routine KINP (shown in Fig. 1) processes the reaction mechanism and much of the input data required to solve the problem. The routine SENDDM is the main subroutine in that section of the code which performs sensitivity analysis. This routine also manages the calls to LSODE.

LSSENS has been arranged as much as possible in a "modular" fashion, with different subprograms performing different tasks. However, to avoid unnecessary work, some computations are performed in subprograms other than where they naturally belong. Because the code is designed to be modular, the number of subprograms is fairly large. However, this feature aids in both understanding and, if necessary, modifying the code. In addition, as improvements are made in any calculation procedures or methods built into the
code, only the subprograms using these procedures need to be replaced. An example is the computation of thermodynamic data. The relations built into the code are based on fitting data over two temperature ranges. Work is now underway to extend the temperature range over which the calculations are valid [25].

Communication among different subprograms is accomplished by means of calling sequences and common blocks, which are used extensively in LSENS. The reason for using common blocks is to avoid lengthy calling sequences, which can significantly deteriorate the efficiency of the program. Each subprogram contains type declarations for all variables used within it. Such declarations are useful for debugging and provide a list of all variables that occur in a routine. The type declarations are arranged in a specific order to enhance their utility.

The remainder of this section describes briefly the computational capabilities of LSENS. Also discussed are convenience features and calculation procedures used for thermodynamic and transport properties. Detailed descriptions of the code, the theory and numerical solution procedures built into it, and its usage and accuracy and efficiency comparisons with other methods and codes are given by Radhakrishnan and Bittker [1-3].

### Types of Chemical Reaction and Rate Coefficient

Many different types of elementary chemical reactions are considered. In addition, provision has been made for both reversible and irreversible reactions. Each reaction is assumed to involve up to a maximum of two different reactant and two different product species and thus can be written in the general form:

\[
v'_1S_1 + v'_2S_2 \rightarrow k_f v''_3S_3 + v''_4S_4,
\]

where \( v'_i \) is the stoichiometric coefficient (i.e., number of moles) of reactant species \( i \) in the reaction, \( v''_i \) is the stoichiometric coefficient of product species \( i \) in the reaction, \( S_i \) is the chemical symbol for species \( i \), and \( k_f \) and \( k_r \) are the forward and reverse rate coefficients, respectively. In equation (1), either species \( S_1 \) or \( S_4 \) or both may be either absent or the general third-body collisional partner \( M \). Therefore all collisional processes, including isomerization and spontaneous activation and deactivation of excited species, are considered. In addition, photochemical reactions of the following type are allowed:

\[
h v + v'_2S_2 \rightarrow v''_3S_3 + v''_4S_4,
\]

where \( h v \) represents a single quantum of radiation absorbed by the reactant.

All reactions are assumed to be elementary, that is, real molecular events (e.g., [4]), and so the \( \{ v'_i \} \) and \( \{ v''_i \} \) are integers. Also, all species are assumed to be ideal gases. For each reaction, irrespective of its type, the forward rate coefficient is usually given by the empirical expression [4]:

\[
k_f = AT^n \exp \left( -\frac{E}{RT} \right),
\]

where the preexponential factor \( A \), the temperature exponent \( n \) and the activation energy \( E \) are constants, \( R \) is the universal gas constant and \( T \) is the temperature. For a reversible elementary reaction, either the user may specify the reverse rate coefficient or it is computed within the code, by using the principle of detailed balancing or microscopic reversibility [4]:

\[
k_r = k_f / K_c,
\]

where \( K_c \) is the concentration equilibrium constant for the reaction [1].

### Thermodynamic and Transport Properties

The thermodynamic properties of the species are computed by using the empirical equations given by Gordon and McBride [25]. For each species \( i \) the nondimensional properties molar-specific heat at constant pressure \( (c_p_i/R) \), molar-specific enthalpy \( (h_i/RT) \) and the one-atmosphere molar-specific entropy \( (s_i^0/R) \) are specified as functions of temperature as follows:

\[
\frac{c_{pi}}{R} = a_{i,1} + a_{i,2}T + a_{i,3}T^2 + a_{i,4}T^3 + a_{i,5}T^4,
\]

\[
\frac{h_i}{RT} = a_{i,1} + \frac{a_{i,2}}{2}T + \frac{a_{i,3}}{3}T^2 + \frac{a_{i,4}}{4}T^3 + \frac{a_{i,5}}{5}T^4 + a_{i,6},
\]

\[
\frac{s_i^0}{R} = a_{i,1} \ln T + a_{i,2}T + \frac{a_{i,3}}{2}T^2 + \frac{a_{i,4}}{3}T^3 + \frac{a_{i,5}}{4}T^4 + a_{i,7},
\]

where \( a_{i,1} - a_{i,7} \) are least-squares coefficients. For each species, two sets of coefficients for use on two adjacent temperature intervals, 300 to 1000 K and 1000 to 5000 K, are included. The data are constrained to give the same results at 1000 K. The thermodynamic properties of the reacting gas mixture are evaluated by applying the Gibbs theorem [27]: that is, by simply summing the contributions made by each species.
The transport properties of the species are computed by using the empirical equations given by Svehla [28]. For each species i, the dynamic viscosity \( \mu_i \) and thermal conductivity \( \kappa_i \) are specified as functions of temperature, in the form used by Maitland and Smith [29]:

\[
\ln \mu_i = a_{i,\mu} \ln T + \frac{b_{i,\mu}}{T} + \frac{c_{i,\mu}}{T^2} + d_{i,\mu}, \tag{8}
\]

\[
\ln \kappa_i = a_{i,\kappa} \ln T + \frac{b_{i,\kappa}}{T} + \frac{c_{i,\kappa}}{T^2} + d_{i,\kappa}. \tag{9}
\]

The coefficients in these equations are obtained by least-squares fitting. The gas mixture dynamic viscosity \( \mu \) and thermal conductivity \( \kappa \) are computed using formulae suggested by Wilke [30,31] and Lindsay and Bromley [32].

**Heat Transfer Models**

The heat transfer rate between a reacting system and its surroundings is, in general, a function of the reacting gas and ambient temperatures, as well as flow rate and geometry. It is most likely that exact heat exchange rates will not be known when modeling an experimental reacting system. Therefore the main usefulness of the code will be in determining the effects of various assumed heat transfer rates. The default method is to prescribe the heat loss rate, \( Q \). (or heat loss rate per unit length, \( Q' \)) as a polynomial function of the reacting mixture temperature:

\[
Q \text{ (or } Q') = H_0 + H_1 T + H_2 T^2 + H_3 T^3 + H_4 T^4, \tag{10}
\]

where the \( \{H_k\} \) are user-specified constants. For one-dimensional flow problems, the code also contains built-in procedures for computing \( Q' \) by using standard correlations for the heat transfer coefficient for laminar and turbulent pipe flows (e.g., [33]).

**Chemical Kinetics Problems**

To describe the temporal or spatial evolution of homogeneous chemical reaction systems, equations are needed for species concentrations, temperature, density, pressure, and possibly, velocity [4]. Mathematical descriptions of combustion chemistry. LSENS includes a variable-step, variable-order implicit Adams method and a variable-step, variable-order backward differentiation formula (BDF) method. Both methods are step-by-step methods, and advance the numerical solution at each step \([\xi_{m-1}, \xi_m]\) by using linear multistep formulae of the type

\[
\sum_{j=1}^{K_1} \alpha_j Y_{n-j} + h_n \sum_{j=0}^{K_2} \beta_j L_{n-j} \tag{12}
\]

where \( q \) and \( K_2 \) are associated with a particular integration method, \( h_n (= \xi_m - \xi_{m-1}) \) is the stepsize for the current step, \( \Sigma_{n-j} \) is the numerical solution vector at \( \xi_{n-j} \), and \( L_{n-j} \) \( (= f(\Sigma_{n-j}) \) is the approximation to the exact derivative vector \( \Sigma_{n-j} \) at \( \xi_{n-j} \).

Extensive experimentation by Radhakrishnan [1] showed that the BDF method is consistently superior to the Adams method for chemical kinetics applications, and is therefore the default method in LSENS. For the BDF method, \( K_1 = q \) and \( K_2 = 0 \), and equation (12) reduces to

\[
Y_n = \sum_{j=1}^{q} \alpha_j Y_{n-j} + h_n \beta_0 L_n, \tag{13}
\]

where \( q \) is the method order for the step.

LSODE uses a predictor-corrector scheme, wherein at each step \([\xi_{n-1}, \xi_n]\) an initial guess \( y_n(0) \) for the solution vector at \( \xi_n \) is first produced, and then the guess is improved upon by iteration. That is, improved estimates \( y_n(m) \) \( (m = 1,2,\ldots) \) are computed until the iteration converges. LSODE includes a range of iteration techniques, which can be generalized by the recursive relation [21]:
where the $N \times N$ iteration matrix $P$ depends on the iteration technique. Radhakrishnan [1] examined the different iteration techniques built into LSODE for chemical kinetics applications, and the most efficient and accurate technique was Newton-Raphson iteration, for which

$$P = I - h_n \beta J.$$  

where $I$ is the $N \times N$ identity matrix, and the $N \times N$ matrix $J$ is the Jacobian matrix, with element $J_{ij}$ defined as

$$J_{ij} = \frac{\partial f_i}{\partial Y_j}, \hspace{1cm} i,j = 1, \ldots, N,$$

where $f_i$ is the $i$th ODE for the chemistry also solves for the sensitivity coefficients. The rationale for using the same solver is that the two model equations. The same algorithm that solves the sensitivity equations separately from, but sequentially with, the model problem is advanced. Then the new solution $\xi_2$ is generated by equation (19) to advance the $\{S_i\}$ by the same step. Of equation (14). At each step $[\xi_{n-1}, \xi_n]$ the solution for the model problem is sufficiently accurate, to guarantee accuracy of the computed sensitivities.

A useful feature of LSODE is that it will estimate the elements of the Jacobian matrix by finite-difference approximations, if the user chooses not to provide analytical expressions for them. However, this method requires $(N + 1)$ derivative evaluations for a system of $N$ ODEs and so, as shown by Radhakrishnan [1], is much more expensive than the use of an analytical Jacobian, especially for large $N$. Therefore, LSENS includes analytical Jacobians for all problem types, although the user has the option of using the numerical Jacobians generated by LSODE.

**Sensitivity Analysis**

For any static reaction problem the first-order sensitivity coefficients $\{S_{ij} = \partial Y_i/\partial \eta_j\}$ can be computed. Here $Y_i$ is the numerical solution for the $i$th ($i = 1, \ldots, N$) dependent variable and $\eta_j$ is either an initial condition value or a rate coefficient parameter (i.e., $A$, $n$, $E$; see equation (3)). The ODEs for the sensitivity coefficients are obtained by differentiating equation (11) with respect to $\eta_j$ and then interchanging the order of differentiation with respect to $\xi$ and $\eta_j$. The result is

$$\frac{dS_j}{d\xi} = \xi S_j + \frac{dJ}{d\eta_j} = J S_j + \frac{\partial f}{\partial \eta_j}$$

where $J$ is the Jacobian matrix (eq. (16)) and $S_j = \partial Y / \partial \eta_j$.

The sensitivity analysis computations use the decoupled direct method (DDM) [1,12,23,24]. The DDM solves the sensitivity equations separately from, but sequentially with, the model equations. The same algorithm that solves the ODEs for the chemistry also solves for the sensitivity coefficients. The rationale for using the same solver is that the two systems of ODEs have the same Jacobian. If the BDF method is used to solve equation (17), the resulting formula is

$$P S_{n} = \sum_{k=1}^{q} \alpha_j S_{n-k} + h_n \beta_0 s_n.$$

By using equation (17), equation (18) can be rewritten as

$$P S_{n} = \sum_{k=1}^{q} \alpha_j S_{n-k} + h_n \beta_0 s_n.$$
Chemical Equilibrium

The code has built-in procedures for computing the equilibrium composition for the following four assigned states: (1) pressure and temperature; (2) pressure and mixture mass-specific enthalpy; (3) specific volume and temperature; and (4) specific volume and mixture mass-specific internal energy. For cases 2 and 4 the equilibrium temperature is also determined. The code automatically performs the appropriate type of equilibrium calculation, depending on the kinetics problem being solved.

The calculation procedures for the equilibrium states were adapted from the code CEA [25]. The equilibrium state is obtained by minimizing either the Gibbs or Helmholtz function. In either case, the chemical equilibrium criterion that must be satisfied by the system is given by [35]

\[ \sum_{j=1}^{NS} \mu_j \sigma_j = 0, \tag{20} \]

where NS is the total number of species in the reacting gas mixture and \( \mu_j \) is here the chemical potential of species j. This minimization is subject to several constraints imposed by the conservation of atomic species (all cases), the assigned enthalpy (case 2), and assigned internal energy (case 4).

The resulting algebraic equations are solved by using a descent Newton-Raphson iteration method, which automatically limits the size of the corrections at each iteration to avoid convergence difficulties. Also, to prevent negative concentrations and temperature, the code solves for the logarithm of the variables.

Incident Shock

LSENS includes an option to compute the thermodynamic state and velocity behind an incident shock. Two types of computations are performed. First, the code solves for the "equilibrium" shock conditions, that is, after the shock initiated reactions have equilibrated. The second calculation produces the "frozen" shock conditions immediately after shock passage, when the composition is unchanged from its initial value. In both cases, the post-shock conditions are obtained by solving the mass, momentum and energy conservation equations describing steady, inviscid flow of an ideal gas:

\[ \rho_1 V_1 = \rho_2 V_2, \tag{21} \]

\[ p_1 + p_1 V_1^2 = p_2 + p_2 V_2^2, \tag{22} \]

\[ h_1 + \frac{V_1^2}{2} = h_2 + \frac{V_2^2}{2}, \tag{23} \]

In these equations, which assume that the coordinate system is attached to the shock, \( \rho, V, p, \) and \( h \) are respectively the mixture density, velocity, pressure and mass-specific enthalpy. The subscripts 1 and 2 indicate, respectively, conditions upstream and downstream of the shock.

The calculation procedure for both the frozen and equilibrium states were adapted from the NASA code CEA [25]. Essentially, starting with a guess for the post-shock state, a Newton-Raphson iteration procedure, which automatically limits the size of the corrections to minimize convergence difficulties, is used. To avoid negative variables during the solution procedure, the equations are cast in terms of the logarithm of the variables.

Post-Shock Kinetics Problem: Starting with the frozen shock state, LSENS follows the progress of the chemical reaction in the shocked gas by integrating the ODEs describing one-dimensional flow with assigned area, over a prescribed time or distance interval. The flow area profile is given by a special function, which corrects for frictional losses: corrections are included for both laminar and turbulent boundary layers [2,36-38].

Perfectly Stirred Reactor (PSR)

Steady state PSR computations can be performed for either a specified mass flow rate or a specified reactor temperature. In the former case, the code solves for the mixture composition and temperature at reactor exit. In the latter case, the mass flow rate and reactor exit mixture composition are computed. The problem type is identified by examining the input parameters required for problem solution, and so a separate switch need not be set. For steady operation of the PSR and both problem types outlined above, the governing algebraic equations are obtained from the conservation equations for species concentrations and energy:

\[ \dot{m}(\sigma_i - \sigma_i^*) = W_i V, \quad i = 1, \ldots, NRS, \tag{24} \]

\[ \dot{m}(h - h^*) = -Q. \tag{25} \]

In equations (24) and (25), \( \dot{m} \) is the mass flow rate through the reactor, \( W_i \) is the molar rate of production of species \( i \) per unit volume, \( V \) is the reactor volume, NRS is the total number of reacting species, \( Q \) is the heat loss rate from the reactor, and the superscript * denotes inlet conditions. The heat loss rate can be specified as a polynomial function of up to fourth degree in reactor temperature (see equation (10)).

To solve the system of algebraic equations, LSENS uses essentially the same Newton-Raphson iteration technique that is built into CEA [25] for computing the equilibrium state. In particular, this method automatically limits the size of the corrections, to reduce convergence difficulties. Also, to avoid negative results, the code solves for the logarithm of
the variables. Starting with conditions close to the equilib-
rium state, a series of perfectly stirred reactor computa-
tions is performed until the desired mass flow rate or reactor
temperature is reached. This technique is used to minimize the
possibility of convergence to a false solution—one that is
mathematically correct but physically unrealistic. The rou-
tine includes several tests and, when necessary, restarts of
the calculation to ensure that the solution is physically mean-
ingful. Finally, it includes tests for possible blowout of the
chemical reaction within the reactor.

**Combined PSR and plug flow problem:** A plug flow calcula-
tion may be performed after a PSR problem in a single com-
puter run. Such a two step process is sometimes used as a
simplified model of a gas turbine combustor. To solve this
type of problem, the code first performs a PSR calculation
using the necessary input information for this problem type,
and then solves the ODEs governing one-dimensional invisi-
cid flow. For the flow problem, the code uses the PSR solu-
tion as the initial (i.e., inflow) conditions.

**Error Messages**

LSENS contains many error messages—every input
parameter is tested for legality and consistency with the
other input variables. For example, the code includes an
option to check the legality of the reaction mechanism. The
code verifies that no reaction is duplicated and that each
reaction satisfies charge and atom balance requirements.
Reaction duplication may arise because the same reaction is
written in different forms in different regions of the input
file.

If illegal input is discovered, a detailed message is
printed. Each error message is self-explanatory and com-
plete. It not only describes the mistake, but also tells the user
how to fix the problem. During execution, some tests are
made to ensure that variables are within either given or rea-
sonable bounds. Any difficulty encountered during execution
will result in an error exit. A message giving the reason for
the termination and the name of the subprogram where the
problem occurred will also be printed. If the computation
stops prematurely, the user should look for the error message
near the end of the output file.

**Code Usage**

Two input data files are normally required to execute the
code. The first one, referred to as the “Standard Thermody-
amic Data File,” must contain the chemical symbol and
composition, molar mass and thermodynamic data for each
species. The second file, the “Problem Data File,” must give
information about, and data required by, the problem(s) to be
solved. A third input file, the “Transport Properties Data
File,” which contains transport property data, is required for
certain computations. Standard Thermodynamic and Trans-
port Properties Data Files, containing data from the most
recent thermodynamic data base of the CEA code [25] for
many species in the C-H-N-O system, are supplied with
LSENS. However, the user has the option of specifying ther-
donamic data for some (or all) species, by including them
in the Problem Data File. Thus the user has a convenient way
of temporarily changing any thermodynamic data or adding
new species without modifying the Standard File. However,
the same thermodynamic data, that is, those given in the
Standard Thermodynamic Data File and/or the Problem Data
File, are used for every problem included in the Problem
Data File.

The Problem Data File must specify the problem type
and provide the data needed to solve the problem. This infor-
mation includes the chemical reaction mechanism, including
rate coefficient parameters, third-body collisional efficien-
cies and inert species, if any. The user must also provide
details of the problem, including: the independent variable
for a flow problem; assigned-variable profile; temperature
data, if a temperature profile is to be specified; heat transfer
data, heat transfer is to be calculated; optional information
such as the input units and output units (e.g., SI, or US cus-
tomary); and output controls. For a perfectly stirred reactor
problem, the desired reactor conditions should be specified,
and data given for the variables that tell the code when inter-
mediate solutions are needed. The above information must
be followed by the initial (or inlet) conditions, including
mixture composition. For static and flow problems, the
required integration controls must also be specified. Finally,
if sensitivity analysis is required, the Problem Data File must
include lists of the parameters with respect to which sensitiv-
ity coefficients are to be computed and of the dependent vari-
ables of interest.

If multiple problems are to be solved during a single exe-
cution of the code, the required data for the second and sub-
sequent cases depend on the information, if any, that can be
used from the previous case(s). The code was designed to
minimize the amount of input data, especially when they are
the same for two or more consecutive cases. Thus, for exam-
ple, the user need not provide the reaction mechanism when
it either is identical to that used on the previous case or can
be obtained from it by adding reactions or by modifying rate
coefficient parameters. Other data for the problem may either
be included for every new case or, in most cases, the data
most recently specified will be used.

**Illustrative Test Problems**

In addition to the Thermodynamic and Transport Propert-
ies Files, two Problem Data Files are provided with the
code. These files serve to illustrate the problem types that
can be solved by LSENS and the options built into it. The
data files also help elucidate the construction of the problem
data file required to execute LSENS. To demonstrate the
capability of solving multiple problems in a single run, both
problem files contain several test cases, as discussed below.
The first data file contains 16 kinetics-only (i.e., no sensitivity analysis) test cases, which are summarized in Table 1. In this table NR is the total number of elementary reactions in the reaction mechanism, NS is the total number of species, and N is the total number of ODEs solved. It is clear from this table that the test cases cover a variety of reaction models and problem types. The problem size varies from a simple case involving one reaction among three species (case 1) to one involving 42 species participating in 143 reactions (case 11). On a Silicon Graphics Indigo 2 computer with the R4000 processor chip, LSENS required approximately 14 s CPU time to solve all 16 problems. The execution time alone (i.e., not including the CPU time for preprocessing and input/output) was less than 10 s. On a Pentium II the total CPU time (i.e., including preprocessing and input/output) for the 16 cases was comparable (approximately 9 s). Additional run time information and detailed comparisons with other methods and codes is available in Radhakrishnan [1].

The second data file contains nine sensitivity analysis test cases, which are summarized in Table 2. Here, NS is the total number of sensitivity parameters (i.e., initial conditions and rate coefficient parameters). The first six cases are isothermal problems, for which sensitivity analysis results have been published in the literature. Results for cases 1 and 2 were also computed analytically. Excellent agreement was obtained for all six cases between the LSENS and literature results [1]. The last three cases are nonisothermal problems, which illustrate the application of sensitivity analysis to combustion kinetics. Case 7 describes a simple nonisothermal problem for which the analytical solution is known [1,24]. The results produced by LSENS showed excellent agreement with the analytical solution for the model problem and for the sensitivity coefficients with respect to all four initial conditions and the three rate coefficient parameters of the reaction. For cases 8 and 9, sensitivity coefficients were generated by using finite difference approximations; agreement with LSENS was again excellent [1].

On a Silicon Graphics Indigo 2 computer with the R4000 processor chip, LSENS required approximately 10 s CPU time for all nine problems. The execution time alone (i.e., not including the CPU time for preprocessing and input/output) was approximately 8 s. On the Pentium II the total CPU time (i.e., including preprocessing and input/output) for the nine cases was approximately 7 s. As for the kinetics-only test cases, the sensitivity analysis method built into LSENS has been compared with other methods and computer codes [1,12]. For example, case 9, which describes the ignition and subsequent combustion of a benzene-oxygen-argon mixture, consists of 120 reversible reactions among 39 reacting species and the inert species argon. To solve for the dependent variables required 6.3 s of execution time on the Amdahl computer, using the UTS operating system and the Fujitsu 77 compiler. To solve for the dependent variables and sensitivity coefficients with respect to all 42 initial conditions (40 species mass-specific mole numbers, density, and temperature) and all 360 rate coefficient parameters required 246 s. This execution time may be compared to the more than 2500 s that would be required by a finite difference method wherein parameters are varied one at a time.

Concluding Remarks

The NASA Lewis kinetics and sensitivity analysis code, LSENS, which has been developed for complex, homogeneous, gas-phase reactions, was described. The code has been designed for a variety of reaction models: static system; steady, one-dimensional, inviscid flow; reaction initiated by an incident shock; and a perfectly stirred reactor. The different capabilities and computational procedures built into the code and its usage were briefly described. Standard thermodynamic and transport properties data are provided with the code, as are two problem data files that illustrate the different options built into LSENS.

Details regarding code availability and procurement can be obtained from the National Technology Transfer Center, Wheeling Jesuit University, 316 Washington Avenue, Wheeling, WV 26003. Telephone: (800) 678-6882. (NTTC supersedes COSMIC, University of Georgia, in this activity.)

References


### TABLE 1. Description of kinetics-only test problems supplied with LSENS

<table>
<thead>
<tr>
<th>Test problem</th>
<th>Description of problem</th>
<th>Problem parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Dissociation of bromine in a shock tube.</td>
<td>NR 1 NS 3 N 5</td>
</tr>
<tr>
<td>2</td>
<td>Hydrogen-air combustion in supersonic flow with heat transfer.</td>
<td>36 18 19</td>
</tr>
<tr>
<td>3</td>
<td>Same problem as case 2, but with a larger reaction mechanism.</td>
<td>37 18 19</td>
</tr>
<tr>
<td>4</td>
<td>Hydrogen-oxygen combustion in subsonic flow with heat transfer.</td>
<td>18 8 11</td>
</tr>
<tr>
<td>5</td>
<td>Static, rich methane-air combustion at assigned pressure with heat transfer.</td>
<td>133 39 41</td>
</tr>
<tr>
<td>6</td>
<td>Lean methane-air adiabatic combustion in supersonic flow at constant pressure.</td>
<td>133 39 41</td>
</tr>
<tr>
<td>7</td>
<td>Same problem as case 6, except that the area obtained from case 6 is assigned as input.</td>
<td>133 39 42</td>
</tr>
<tr>
<td>8</td>
<td>Same problem as case 6, except that the area and temperature obtained from case 6 are assigned as input.</td>
<td>133 39 41</td>
</tr>
<tr>
<td>9</td>
<td>Constant-density, static methanol-air adiabatic combustion.</td>
<td>133 39 40</td>
</tr>
<tr>
<td>10</td>
<td>Constant-temperature, constant-pressure, methane-air reaction in supersonic flow.</td>
<td>143 42 42</td>
</tr>
<tr>
<td>11</td>
<td>Perfectly stirred reactor combustion of a rich propane-air mixture, followed by supersonic expansion of the combustion products through a diverging nozzle with heat loss.</td>
<td>136 42 44</td>
</tr>
<tr>
<td>12</td>
<td>High-temperature, adiabatic air ionization reaction in constant-area subsonic flow.</td>
<td>12 12 15</td>
</tr>
<tr>
<td>13</td>
<td>High-temperature, high-pressure, static reaction of carbon monoxide-hydrogen mixture at constant density and constant temperature.</td>
<td>23 13 13</td>
</tr>
<tr>
<td>14</td>
<td>Constant-density, adiabatic, static problem involving photolytic ignition of hydrogen-oxygen mixture at low initial temperature.</td>
<td>20 8 9</td>
</tr>
<tr>
<td>15</td>
<td>Methane-air combustion in supersonic flow, with assigned duct area and output required at axial locations corresponding to specified values of duct area.</td>
<td>133 39 42</td>
</tr>
<tr>
<td>16</td>
<td>Same problem as case 15, except here output is required at specified axial positions.</td>
<td>133 39 42</td>
</tr>
</tbody>
</table>

### TABLE 2. Description of sensitivity test problems supplied with LSENS

<table>
<thead>
<tr>
<th>Test problem</th>
<th>Description of problem</th>
<th>Problem parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Constant-volume, isothermal reaction, ( A = B ).</td>
<td>NR 2 NS 2 N 4</td>
</tr>
<tr>
<td>2</td>
<td>Constant-volume, isothermal reaction, ( A = B = C ).</td>
<td>4 3 3 4</td>
</tr>
<tr>
<td>3</td>
<td>Constant-volume, isothermal pyrolysis of ethane, using a simplified mechanism.</td>
<td>5 7 8 4</td>
</tr>
<tr>
<td>4</td>
<td>Constant-volume, isothermal reaction of a methane-oxygen-argon mixture containing trace concentrations of carbon dioxide and hydrogen.</td>
<td>44 14 13 44</td>
</tr>
<tr>
<td>5</td>
<td>Constant-volume, isothermal oxidation of a formaldehyde-carbon monoxide mixture.</td>
<td>25 15 14 25</td>
</tr>
<tr>
<td>6</td>
<td>Constant-volume, isothermal reaction of a wet carbon monoxide-oxygen-nitrogen mixture.</td>
<td>52 12 11 58</td>
</tr>
<tr>
<td>7</td>
<td>Adiabatic, constant-pressure isomerization reaction with simplified rate coefficient expression that permits analytical solution.</td>
<td>1 2 4 7</td>
</tr>
<tr>
<td>8</td>
<td>Constant-pressure, adiabatic, static ignition of a stoichiometric hydrogen-air mixture, seeded with 0.45% nitric oxide.</td>
<td>40 19 20 20</td>
</tr>
<tr>
<td>9</td>
<td>Adiabatic, constant-density, static combustion of a shock-heated, near-stoichiometric benzene-oxygen-argon mixture.</td>
<td>120 40 42 51</td>
</tr>
</tbody>
</table>
Figure 1. Flowchart of MAIN Program
**Title and Subtitle**

LSENS, The NASA Lewis Kinetics and Sensitivity Analysis Code

**Author(s)**

K. Radhakrishnan

**Performing Organization Name(s) and Address(es)**

Institute for Computational Mechanics in Propulsion
22800 Cedar Point Road
Cleveland, Ohio 44142

**Sponsoring/monitoring Agency Name(s) and Address(es)**

National Aeronautics and Space Administration
John H. Glenn Research Center at Lewis Field
Cleveland, Ohio 44135–3191

**Abstract**

A general chemical kinetics and sensitivity analysis code for complex, homogeneous, gas-phase reactions is described. The main features of the code, LSENS (the NASA Lewis kinetics and sensitivity analysis code), are its flexibility, efficiency and convenience in treating many different chemical reaction models. The models include: static system; steady, one-dimensional, inviscid flow; incident-shock initiated reaction in a shock tube; and a perfectly stirred reactor. In addition, equilibrium computations can be performed for several assigned states. An implicit numerical integration method (LSODE, the Livermore Solver for Ordinary Differential Equations), which works efficiently for the extremes of very fast and very slow reactions, is used to solve the “stiff” ordinary differential equation systems that arise in chemical kinetics. For static reactions, the code uses the decoupled direct method to calculate sensitivity coefficients of the dependent variables and their temporal derivatives with respect to the initial values of dependent variables and/or the rate coefficient parameters. Solution methods for the equilibrium and post-shock conditions and for perfectly stirred reactor problems are either adapted from or based on the procedures built into the NASA code CEA (Chemical Equilibrium and Applications).