

**NASA
Reference
Publication
1329**

1994

**LSENS, A General Chemical
Kinetics and Sensitivity
Analysis Code for
Homogeneous Gas-Phase
Reactions**

II. Code Description and Usage

Krishnan Radhakrishnan
NYMA, Inc.
Lewis Research Center Group
Brook Park, Ohio

David A. Bittker
Lewis Research Center
Cleveland, Ohio



National Aeronautics and
Space Administration

Office of Management

Scientific and Technical
Information Program



Preface

LSENS, the Lewis General Chemical Kinetics and Sensitivity Analysis Code, has been developed for homogeneous, gas-phase chemical kinetics computations and contains sensitivity analysis for a variety of problems, including nonisothermal situations. The code is described in a series of three reference publications, which also provide a detailed guide to its use and many illustrative test problems.

LSENS has been designed for accuracy, efficiency, flexibility, and convenience. A variety of chemical reaction models can be considered: static system; steady, one-dimensional, inviscid flow; reaction behind an incident shock wave, including boundary layer correction; and perfectly stirred (highly backmixed) reactor. In addition, the chemical equilibrium state can be computed for the assigned states of temperature and pressure, enthalpy and pressure, temperature and volume, and internal energy and volume. Any reaction problem can be adiabatic, have an assigned heat transfer profile, or for static and flow problems, have an assigned-temperature profile. For static problems either the density is constant or the pressure-versus-time profile is assigned. For flow problems either the pressure or area can be assigned as a function of time or distance. For a static reaction sensitivity coefficients of the dependent variables and their temporal derivatives with respect to the initial values of the dependent variables and/or the three rate coefficient parameters of the chemical reactions can be obtained.

LSENS checks the legality and sufficiency of all input. At the user's option LSENS checks the reaction mechanism for uniqueness and ensures that each reaction satisfies charge and atom balance requirements.

Part I of the series (NASA RP-1328), consisting of chapters 1 to 7, presents the theory and the numerical solution procedures used in LSENS. The ordinary differential equations (ODE's) describing chemical kinetics problems are derived in chapter 2. Chapter 3 describes the numerical integration method and how it is implemented. In chapter 4 the governing ODE's for sensitivity analysis are derived and the solution method and numerical algorithm explained. The governing equations and solution methods for the chemical equilibrium state, equilibrium and frozen thermodynamic states behind an incident shock wave, and perfectly stirred reactor problems are presented in chapters 5 to 7.

This volume, part II of the series (chapters 8 to 13 and appendixes A to C), describes LSENS, its usage, and how to modify it. Chapter 8 describes the computational capabilities and convenience features built into the code. Chapter 9 presents its structure and description. Chapter 10 lists modifications that may be required to implement LSENS on the user's computer system. Chapter 11 provides a guide to code usage and describes how to prepare the input data files required to execute LSENS. The output information generated by the code is discussed in chapter 12. Example problems illustrating both problem data file construction and code usage are given in chapter 13. These examples supplement chapter 11 by providing additional guidance on preparation of the problem data file.

Preface

The partial derivatives required by the numerical solution procedures detailed in chapters 3, 4, and 7 are derived in appendix A. Appendix B shows how to access the system clock for several common computing systems so that execution times can be measured. Appendix C describes the modifications required to change the built-in values for various quantities.

Part III of the series (NASA RP-1330), consisting of appendixes D and E, explains the example problems provided with LSENS and presents sample results. Appendix D describes the kinetics test cases. These problems illustrate the various reaction models that can be solved by, and options built into, LSENS. Appendix E describes the kinetics-plus-sensitivity-analysis test cases supplied with the code. The examples in the two appendixes cover a variety of problem types and so should serve as useful models for the structure of the problem data file required to execute the code. Indeed, it is likely that the desired file can be produced by modifying one of the test cases.

Details regarding code availability and procurement can be obtained from COSMIC, 328 East Broad Street, University of Georgia, Athens, GA 30602 (Telephone: 706-542-3265).

Contents

List of Figures	ix
List of Tables	xi
Symbols	xv
Chapter 8. Capabilities of Code	1
8.1 Types of Chemical Reaction and Rate Coefficient	1
8.2 Assigned Variable and Specification	2
8.3 Assigned-Temperature Problem	2
8.4 Time and Distance Computations for Flow Problem	3
8.5 Heat Transfer Models	4
8.6 Thermodynamic Properties	5
8.7 Transport Properties	6
8.8 Simplified Composition Input	6
8.9 Net Reaction and Species Formation Rates	7
8.10 Equilibration Factor	8
Chapter 9. Code Organization and Description	9
9.1 Introduction	9
9.2 Control	19
9.3 Input/Output	21
9.4 Numerical Integration Procedure	24
9.5 Sensitivity Analysis	29
9.6 Chemical Equilibrium Computations	33
9.7 Incident Shock Computations	35
9.8 Perfectly Stirred Reactor Computations	37
9.9 Error Messages	37
Chapter 10. Code Implementation	39
10.1 Computer Language	39
10.2 Suppressing Underflow Error Messages	39
10.3 Function SECCPU	39
10.4 Modifying Subroutine XERRWV	39
10.5 BLOCK DATA Variables	39
10.6 Reducing Storage Requirement	40
10.7 Code Limitations	40
10.8 Compiler Optimization Level	40

Contents	
Chapter 11. Description of Code Usage	41
11.1 Thermodynamic Data Information	41
11.2 Transport Properties Data	43
11.3 Problem Data File for Single Case	43
11.3.1 Title Line	44
11.3.2 Chemical Reactions and Collisional Efficiencies	44
11.3.3 Inert Species List	47
11.3.4 Integration and Assigned Variables, Units, and Fuel Name Line	47
11.3.5 Problem Data, Related Options, Heat Transfer Data, and Output Controls (Namelist PROB)	47
11.3.5.1 Problem types and related options	49
11.3.5.1.1 Static problem	49
11.3.5.1.2 One-dimensional flow problem	49
11.3.5.1.3 Constant-temperature problem	52
11.3.5.1.4 Assigned-temperature problem	52
11.3.5.1.5 Sensitivity analysis	52
11.3.5.1.6 Chemical equilibrium computations	52
11.3.5.1.7 Shock tube reaction	53
11.3.5.1.8 Perfectly stirred reactor (PSR) problem	53
11.3.5.1.9 Combined PSR-and-plug-flow problem	53
11.3.5.1.10 Summary	53
11.3.5.2 Heat transfer models	53
11.3.5.3 Reaction mechanism tests	55
11.3.5.4 Print stations	55
11.3.5.5 Output information specification	56
11.3.6 Assigned-Temperature Profile (Namelist TMPDAT)	57
11.3.7 Perfectly Stirred Reactor Problem Data (Namelist WSPROB)	58
11.3.8 Initial Conditions (and Namelist START)	60
11.3.8.1 Static and flow kinetics problems	62
11.3.8.2 Incident shock problem	62
11.3.8.3 PSR problem	62
11.3.8.4 PSR-plus-plug-flow problem	62
11.3.8.5 Summary	63
11.3.8.6 Initial mixture composition	63
11.3.9 Local Error Tolerances and Other Integration Controls (Namelist SOLVER)	63
11.3.10 Sensitivity Analysis Data (and Namelist SENRXN)	66
11.3.11 FINIS Line	67
11.4 Problem Data File for Multiple Cases	67
11.4.1 Title Line	69
11.4.2 ACTION Line	69
11.4.3 Integration and Assigned Variables, Units, and Fuel Name Line	70
11.4.4 Problem Data, Related Options, Heat Transfer Data, and Output Controls	70
11.4.5 Assigned-Temperature Profile	70
11.4.6 Perfectly Stirred Reactor Problem Data	70
11.4.7 Initial Conditions	70
11.4.8 Local Error Tolerances and Other Integration Controls	70
11.4.9 Sensitivity Analysis Data	71
11.4.10 FINIS Line	71
Chapter 12. Description of Output	73
12.1 Introduction	73
12.2 Static and Flow Kinetics Problems	73
12.2.1 Standard Output	73

Contents	
Assigned area and temperature	116
Sensitivity Analysis	117
Assigned-Pressure Static Problem	117
Constant-Density Static Problem	117
Perfectly Stirred Reactor Problem	117
Assigned-Mass-Flow-Rate Problem	118
Assigned-Temperature Problem	118
Appendix B. Accessing System Clock	119
Appendix C. Code Modifications	121
References	129

List of Figures

Figure 9.1. —Structure of LSENS code	20
Figure 9.2. —Flowchart of MAIN program	21
Figure 9.3. —Structure of subroutine KINP	22
Figure 9.4. —Flowchart of subroutine KINP	23
Figure 9.5. —Structure of LSODE package	25
Figure 9.6. —Flowchart of subroutine LSODE, as used by LSENS	26
Figure 9.7. —Flowchart of subroutine STODE, as used by LSENS	27
Figure 9.8. —Flowchart of subroutine DIFFUN	28
Figure 9.9. —Flowchart of subroutine PEDERV	29
Figure 9.10. —Flowchart of subroutine SENDDM	30
Figure 9.11. —Structure of subroutine DMSTOD	31
Figure 9.12. —Flowchart of subroutine DMSTOD	32
Figure 9.13. —Structure of chemical equilibrium section	33
Figure 9.14. —Flowchart of equilibrium calculation procedure (subroutine EQLBRM)	34
Figure 9.15. —Structure of incident shock section	35
Figure 9.16. —Flowchart of incident shock calculation procedure (subroutine SHOCKS)	36
Figure 9.17. —Structure of perfectly stirred reactor section	37
Figure 9.18. —Flowchart of perfectly stirred reactor calculation procedure (subroutine WSR)	38

List of Figures

Figure 11.1.—Illustration of structure of standard thermodynamic data file 42

Figure 11.2.—Transport properties data file supplied with LSENS 43

Figure 11.3.—Structure of problem data file for single (or first) case 45

Figure 11.4.—Structure of problem data file for multiple cases 68

Figure B.1.—Listing of function SECCPU for IBM 370 computer 119

Figure B.2.—Listing of function SECCPU for Amdahl 5870 computer and
VM operating system 119

Figure B.3.—Listing of function SECCPU for Amdahl 5870 computer,
UTS operating system, and Fujitsu 77 compiler 119

Figure B.4.—Listing of function SECCPU for VAX computers 120

Figure B.5.—Listing of function SECCPU for CDC Cyber and
Cray computers 120

Figure B.6.—Listing of function SECCPU for Convex C220 minicomputer,
Alliant FX/S computer, and IRIS workstations 120

Figure B.7.—Listing of function SECCPU for Sun SPARCstation 1 120

Figure B.8.—Listing of function SECCPU for IBM RISC System/6000 120

Figure B.9.—Listing of dummy function SECCPU that can be used if call to
system clock is not available or known 120

List of Tables

Table 8.1. —Standard oxidant composition built into LSENS	7
Table 9.1. —Description of subprograms used in LSENS	10
Table 9.2. —Routines with common blocks, subprograms, and calling subprograms	12
Table 9.3. —Common blocks with variables and subprograms	16
Table 9.4. —Summary of integration methods included in LSODE and corresponding values of METH	24
Table 9.5. —Corrector iteration techniques available in LSODE and corresponding values of MITER	24
Table 11.1. —Format of reaction line	46
Table 11.2. —Format of third-body collisional efficiency line	47
Table 11.3. —Units for input variables in the three systems allowed by code	48
Table 11.4. —Description of integration and assigned variables, units, and fuel name line	48
Table 11.5. —Description of namelist PROB	50
Table 11.6. —Variables that must be set in namelist PROB for different problem types and related options built into code	54
Table 11.7. —Variables that must be set in namelist PROB for heat transfer computations	55
Table 11.8. —Options that specify when output is to be generated and variables that must be set in namelist PROB for each option	56
Table 11.9. —Description of namelist TMPDAT	57
Table 11.10. —Variables that must be set in namelist TMPDAT for assigned- temperature problem	58

List of Tables

Table 11.11. —Description of namelist WSPROB	59
Table 11.12. —Variables that must be set in namelist WSPROB for perfectly stirred reactor problem	60
Table 11.13. —Description of namelist START	61
Table 11.14. —Variables that must be set in namelist START if simplified-composition input option is used to specify initial mixture composition	62
Table 11.15. —Summary of thermodynamic and fluid dynamic variables whose initial values must be set in namelist START	64
Table 11.16. —Description of namelist SOLVER	65
Table 11.17. —Description of namelist SENRXN	67
Table 12.1. —Units for output variables in the three systems allowed by code	75
Table 12.2. —Summary of output	79
Table 13.1. —Problem data file for propane-air, PSR-plus-flow example problem ..	82
Table 13.2. —Computed results for propane-air, PSR-plus-flow example problem ..	85
Table 13.3. —Problem data file for hydrogen-air supersonic flow example problem	87
Table 13.4. —Variation of computational work and solution with EMAX for H ₂ -air example problem	88
Table 13.5. —Variation of computational work and solution with ATOLSP for H ₂ -air example problem	89
Table 13.6. —Reactions for which sensitivity analysis is required for benzene-oxygen-argon example problem	90
Table 13.7. —Problem data file for benzene-oxygen-argon sensitivity example problem	91
Table 13.8. —Variation of computational work and kinetics solution with EMAX for benzene-oxygen-argon example problem	94
Table 13.9. —Variation of computational work and kinetics solution with ATOLSP for benzene-oxygen-argon example problem	95
Table 13.10. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial \sigma_{C_6H_6,0} \rangle$ with EMAX for benzene-oxygen-argon example problem	96
Table 13.11. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial \sigma_{OH,0} \rangle$ with EMAX for benzene-oxygen-argon example problem	96
Table 13.12. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial \sigma_{H_2O,0} \rangle$ with EMAX for benzene-oxygen-argon example problem	97

Table 13.13. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial \sigma_{Ar,0} \rangle$ with EMAX for benzene-oxygen-argon example problem	98
Table 13.14. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial T_0 \rangle$ with EMAX for benzene-oxygen-argon example problem	98
Table 13.15. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial \rho_0 \rangle$ with EMAX for benzene-oxygen-argon example problem	99
Table 13.16. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial A_1 \rangle$ with EMAX for benzene-oxygen-argon example problem	99
Table 13.17. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial A_6 \rangle$ with EMAX for benzene-oxygen-argon example problem	100
Table 13.18. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial A_8 \rangle$ with EMAX for benzene-oxygen-argon example problem	100
Table 13.19. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial A_{12} \rangle$ with EMAX for benzene-oxygen-argon example problem	101
Table 13.20. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial A_{17} \rangle$ with EMAX for benzene-oxygen-argon example problem	101
Table 13.21. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial A_{18} \rangle$ with EMAX for benzene-oxygen-argon example problem	102
Table 13.22. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial A_{104} \rangle$ with EMAX for benzene-oxygen-argon example problem	102
Table 13.23. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial n_8 \rangle$ with EMAX for benzene-oxygen-argon example problem	103
Table 13.24. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial E_8 \rangle$ with EMAX for benzene-oxygen-argon example problem	103
Table 13.25. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial \sigma_{C_6H_6,0} \rangle$ with ATOLSP for benzene-oxygen-argon example problem	104
Table 13.26. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial \sigma_{H_2O,0} \rangle$ with ATOLSP for benzene-oxygen-argon example problem	104
Table 13.27. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial T_0 \rangle$ with ATOLSP for benzene-oxygen-argon example problem	105
Table 13.28. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial A_1 \rangle$ with ATOLSP for benzene-oxygen-argon example problem	105
Table 13.29. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial n_8 \rangle$ with ATOLSP for benzene-oxygen-argon example problem	106
Table 13.30. —Variation of normalized sensitivity coefficient $\langle \partial Y_i / \partial E_8 \rangle$ with ATOLSP for benzene-oxygen-argon example problem	106

List of Tables

Table C.1.—User-adjustable variables in BLOCK DATA module	122
Table C.2.—Modifications needed to change maximum number of elements (NLMAX)	123
Table C.3.—Modifications needed to change maximum number of species (LSMAX)	124
Table C.4.—Minimum length required by real work array RWORK (i.e., minimum LRW)	125
Table C.5.—Minimum length required by integer work array IWORK (i.e., minimum LIW)	125
Table C.6.—Modifications needed to change maximum number of reactions (LRMAX)	126
Table C.7.—Modifications needed to change maximum number of third-body reactions (TBRMAX)	126
Table C.8.—Modifications needed to change maximum number of species with third-body efficiencies different from unity (TBSMAX)	127
Table C.9.—Modifications needed to change maximum number of tabular values for assigned variable (pressure or area) and temperature (NTBMAX)	127
Table C.10.—Modifications needed to change maximum number of print stations (NPMAX)	128

Symbols

A	flow cross-sectional area
A_j	preexponential factor in forward rate coefficient expression for reaction j
$(A_j)_k$	preexponential factor for third-body collisional reaction with species k as collision partner
$(A_j)_0$	preexponential factor for third-body collisional reaction with standard species as collision partner
A_r	area ratio, defined by equation (12.16)
A_s	shock tube cross-sectional area or area of curved surface of Otto-cycle engine cylinder
A_t	nozzle throat area used in rocket performance calculations, in. ²
A_v	assigned variable (pressure or area) for kinetics problem
ATOLSP	local absolute error tolerance for reacting species mole numbers, used in solving chemical kinetics ODE's
AVH	average step size used by integrator
\mathcal{A}	defined by equation (A45)
$a_{i,\kappa}, b_{i,\kappa}, c_{i,\kappa}, d_{i,\kappa}$	coefficients in equation (8.38) for thermal conductivity of species i
$a_{i,\mu}, b_{i,\mu}, c_{i,\mu}, d_{i,\mu}$	coefficients in equation (8.37) for dynamic viscosity of species i
$a_{i,1}, \dots, a_{i,7}$	coefficients in equations (8.32) to (8.34) for thermodynamic properties of species i
a_0, a_1, a_2, a_3	coefficients in cubic polynomial expression for velocity, equation (8.15)
B	cylinder bore of Otto-cycle engine

Symbols

\mathcal{B}	defined by equation (A46)
C_f	units conversion factor used in equation (A44)
C_i	molar concentration of <i>i</i> th species
C_{ij}	coefficient in equation (8.44) for Sutherland interaction constant
C_m	total molar concentration of catalyst species, equation (A16)
$C_{t,0}, C_{t,1},$ $C_{t,2}, C_{t,3}$	constants in polynomial pressure-time expression, equation (8.9)
$C_{\xi_A,0}, C_{\xi_A,1},$ $C_{\xi_A,2}, C_{\xi_A,3}$	constants in polynomial expression, equation (8.10), for assigned variable (pressure or area)
$C_{\xi_T,0}, C_{\xi_T,1},$ $C_{\xi_T,2}, C_{\xi_T,3}$	constants in polynomial expression, equation (8.13), for temperature
CPU	central processing unit (i.e., execution) time
c	sonic velocity
c^*	characteristic velocity, equation (12.17), in rocket performance calculations
c_f	thrust coefficient, equation (12.18), in rocket performance calculations
c_j	rate coefficient parameter in special rate coefficient expression, equation (8.4)
c_p	constant-pressure, mass-specific heat of mixture
$c_{p,i}$	constant-pressure, molar-specific heat of species <i>i</i>
D	diameter of flow cross section
\mathcal{D}	heat transfer function defined by equations (A36), (A41), and (A47)
d_H	shock tube hydraulic diameter
E_j	activation energy in forward rate coefficient expression for reaction <i>j</i>
$E_{q,j}$	equilibration factor for reaction <i>j</i> , equation (8.61)
EMAX	local relative error tolerance for dependent variables, used in solving chemical kinetics ODE's
EQL	logical variable defined within code and denoting type of incident shock calculation—equilibrium or frozen
\mathbf{e}	vector proportional to local truncation error vector in solution to chemical kinetics ODE's

F/O	fuel-oxygen molar ratio
FSUM	sum of squares of Newton-Raphson functionals in PSR calculation
f_i	first derivative of i th dependent variable y_i with respect to independent variable ξ
f/a	fuel-oxidant mass ratio
f_i	Newton-Raphson functional for i th dependent variable in PSR computation
$\Delta G_{T,j}^\circ$	standard-state (1 atm) Gibbs function change at temperature T for reaction j , equation (8.8)
g_c	units conversion factor used in equations (12.14) and (12.17)
g_i	molar-specific Gibbs function of species i
$\Delta H_{T,j}^\circ$	standard-state (1 atm) enthalpy change at temperature T for reaction j , equation (12.10)
$H_{T,0}, H_{T,1}, H_{T,2}, H_{T,3}, H_{T,4}$	coefficients in equation (8.21) for heat transfer rate \dot{Q} (or \dot{Q}') as polynomial function of temperature
H	integration step size
HMAX	maximum step size to be attempted by integrator on any step
HMIN	minimum step size to be attempted by integrator on any step
HT1	denotes heat transfer model 1 defined by equation (8.21)
HT2	denotes heat transfer model 2 defined by equation (8.22)
\mathcal{H}	heat transfer coefficient, equations (8.25) and (8.28)
h	mass-specific enthalpy of mixture
h_n	integration step size on step $[\xi_{n-1}, \xi_n]$
$h\nu$	single quantum of radiation
h_i	molar-specific enthalpy of species i
I_{sp}	specific impulse, equation (12.14)
I_{vac}	vacuum impulse, equation (12.15)
IALTH	integer variable defined within code and indicating how many more steps are to be taken with current step size and method order
ICALL	integer variable defined within code and indicating whether present call to integrator is first one for problem

Symbols	
ICPU	incremental CPU time since last output of results
INST	incremental number of integration steps since last output of results
ISTATE	integer variable defined within code and indicating state of integration
ITASK	integer variable defined within code and indicating task to be performed by integrator
ITER	iteration number for PSR computation
I	identity matrix
JSTART, KFLAG	integer variables defined within code and used for communication between integration subprograms
J	Jacobian matrix, used in solution of ODE's for chemical kinetics and sensitivity coefficients, equation (9.2)
$K_{c,j}$	concentration equilibrium constant for reaction j, equation (8.6)
k_j, k_{-j}	forward (eqs. (8.3) and (8.4)) and reverse (eq. (8.5)) rate coefficients for reaction j
L_m	characteristic length for shock tube kinetics calculations, equation (9.5)
L_s	cylinder stroke of Otto-cycle engine
L_w	perimeter of flow cross section, equation (8.23)
$\underline{\ell}$	integration method coefficient vector
ℓ_0	zeroth component of $\underline{\ell}$
M_j	third-body efficiency factor for reaction j, equation (A17)
M_w	mixture molar mass
$M_{w,i}$	molar mass of species i
M	general third-body collision partner in third-body reaction
METH	identifies integration method to be used for solving chemical kinetics ODE's (see eq. (9.1))
MF	method flag that specifies both integration method and corrector iteration technique to be used for solving chemical kinetics ODE's, equation (9.1)
MITER	identifies corrector iteration technique to be used in integrating chemical kinetics ODE's (see eq. (9.1))
\mathcal{M}	Mach number
m	mass

\dot{m}	mass flow rate
\dot{m}_0	mass flow rate to start iteration for first solution of assigned-temperature PSR problem
$\Delta \ln \dot{m}$	logarithmic correction of mass flow rate for assigned-temperature PSR calculation
\dot{m}''	mass flux (mass flow rate per unit cross-sectional area)
$m_{\ell j}$	relative third-body efficiency of species ℓ in reaction j , equation (A18)
N	number of ODE's in integration problem or number of algebraic equations in PSR problem
N_A	number of entries in assigned-variable (pressure or area) table
N_T	number of entries in assigned-temperature table
NCF	number of convergence failures on integration step while solving chemical kinetics ODE's
NCONV	convergence number for PSR calculation
NFE	total number of derivative evaluations required by chemical kinetics ODE solver for problem
NINERT	number of inert species in mixture
NITER	total number of iterations for equilibrium, shock, or PSR calculation
NJE	total number of Jacobian matrix evaluations required by chemical kinetics ODE solver for problem
NQ	method order for integration step
NQMAX	maximum method order to be attempted on any step
NR	number of chemical reactions in reaction mechanism
NRS	number of reacting species in mixture
NS	number of (reacting plus inert) species in mixture
NSTEP	total number of integration steps required by chemical kinetics ODE solver for problem
n_C	number of carbon atoms in one molecule of fuel
n_H	number of hydrogen atoms in one molecule of fuel
n_j	temperature exponent in forward rate coefficient expression for reaction j

Symbols	
Δn_j	change in number of moles for j th reaction, equation (8.7)
n_O	number of oxygen atoms in one molecule of fuel
Pr	Prandtl number, equation (8.27)
PFSUM	FSUM on previous iteration in PSR calculation
P	iteration matrix, used in solution of ODE's for chemical kinetics and sensitivity coefficients, equation (9.3)
p	pressure
p_c	rocket chamber pressure, psia
\dot{Q}	heat transfer rate
\dot{Q}'	heat transfer rate per unit length
q	method order for integration step
R	universal gas constant
R_j, R_{-j}	forward (eqs. (A3) and (A19)) and reverse (eqs. (A4) and (A20)) molar reaction rates per unit volume of reaction j
R	ratio of integration step size to be attempted next to its current value
Re	Reynolds number, equation (8.26)
RHDN	ratio of integration step size to be attempted next to its current value if method order is to be decreased
RHOCON	logical variable that indicates whether density is constant
RHSM	ratio of integration step size to be attempted next to its current value if method order is to be unchanged
RHUP	ratio of integration step size to be attempted next to its current value if method order is to be increased
RMAX	maximum factor by which integration step size can be increased when step size change is next considered
RPM	Otto-cycle engine speed, rpm
r_j	net molar rate per unit volume of reaction j , equation (8.57)
h_i	ratio of mole fraction of oxidant species i to that of oxygen, equation (8.47)
S_i	Sutherland constant for species i

Symbols

S_{ij}	sensitivity coefficient of i th dependent variable with respect to j th sensitivity parameter or Sutherland interaction constant for species i and j , equation (8.44)
$\langle S_{ij} \rangle$	normalized sensitivity coefficient of i th dependent variable with respect to j th sensitivity parameter
S_m	sum of mass fractions of all species, equation (12.11)
\underline{S}_j	sensitivity coefficient vector with respect to j th sensitivity parameter
S_i	chemical symbol for species i
\underline{S}_1	defined by equation (A34) or (A39)
\underline{S}_2	defined by equation (A35)
s	mass-specific entropy of mixture
s_i	molar-specific entropy of species i
T	temperature
$\Delta \ln T$	logarithmic correction of temperature for assigned-mass-flow-rate PSR calculation
T_{eq}	equilibrium temperature
T_w	wall temperature
TCONST	logical variable defined within code and indicates whether temperature is constant
TCPU	total CPU time for problem
TOUT	next value of independent variable at which solution to ODE's is required
t	time
u	mass-specific internal energy of mixture
V	flow velocity
\mathcal{V}	volume
v	mass-specific volume of mixture
W_i	net molar production rate per unit volume of species i , equation (12.8)
X_H	total energy exchange rate, equation (12.9)
$X_{H,j}$	net energy exchange rate for reaction j , equation (12.13)

Symbols

X_j	net reaction conversion rate for reaction j, equation (8.56)
x	distance
x_i	mole fraction of species i
\tilde{x}_i	mole fraction of species i in oxidant
Y_i	i th component of \mathbf{Y}
\mathbf{Y}	numerical solution vector to chemical kinetics ODE's
y_i	exact solution of i th dependent variable for chemical kinetics problems or mass fraction of species i
\tilde{y}_i	mass fraction of species i in oxidant
$\mathbf{Z}_j(k)$	k th column of \mathbf{Z}_j
\mathbf{Z}_j	Nordsieck history matrix for sensitivity coefficients with respect to j th sensitivity parameter
\mathbf{z}	Nordsieck history matrix for solution to chemical kinetics problem
β	shock tube boundary layer thickness parameter
β_0	integration method coefficient (see eq. (9.3))
Γ	quantity defined by equation (12.22)
γ	frozen specific heat ratio of mixture, equation (12.4)
δ_{ij}	Kronecker symbol, equation (A150)
η	exponent in equations (9.4) and (9.5) for shock tube flow area and characteristic length
η_j	j th sensitivity parameter
κ	thermal conductivity of mixture
κ_i	thermal conductivity of species i
Λ, Λ_1	underrelaxation factors used in controlling magnitude of corrections in PSR computation
μ	dynamic viscosity of mixture
μ_i	dynamic viscosity of species i
ν_i', ν_i''	stoichiometric coefficients of reactant and product species i in reactions (8.1) and (8.2)
ν_{ij}', ν_{ij}''	stoichiometric coefficients of reactant and product species i in reaction j

Symbols

ξ	“integration variable” (i.e., independent variable for chemical kinetics ODE’s)—time or distance
ξ_A	independent variable (time or distance) for assigned-variable (pressure or area) profile
ξ_T	independent variable (time or distance) for temperature profile
ρ	mixture mass density
σ_i	mole number of species <i>i</i> —moles of species <i>i</i> per unit mass of mixture
$\Delta \ln \sigma_i$	logarithmic correction of mole number of species <i>i</i> for PSR calculation
σ_m	sum of mole numbers of species, equation (12.23)
τ_r	average residence time for PSR
φ	fuel-oxidant equivalence ratio
ϕ_{ij}	interaction term used to calculate mixture dynamic viscosity, equation (8.40)
$\bar{\phi}_{ij}$	interaction term used to calculate mixture thermal conductivity, equation (8.42)
χ	quantity defined by equation (A32)
Ω	cranking angular velocity of Otto-cycle engine, equation (8.31)
ω_{ij}	net molar formation rate of species <i>i</i> per unit volume by reaction <i>j</i> , equation (8.58)
Subscripts:	
air	oxidant
<i>f</i>	fuel
<i>N</i>	index for either temperature or mass flow rate in PSR calculation
NR	index for density derivative in chemical kinetics calculation
NT	index for temperature derivative in chemical kinetics calculation
NV	index for velocity derivative in chemical kinetics calculation
<i>n</i>	value at ξ_n
out	value at next output station
<i>s</i>	stoichiometric
st	standard conditions (1 atm)

Symbols

0	initial condition value
1	condition upstream of incident shock wave or assigned value for first solution of PSR problem
2	condition downstream of incident shock wave

Superscripts:

(j)	j th derivative
$[m]$	value at m th iteration
$[0]$	predicted value
$^{\circ}$	standard state (1 atm)
*	value at perfectly stirred reactor inlet

Chapter 8

Capabilities of Code

The LSENS code has been designed for the following reaction models and computations:

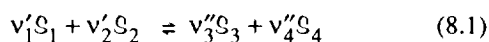
- (1) Static reaction either at constant density or with assigned pressure
- (2) One-dimensional flow reaction with an assigned pressure or area profile
- (3) Static or flow reaction with an assigned temperature profile
- (4) Sensitivity analysis for a static reaction
- (5) Equilibrium reaction
- (6) Reaction initiated by an incident shock wave
- (7) Reaction in a perfectly stirred reactor

Any kinetic reaction problem for which temperature is not assigned may either be adiabatic or have a prescribed rate of heat exchange with its environment.

In this chapter we describe the computational capabilities of LSENS. We also discuss convenience features and the calculation procedures used for the thermodynamic and transport properties.

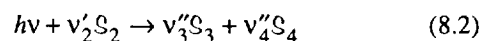
8.1 Types of Chemical Reaction and Rate Coefficient

Many different types of elementary chemical reaction 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 species and two different product species and thus can be written in the general form



where v_i' is the stoichiometric coefficient (i.e., number of moles) of reactant species i in the reaction, v_i'' is the stoichio-

metric coefficient of product species i in the reaction, and S_i is the chemical symbol for species i . In equation (8.1) species S_1 and/or S_4 may be either absent or the general third-body collision 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:



where $h\nu$ represents a single quantum of radiation absorbed by the reactant. This reaction is an irreversible decomposition.

All reactions are assumed to be elementary (i.e., real molecular events, e.g., ref. 1), so that the $\{v_i'\}$ and $\{v_i''\}$ are integers. Also all species are assumed to be ideal gases. For each reaction j , irrespective of its type, the forward rate coefficient k_j is usually given by the empirical expression (ref. 2)

$$k_j = A_j T^{n_j} \exp\left(\frac{-E_j}{RT}\right) \quad (8.3)$$

In this equation the preexponential factor A_j , the temperature exponent n_j , and the activation energy E_j are constants, R is the universal gas constant, and T is the temperature. Provision has also been made for the following alternative form of the rate coefficient expression (ref. 3):

$$k_j = A_j T^{n_j} \exp(c_j T) \quad (8.4)$$

where c_j is a constant.

The backward rate coefficient k_{-j} need not be specified for a reversible reaction. For both forward rate coefficient expressions k_{-j} is computed within the code by using the principle of detailed balancing or microscopic reversibility (refs. 1 and 4):

8. Capabilities of Code

$$k_{-j} = \frac{k_j}{K_{c,j}} \quad (8.5)$$

Here $K_{c,j}$ is the concentration equilibrium constant for reaction j and for a given temperature is computed as follows (ref. 4):

$$K_{c,j} = (RT)^{-\Delta n_j} \exp\left(\frac{-\Delta G_{T,j}^\circ}{RT}\right) \quad (8.6)$$

In this equation Δn_j is the change in the total number of moles when the reactants are converted into products

$$\Delta n_j = \sum_{i=3}^4 v_{ij}'' - \sum_{i=1}^2 v_{ij}' \quad (8.7)$$

where v_{ij}' and v_{ij}'' are, respectively, the stoichiometric coefficients of reactant species i and product species i in reaction j . The term $\Delta G_{T,j}^\circ$ is the standard-state (here 1 atm) Gibbs function change for the j th reaction at temperature T

$$\Delta G_{T,j}^\circ = \sum_{i=3}^4 v_{ij}'' g_i^\circ(T) - \sum_{i=1}^2 v_{ij}' g_i^\circ(T) \quad (8.8)$$

where $g_i^\circ(T)$ is the standard-state Gibbs function of species i at temperature T . The calculation procedure for g_i° is described in section 8.6.

8.2 Assigned Variable and Specification

For a static problem the density ρ is constant or the pressure p is specified, either as a constant or as a function of time t . The following two methods for specifying the pressure-time profile are built into the code: (1) as a polynomial function of up to third degree

$$p(t) = C_{t,0} + C_{t,1} t + C_{t,2} t^2 + C_{t,3} t^3 \quad (8.9)$$

where the $\{C_{t,j}\}$ are constants, and (2) in tabular form, wherein the pressure p_i ($i = 1, \dots, N_A$) is assumed to be given at each of the N_A discrete points t_i ($i = 1, \dots, N_A$).

For a flow problem the pressure p or the area A is assigned, either as a constant or as a function of time or distance, independently of the choice of the independent variable ξ for the governing ordinary differential equations (ODE's). For clarity in presentation we will refer to ξ as the "integration variable." To specify the assigned variable (pressure or area), that is, the variable that can be assigned as nonconstant, the two methods given previously for a static problem are avail-

able. If we represent the assigned variable by A_V and the variable it is a function of by ξ_A , the third-degree polynomial representation used in the code is given by

$$A_V(\xi_A) = C_{\xi_A,0} + C_{\xi_A,1} \xi_A + C_{\xi_A,2} \xi_A^2 + C_{\xi_A,3} \xi_A^3 \quad (8.10)$$

where the $\{C_{\xi_A,j}\}$ are constants. The second method requires values for $\xi_{A,i}$ ($i = 1, \dots, N_A$) and the corresponding $A_{V,i}$ ($i = 1, \dots, N_A$) at the N_A discrete points.

For both static and flow problems A_V and its first derivative $dA_V/d\xi$ must be evaluated for a given value of ξ (see the equations in appendix A). Because the $p-t$ profile for a static problem is a particular case of the $A_V-\xi_A$ profile for a flow problem, we restrict discussion to the latter case. However, there is one fundamental difference between the two cases. For static problems $\xi_A (= t)$ and $\xi (= t)$ are the same, but they may be different for a flow problem. We postpone to section 8.4 a description of how ξ_A is calculated and will assume in the rest of this section that it is known.

If the assigned variable profile is specified by a polynomial, its value is obtained from equation (8.10), its first derivative with respect to ξ_A is given by

$$\frac{dA_V}{d\xi_A} = C_{\xi_A,1} + 2C_{\xi_A,2} \xi_A + 3C_{\xi_A,3} \xi_A^2 \quad (8.11)$$

and $dA_V/d\xi$ is then computed by using the chain rule of differentiation as

$$\frac{dA_V}{d\xi} = \frac{dA_V}{d\xi_A} \frac{d\xi_A}{d\xi} \quad (8.12)$$

When the assigned variable profile is specified in tabular form, A_V and $dA_V/d\xi_A$ are determined by cubic spline interpolation with parabolic runout at the end conditions (e.g., refs. 5 and 6); $dA_V/d\xi$ is then given by equation (8.12).

8.3 Assigned-Temperature Problem

The temperature of the reacting system may be assigned for both static and flow problems, either as a constant or as a function of time or distance, independently of both ξ and ξ_A . The same two methods described in section 8.2 for the assigned variable are built into the code for specifying the temperature.

If we denote by ξ_T the variable that T is assigned as a function of, the polynomial representation used in the code is given by

$$T(\xi_T) = C_{\xi_T,0} + C_{\xi_T,1} \xi_T + C_{\xi_T,2} \xi_T^2 + C_{\xi_T,3} \xi_T^3 \quad (8.13)$$

where the $\{C_{\xi_T,j}\}$ are constants. The tabular form of profile specification requires values for $\xi_{T,i}$ ($i=1, \dots, N_T$) and the corresponding T_i ($i=1, \dots, N_T$) at the N_T discrete points. In either case T and its first derivative $dT/d\xi$ must be evaluated for a given ξ . For a static problem $\xi_T = \xi$ ($= t$), but for a flow problem ξ_T may be different from ξ and, if so, is evaluated as described in section 8.4.

The procedures for computing T and $dT/d\xi$ are exactly the same as those described in section 8.2 for computing the assigned variable and its first derivative. For a polynomial representation T and $dT/d\xi_T$ are obtained from equation (8.13). If the temperature is specified in tabular form, the same cubic spline interpolation routine used to compute the assigned variable and its first derivative gives T and $dT/d\xi_T$. In both cases, if $\xi_T \neq \xi$, the chain rule of differentiation, analogous to equation (8.12), is used to calculate $dT/d\xi$.

8.4 Time and Distance Computations for Flow Problem

As discussed in sections 8.2 and 8.3, for a flow problem the independent variable ξ_A for the assigned variable (pressure or area) and/or the independent variable ξ_T for the temperature may be different from the integration variable ξ . It is therefore necessary to compute the distance x (or time t) when time (or distance) is the integration variable. The most obvious way of accomplishing this objective is to treat distance (or time) as an additional dependent variable when time (or distance) is the integration variable and solve its ODE. For example, if time is the integration variable, the ODE for distance is given by

$$\left. \begin{aligned} \frac{dx}{dt} &= V \\ x(t=t_0) &= \text{Given} \end{aligned} \right\} \quad (8.14)$$

where V is the velocity. However, this method introduces another ODE and increases the size of the iteration matrix. In the present work an alternative, easier calculation procedure, a cubic spline interpolation that is described here, is used to compute x (or t) from t (or x).

The ODE solver used in LSENS generates numerical solutions at discrete points ξ_n ($n=1, 2, \dots$). We assume that time is the integration variable and that the solution has been advanced over the time step $[t_{n-1}, t_n]$ of size h_n ($= t_n - t_{n-1}$). We now assume that V varies locally (i.e., in the interval $[t_{n-1}, t_n]$) as a cubic polynomial

$$V(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (8.15)$$

where the constants $\{a_j\}$ have to be determined. By requiring that V and dV/dt satisfy the constraints

$$\left. \begin{aligned} V(t_{n-1}) &= V_{n-1} \\ V(t_n) &= V_n \\ \frac{dV}{dt} \Big|_{t=t_{n-1}} &= \dot{V}_{n-1} \\ \frac{dV}{dt} \Big|_{t=t_n} &= \dot{V}_n \end{aligned} \right\} \quad (8.16)$$

where V_n and \dot{V}_n are the numerical solution values for V and dV/dt at $t=t_n$, we can solve for the four unknowns a_0 to a_3 . Substituting the solutions for these quantities into equation (8.15) and rearranging terms gives

$$\begin{aligned} V(t) &= V_{n-1} + (t - t_{n-1}) \dot{V}_{n-1} \\ &+ \left(\frac{t - t_{n-1}}{h_n} \right)^2 \left(3V_n - 3V_{n-1} - h_n \dot{V}_n - 2h_n \dot{V}_{n-1} \right) \\ &- \left(\frac{t - t_{n-1}}{h_n} \right)^3 \left(2V_n - 2V_{n-1} - h_n \dot{V}_n - h_n \dot{V}_{n-1} \right) \end{aligned} \quad (8.17)$$

The derivatives \dot{V}_{n-1} and \dot{V}_n are obtained from the Nordsieck history arrays \mathbf{z}_{n-1} and \mathbf{z}_n , respectively. This history array contains the solution and its scaled derivatives. Substituting equation (8.17) into equation (8.14), integrating over the interval $[t_{n-1}, t_n]$, and collecting terms give the following expression for x_n :

$$x_n = x_{n-1} + h_n \left(\frac{V_{n-1} + V_n}{2} \right) + h_n^2 \left(\frac{\dot{V}_{n-1} - \dot{V}_n}{12} \right) \quad (8.18)$$

If distance is the integration variable, we replace h_n in equation (8.18) by the difference $t_n - t_{n-1}$, where t_{n-1} is

8. Capabilities of Code

known and t_n has to be computed. Also, we replace the difference $x_n - x_{n-1}$ by h_n , the step size used on the step $[x_{n-1}, x_n]$. Finally, because x is the integration variable, we use the chain rule of differentiation to replace dV/dt by $V dV/dx (= V\dot{V})$. The resulting equation can then be regarded as a quadratic equation in the quantity $t_n - t_{n-1}$. Its solution is given by

$$t_n - t_{n-1} = \frac{-(V_{n-1} + V_n) + \left[(V_{n-1} + V_n)^2 + 4h_n \left(\frac{V_{n-1}\dot{V}_{n-1} - V_n\dot{V}_n}{3} \right) \right]^{1/2}}{\frac{V_{n-1}\dot{V}_{n-1} - V_n\dot{V}_n}{3}} \quad (8.19)$$

where we have used the positive root to ensure that t increases as the integration proceeds. (Because the overall reaction problem is irreversible, we can solve the governing ODE's only for increasing t .) The difficulty with equation (8.19) is the possibility of overflows if $V_{n-1}\dot{V}_{n-1} = V_n\dot{V}_n$. For example, for constant-velocity flow $V_{n-1} = V_n$ and $\dot{V}_{n-1} = \dot{V}_n (= 0)$, and this condition is obtained. We avoid this difficulty by multiplying both the numerator and denominator of equation (8.19) by the quantity

$$V_{n-1} + V_n + \left[(V_{n-1} + V_n)^2 + 4h_n \left(\frac{V_{n-1}\dot{V}_{n-1} - V_n\dot{V}_n}{3} \right) \right]^{1/2}$$

Performing this operation and then simplifying the resulting expression give the following equation for t_n :

$$t_n = t_{n-1} + \frac{4h_n}{V_{n-1} + V_n + \left[(V_{n-1} + V_n)^2 + 4h_n \left(\frac{V_{n-1}\dot{V}_{n-1} - V_n\dot{V}_n}{3} \right) \right]^{1/2}} \quad (8.20)$$

which avoids the problem of overflow errors.

8.5 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 transfer 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. It was previously assumed (ref. 7) that the simplest expression for the heat loss rate is a polynomial function of the reacting mixture temperature. Therefore the standard option for specifying the heat transfer rate \dot{Q} for a static problem or the heat transfer rate per unit length in the flow direction \dot{Q}' for a flow problem is given by the polynomial expression

$$\dot{Q} \text{ (or } \dot{Q}') = H_{T,0} + H_{T,1} T + H_{T,2} T^2 + H_{T,3} T^3 + H_{T,4} T^4 \quad (8.21)$$

where the $\{H_{T,j}\}$ are constants. Note that \dot{Q} (or \dot{Q}') can be made negative to simulate a heat source for an ignition problem. This heat transfer model is also used to compute the heat transfer rate for a perfectly stirred reactor problem.

For one-dimensional flow problems another calculation procedure for \dot{Q}' is included in LSENS. In this model \dot{Q}' is given by the following empirical expression (e.g., ref. 8):

$$\dot{Q}' = \mathfrak{H} L_w (T - T_w) \quad (8.22)$$

where \mathfrak{H} is the heat transfer coefficient (heat transfer rate per unit surface area per unit temperature difference), L_w is the perimeter of the flow cross section, and T_w is the wall temperature. For a cylindrical tube of diameter D the cross-sectional area $A = \pi D^2/4$, so that

$$L_w = \pi D = 2\sqrt{\pi A} \quad (8.23)$$

Substituting this equation into equation (8.22) gives

$$\dot{Q}' = 2\sqrt{\pi A} \mathfrak{H} (T - T_w) \quad (8.24)$$

Many correlations can be used to estimate \mathfrak{H} (e.g., ref. 8). For turbulent flow we use the following empirical correlation:

$$\frac{\mathfrak{H} D}{\kappa} = 0.023 \text{Re}^{0.8} \text{Pr}^{0.3} \quad (8.25)$$

In this equation Re and Pr are, respectively, the Reynolds and Prandtl numbers:

$$\text{Re} = \frac{\rho V D}{\mu} \quad (8.26)$$

and

$$\text{Pr} = \frac{\mu c_p}{\kappa} \quad (8.27)$$

where μ , c_p , and κ are, respectively, the mixture dynamic viscosity, mass-specific heat at constant pressure, and thermal conductivity. These properties are evaluated at the fluid bulk temperature, which is equal to the reacting fluid temperature for one-dimensional flow (ref. 8). The thermodynamic property c_p and the transport properties μ and κ for the mixture are computed by using the empirical equations given in sections 8.6 and 8.7, respectively. The correlation given by equation (8.25) is used for flows with $Re \geq 2300$. For laminar flow ($Re < 2300$) the following correlation is used (ref. 8):

$$\frac{\mathcal{H}D}{\kappa} = 4.364 \quad (8.28)$$

This model is used to compute the heat transfer rate per unit mass of reacting mixture \dot{Q}/m for one special static reaction application—the assigned-pressure, variable-volume reaction in the cylinder of an internal combustion, or Otto-cycle, engine. We assume that the cylinder has diameter B (bore) and length L_s (stroke) and that heat transfer occurs only through the curved cylindrical surface. The heat transfer rate per unit mass of mixture is then given by

$$\frac{\dot{Q}}{m} = \frac{\mathcal{H}A_s(T - T_w)}{\rho\mathcal{V}} = \frac{4\mathcal{H}(T - T_w)}{B\rho} \quad (8.29)$$

where A_s is the instantaneous area of the curved surface, \mathcal{V} is the instantaneous chamber volume, and we have used the fact that $\mathcal{V}/A_s = B/4$ for a cylinder. To compute \mathcal{H} , we use the correlation of equation (8.25) or (8.28) with D replaced by B and the Reynolds number computed by (ref. 9)

$$Re = \frac{\rho L_s \Omega B}{\pi\mu} \quad (8.30)$$

where Ω is the cranking angular velocity in radians per second and is calculated from the engine speed RPM in revolutions per minute by

$$\Omega = \frac{\pi \text{RPM}}{30} \quad (8.31)$$

8.6 Thermodynamic Properties

The thermodynamic properties of the species are computed by using the empirical equations given by Gordon and

McBride (refs. 10 and 11). For each species i the non-dimensional properties molar-specific heat at constant pressure $c_{p,i}/R$, molar-specific enthalpy h_i/RT , and the 1-atm molar-specific entropy s_i°/R are specified as functions of temperature

$$\frac{c_{p,i}}{R} = a_{i,1} + a_{i,2}T + a_{i,3}T^2 + a_{i,4}T^3 + a_{i,5}T^4 \quad (8.32)$$

$$\frac{h_i}{RT} = a_{i,1} + \frac{a_{i,2}}{2}T + \frac{a_{i,3}}{3}T^2 + \frac{a_{i,4}}{4}T^3 + \frac{a_{i,5}}{5}T^4 + \frac{a_{i,6}}{T} \quad (8.33)$$

$$\frac{s_i^\circ}{R} = a_{i,1} \ln T + a_{i,2}T + \frac{a_{i,3}}{2}T^2 + \frac{a_{i,4}}{3}T^3 + \frac{a_{i,5}}{4}T^4 + a_{i,7} \quad (8.34)$$

In these equations R is the universal gas constant and the degree sign indicates standard conditions (here pressure of 1 atm). This superscript is not attached to the specific heat and enthalpy because the two properties are independent of pressure for ideal gases.

For each species i the seven coefficients $a_{i,1}$ to $a_{i,7}$ are computed with the Properties and Coefficients (PAC) code of McBride and Gordon (ref. 12). For ideal gases this program first calculates thermodynamic properties from molecular constant data given in the JANAF thermochemical tables (e.g., ref. 13) and then reduces them to coefficient form. 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.

Another thermodynamic property that is required for chemical kinetics computations is the standard-state, molar-specific Gibbs function g_i° . The normalized Gibbs function g_i°/RT is given by

$$\frac{g_i^\circ}{RT} = \frac{h_i - Ts_i^\circ}{RT} \quad (8.35)$$

The thermodynamic properties $\{dc_{p,i}/dT\}$ may also be required and are obtained by differentiating equation (8.32) with respect to T .

The thermodynamic properties of the reacting gas mixture are evaluated by applying Gibbs theorem (e.g., ref. 14), that is, by simply summing the contributions made by each species: for example, the mixture mass-specific enthalpy h is given by

8. Capabilities of Code

$$h = \sum_{i=1}^{NS} h_i \sigma_i \quad (8.36)$$

where σ_i is the number of moles of species i in unit mass of mixture and NS is the total number of (reacting and inert) species.

8.7 Transport Properties

The transport properties of the species are computed by using the empirical equations given by Zeleznik and McBride (ref. 9); see also Gordon et al. (refs. 11 and 15). For each species i the dynamic viscosity μ_i and the thermal conductivity κ_i are specified as functions of temperature in the form used by Maitland and Smith (ref. 16):

$$\ln \mu_i = a_{i,\mu} \ln T + \frac{b_{i,\mu}}{T} + \frac{c_{i,\mu}}{T^2} + d_{i,\mu} \quad (8.37)$$

$$\ln \kappa_i = a_{i,\kappa} \ln T + \frac{b_{i,\kappa}}{T} + \frac{c_{i,\kappa}}{T^2} + d_{i,\kappa} \quad (8.38)$$

The coefficients in these equations are obtained by least-squares fitting and give viscosity in units of micrograms per centimeter-second and thermal conductivity in units of microcalories per second-centimeter-Kelvin.

The transport properties μ and κ for the gas mixture are also computed by using the procedures given by Zeleznik and McBride (ref. 9). The mixture viscosity is calculated with the formula suggested by Wilke (refs. 17 and 18):

$$\mu = \sum_{i=1}^{NS} \mu_i \left(\frac{\sigma_i}{\sum_{j=1}^{NS} \phi_{ij} \sigma_j} \right) \quad (8.39)$$

where

$$\phi_{ij} = \frac{\sqrt{2} \left[1 + \left(\frac{\mu_i}{\mu_j} \right)^{1/2} \left(\frac{M_{w,j}}{M_{w,i}} \right)^{1/4} \right]^2}{4 \left(1 + \frac{M_{w,i}}{M_{w,j}} \right)^{1/2}} \quad (8.40)$$

where $M_{w,i}$ is the molar mass of species i .

For the mixture thermal conductivity the formula proposed by Lindsay and Bromley (ref. 19) is used:

$$\kappa = \sum_{i=1}^{NS} \kappa_i \left(\frac{\sigma_i}{\sum_{j=1}^{NS} \bar{\phi}_{ij} \sigma_j} \right) \quad (8.41)$$

where

$$\bar{\phi}_{ij} = \frac{1}{4} \left\{ 1 + \left[\left(\frac{\mu_i}{\mu_j} \right) \left(\frac{M_{w,j}}{M_{w,i}} \right)^{3/4} \left(\frac{1+S_i/T}{1+S_j/T} \right) \right]^{1/2} \right\}^2 \left(\frac{1+S_{ij}/T}{1+S_i/T} \right) \quad (8.42)$$

In this equation S_i is the Sutherland constant for the i th species and S_{ij} is the Sutherland interaction constant for species i and j (refs. 9 and 15). The constant S_i is obtained from the viscosity as follows (ref. 9):

$$1 + \frac{S_i}{T} = \left(\frac{3}{2} - \frac{d \ln \mu_i}{d \ln T} \right)^{-1} \quad (8.43)$$

The constant S_{ij} is generally assumed to be of the form (refs. 9 and 15)

$$S_{ij} = C_{ij} \sqrt{S_i S_j} \quad (8.44)$$

where C_{ij} is a constant. Lindsay and Bromley (ref. 19) recommend the value $C_{ij} = 1$, except when one of the species i and j is highly polar. For this case the value $C_{ij} = 0.733$ is suggested (ref. 9). We use $C_{ij} = 1$ for all pairs of species except those involving water, for which C_{ij} is set equal to 0.733.

8.8 Simplified Composition Input

The initial mixture composition may be specified by means of the species mole fractions x_i ($i = 1, \dots, NS$) or mass fractions y_i ($i = 1, \dots, NS$). The internal composition variables $\{\sigma_i\}$ are then computed by

$$\sigma_i = \frac{x_i}{\sum_{j=1}^{NS} x_j M_{w,j}}, \quad i = 1, \dots, NS \quad (8.45)$$

or

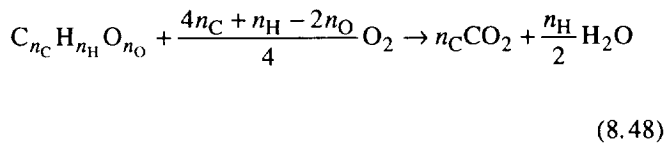
$$\sigma_i = \frac{y_i}{M_{w,i}}, \quad i = 1, \dots, NS \quad (8.46)$$

The code also includes the option of specifying the initial composition for any reaction involving the general hydrocarbon fuel $C_{n_C}H_{n_H}O_{n_O}$ by the simple means of either the fuel-oxidant equivalence ratio ϕ or the fuel-oxidant mass ratio f/a . The oxidant may contain oxygen, nitrogen, argon, and carbon dioxide in any proportion. A "standard air" oxidant has been built into the code, but the user has the option of changing it. The composition of this oxidant is given in table 8.1. Its molar mass is 28.9644. The quantities \bar{x}_i and \bar{y}_i in table 8.1 are the mole and mass fractions, respectively, of species i in the oxidant, and h_i is the ratio of the mole fraction of oxidant species i to that of oxygen

$$h_i = \frac{\bar{x}_i}{\bar{x}_{O_2}} \quad (8.47)$$

For the standard air given in table 8.1 it is the $\{h_i\}$ values that are built into the code.

To obtain the relations between species mole fractions and ϕ or f/a , we use the stoichiometric fuel-oxygen reaction



which is valid for any fuel with $(4n_C + n_H) > 2n_O$. The fuel-oxidant equivalence ratio is defined as

$$\phi = \frac{F/O}{(F/O)_s} \quad (8.49)$$

where F/O is the actual fuel-oxygen molar ratio and $(F/O)_s$ is its stoichiometric value. It is clear from reaction (8.48) and

TABLE 8.1.—STANDARD OXIDANT COMPOSITION BUILT INTO LSENS
[Molar mass, 28.9644.]

Species, S_i	Mole fraction, \bar{x}_i	Mass fraction, \bar{y}_i	Mole fraction ratio, $h_i (= \bar{x}_i/\bar{x}_{O_2})$
N ₂	0.78088	0.75524	3.727350
O ₂	0.20950	0.23145	-----
Ar	0.00932	0.01285	0.044487
CO ₂	0.00030	0.00046	0.001432

the definition of ϕ that

$$\frac{x_f}{x_{O_2}} = \frac{4\phi}{4n_C + n_H - 2n_O} \quad (8.50)$$

where the subscript f denotes fuel.

Now the mole fraction of each oxidant species i in the fuel-oxidant mixture is related to x_{O_2} via

$$x_i = h_i x_{O_2} \quad (8.51)$$

which together with equation (8.50) and the identity

$$x_f + x_{O_2} + x_{N_2} + x_{Ar} + x_{CO_2} = 1 \quad (8.52)$$

gives the following expression for x_{O_2} :

$$x_{O_2} = \frac{4n_C + n_H - 2n_O}{4\phi + (4n_C + n_H - 2n_O)(1 + h_{N_2} + h_{Ar} + h_{CO_2})} \quad (8.53)$$

The fuel and oxidant species mole fractions can then be computed by using equations (8.50) and (8.51). Finally equation (8.45) gives the $\{\sigma_i\}$.

If the fuel-oxidant mass ratio is specified, we first compute x_f/x_{O_2} by using the relation

$$\frac{x_f}{x_{O_2}} = \frac{M_{w,air} f/a}{M_{w,f} \bar{x}_{O_2}} \quad (8.54)$$

where the subscript "air" denotes oxidant and

$$\bar{x}_{O_2} = \frac{1}{1 + h_{N_2} + h_{Ar} + h_{CO_2}} \quad (8.55)$$

Equation (8.50) then gives ϕ , from which the initial mixture composition can be computed by using the procedure described above.

8.9 Net Reaction and Species Formation Rates

The relative importance of the j th reaction to the formation or destruction rate of the i th species can be gauged by examining its net reaction rate per unit volume r_j and the time rate

8. Capabilities of Code

of change of σ_i due to the j th reaction, $(d\sigma_i/dt)_j$. However, these rates are converted to more useful quantities as follows: For each reaction j the code computes the net reaction conversion rate X_j , which is defined as

$$X_j \equiv \frac{r_j}{\rho^2} \quad (8.56)$$

where r_j is given by

$$r_j = R_j - R_{-j} \quad (8.57)$$

Here R_j and R_{-j} are, respectively, the molar forward and reverse rates per unit volume of reaction j . The larger magnitudes of the $\{X_j\}$ make it easier to use them than the $\{r_j\}$ to compare net reaction rates. The net molar rate of formation of species i per unit volume by reaction j , ω_{ij} , which is defined by

$$\omega_{ij} \equiv \rho \left(\frac{d\sigma_i}{dt} \right)_j = (v_{ij}'' - v_{ij}') r_j \quad (8.58)$$

is also generated. The $\{\omega_{ij}\}$ are useful in developing a chemical kinetic mechanism because they show which reactions are important in forming or destroying a species.

8.10 Equilibration Factor

It is often important to know how far a chemical reaction is from its equilibrium condition. To determine the state of the j th reaction, we derive a simple equilibration factor from

r_j , equation (8.57). Now r_j will be positive or negative depending on the magnitudes of R_j and R_{-j} . If R_j is greater than R_{-j} , we divide equation (8.57) by R_j to get

$$\frac{r_j}{R_j} = 1 - \frac{R_{-j}}{R_j} \quad (8.59)$$

If the reaction is far from equilibrium, R_j will be significantly larger than R_{-j} and the ratio r_j/R_j will be a positive number very close to unity. If the reaction is near equilibrium, $R_j \approx R_{-j}$ and r_j/R_j will be a positive number close to zero. Therefore r_j/R_j varies between 0 and 1 for the extreme situations of equilibrium ($R_j = R_{-j}$) and irreversibility ($R_{-j} = 0$).

If, however, R_{-j} is greater than R_j , we divide equation (8.57) by R_{-j} to get

$$\frac{r_j}{R_{-j}} = \frac{R_j}{R_{-j}} - 1 \quad (8.60)$$

and the ratio r_j/R_{-j} varies between 0 and -1 for the extreme conditions of equilibrium ($R_j = R_{-j}$) and irreversibility ($R_j = 0$).

The two ratios, equations (8.59) and (8.60), can be combined into a single equilibration factor $E_{q,j}$

$$E_{q,j} = \left| \frac{r_j}{\max(R_j, R_{-j})} \right| \quad (8.61)$$

where the vertical bars denote absolute value and $0 \leq E_{q,j} \leq 1$. The equilibration factor is useful in determining the effect of a single reaction on the entire complex process and is computed for each reaction.

Chapter 9

Code Organization and Description

9.1 Introduction

The present version of the LSENS code, dated July 22, 1992, is in double precision. It is compatible with most FORTRAN 77 compilers. However, it contains detailed instructions on how to convert it to FORTRAN 66. The MAIN program lists all routines requiring modifications. Also, in each such routine both the changes and the lines where they must be made are given.

The code 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, Alliant FX/S, Convex C220 minicomputer using the Convex 8.0 OS, and VAX 11/750, 11/780, 11/785, 6320, 6520, 8650, 8800, and 9410 using the VAX/VMS OS and VAX FORTRAN compiler; NASA Ames Research Center's Cray-2 and Cray-Y/MP using UNICOS and CFT77; the Sun SPARCstation 1 using the Sun 4.1 OS; several IRIS workstations using the IRIX 4.0.1 OS and F77 compiler; and the IBM RISC System/6000 using the AIX 3.1 OS and the XLF and F77 compilers.

The code consists of a MAIN program, 59 subprograms, and a BLOCK DATA module. Table 9.1 lists these subprograms in the order that they appear in the code and briefly describes them. The subprograms SECCPU, DDOT, DIMACH, IDAMAX, and VNORM are function routines; all others are subroutines. Also the routine CUBS contains statement functions. The function SECCPU, which computes the incremental central processing unit (CPU) time in seconds, calls the system clock to obtain the total CPU time used since initiation of the job. The code uses the following intrinsic and external routines: ALOG, DABS, DBLE, DEXP, DFLOAT, DLOG, DMAX1, DMIN1, DSIGN, DSQRT, EXP, FLOAT, IABS, IFIX, MAX0, MIN0, MOD, READ, SNGL,

and WRITE. Finally, subroutines INIT, KINP, TINP, and SENSIN use namelists.

The different subprograms that make up the LSENS package are arranged in three blocks or groups as follows: The first group contains the MAIN program and those (CIMAGE to WSR) related to thermodynamic, transport, and kinetics computations. The second group includes the subroutines (SENDDM to SNSTAB) required for sensitivity analysis. The last group contains the subprograms (LSODE to XSETUN) included in the code LSODE (refs. 20 to 22), which is used to solve the governing ordinary differential equations (ODE's). Within each group the subprograms are arranged alphabetically. The BLOCK DATA module is located at the end of the code.

LSENS 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. An example is the calculation of the thermodynamic properties $dc_{p,i}/dT$ ($i = 1, \dots, NS$), where $c_{p,i}$ is the constant-pressure, molar-specific heat of species i , T is the temperature, and NS is the total number of (reacting and inert) species. These derivatives are needed only for the computation of certain elements of the Jacobian matrix (see eq. (9.2)), which is required for the numerical integration of the ODE's and for sensitivity analysis. Hence they are computed in subroutine PEDERV, which computes the Jacobian matrix, and not in subroutine THRM, which computes all other thermodynamic properties of the species (table 9.1).

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 under way to extend the temperature range over which the calculations are valid; it requires the

TABLE 9.1.—DESCRIPTION OF SUBPROGRAMS USED IN LSENS

Subprogram	Description
MAIN	Manages solution of problem by calling chemical kinetics preprocessor, integrator or perfectly stirred reactor solver, and output subprograms. Also computes CPU times and prints them.
CIMAGE	Reads in and prints image of input data for each case.
COMB	Manages calls for equilibrium calculations. Also computes CPU time for this calculation and prints it.
CUBS	Computes assigned variable or temperature and its first derivative by using either user-supplied polynomial equation or cubic spline fit of user-provided table of values.
DIFFUN	Computes derivatives of dependent variables with respect to independent variable.
ELEMNT	Computes element composition in atoms per gram of mixture.
EQLBRM	Performs equilibrium computations for assigned pressure and enthalpy, assigned pressure and temperature, assigned density and internal energy, and assigned density and temperature.
GAUSS	Solves a set of linear equations by using Gaussian elimination.
HETRAN	Computes heat loss terms and, if necessary, reads in viscosity and thermal conductivity data.
INIT	Reads in initial (or reactor inlet) mixture composition and values of thermodynamic and flow variables.
KINP	Processes and checks legality of all input data including reaction mechanism but not temperature profile for assigned-temperature problem or sensitivity analysis data. At user's option, tests reaction mechanism uniqueness and atom/charge balance of each reaction.
MATRIX	Sets up matrices used in equilibrium computations.
OUT1	Prints all input information including reactions and rate coefficient parameters.
OUT2	Prints general output for kinetics and sensitivity problems and equilibrium solution for perfectly stirred reactor problem.
PEDERV	Computes elements of Jacobian matrix.
RKTOUT	Computes and prints rocket performance parameters.
RXNTAB	Tabulates and prints reaction numbers in order of decreasing importance for each species (i.e., decreasing rates of production of each species by reactions) and prints all nonzero rates of production by each reaction for every species.
SECCPU	Computes CPU time increment in seconds.
SHOCKS	Solves incident shock equations for equilibrium and frozen cases.
SHOK	Manages setup and subprogram calls for incident shock calculations. Also, computes and prints CPU times for equilibrium and frozen shock calculations.
SPLINE	Performs cubic spline fit by using first-derivative method with parabolic runout for end conditions.
SPOUT	Prints results of shock and equilibrium problems. Also transfers frozen shock results to input of shock-kinetics problem and, if necessary, computes characteristic shock tube reaction length.
THRM	Computes nondimensional thermodynamic properties of species.
TINP	Processes and checks legality of temperature-related data for assigned-temperature problem.
WSOUT	Prints perfectly stirred reactor solution.
WSR	Performs perfectly stirred reactor computation.
SENDDM	Main integration and sensitivity solution driver. Manages call to core integrator and sensitivity subprograms that set up work array pointers and initialize sensitivity arrays.
DFDP	Computes partial derivatives of temporal derivatives of dependent variables with respect to sensitivity parameters.
DMACHK	Sets up real and integer work array pointers for core integrator and sensitivities, and checks that storage requirements are met.
DMINIT	Initializes sensitivity arrays.
DMINTR	Interpolates for sensitivity coefficients and their derivatives at output times.
DMPJAC	Computes iteration matrix and manages subprogram call for its LU-decomposition.
DMPRNT	Prints storage and computational work requirements for sensitivity analysis.
DMSTOD	Advances sensitivity solution by one time step.
SENSIN	Processes and checks legality of input data for sensitivity computations.
SNSOUT	Computes and normalizes sensitivity coefficients when pressure sensitivities are not required. Prints normalized sensitivity coefficients.
SNSPSC	Computes and normalizes sensitivity coefficients when pressure sensitivities are required.
SNSTAB	Tabulates and prints reaction numbers in order of decreasing importance (i.e., normalized sensitivity) and nonzero normalized sensitivity coefficients for each dependent variable.
LSODE	Main core integration routine. Checks legality of input, sets work array pointers, initializes work arrays, computes initial integration step size, manages solutions of ODE's for both kinetics problems and sensitivity coefficients, and returns to calling routine with solutions and errors.
CFODE	Sets method coefficients for solution and test constants for local error test and step size and method order selection.
DAXPY	Forms the sum of one vector and another times a constant.
DDOT	Computes dot product of two vectors.
DGBFA	Performs LU-decomposition of a banded matrix by Gaussian elimination.
DGBSL	Solves a linear system of equations using a previously LU-decomposed banded matrix.

TABLE 9.1.—Concluded.

Subprogram	Description
DGEFA	Performs LU-decomposition of a full matrix by Gaussian elimination.
DGESL	Solves a linear system of equations using a previously LU-decomposed full matrix.
DSCAL	Scales a vector by a constant.
DIMACH	Computes unit roundoff of computer.
EWSET	Sets error weight vector.
IDAMAX	Identifies vector component of maximum absolute value.
INTDY	Computes interpolated values of specified derivative of dependent variables.
PREPJ	Computes iteration matrix and either manages subprogram call for its LU-decomposition or computes its inverse.
RSCOM	Restores contents of common blocks LS0001 and EH0001.
SOLSY	Manages solution of linear system arising from chord iteration.
STODE	Advances solution of ODE's for model problem by one integration step. Also, computes step size and method order to be attempted on next step.
SVCOM	Stores contents of common blocks LS0001 and EH0001.
VNORM	Computes weighted root-mean-square norm of a vector.
XERRWV	Handles error messages from LSODE package and sensitivity routines DMACHK, DMINTR, and DMSTOD.
XSETF	Resets print control flag for error messages from XERRWV.
XSETUN	Resets logical unit number for error messages from XERRWV.

use of several temperature ranges (ref. 12). In order to incorporate the new calculation procedure into LSENS, only THRM has to be replaced and suitable modifications made to PEDERV and appropriate common blocks.

Communication between different subprograms is accomplished by means of both call sequences and common blocks, which are used extensively in LSENS. The reason for using common blocks is to avoid lengthy call sequences, which can significantly deteriorate the efficiency of the program. The common blocks, if any, used by each subprogram are given in table 9.2. This table also lists all subprograms called and referenced (e.g., an external function) by each subprogram. Also, in order to facilitate the use of LSENS in overlay situations, all subprograms that call and reference each subprogram are listed. Finally for each subprogram the table gives entry names in parentheses and dummy procedure names (which are passed in call sequences and have to be declared external in each calling and called subprogram) in brackets. Although we have listed only those common blocks that are actually used by the MAIN program, it contains every common block used in the code so that it can be used in overlay situations.

All the common blocks used in the code are listed in alphabetical order in table 9.3. Also given in this table are the variables contained in each common block and their dimensions, if different from unity. To further assist in user understanding and modification of the code, we have included in table 9.3 the names of all subprograms that use each common block. It must be pointed out that not all variables listed for a given common block are needed by each routine that contains it. For this reason some subprograms may use dummy names, which are not listed in table 9.3, and vari-

ables with the same names may be declared to be of different lengths in different routines.

Each subprogram contains type declarations for all variables used in it. Such declarations are useful for debugging and provide a list of all variables that occur in a routine. This list is useful in overlay situations (ref. 23). The type declarations are arranged in a specific order to enhance their utility. At the beginning of each routine the variables are listed in blocks of statements, with each block corresponding to a particular data type. The variable types are listed in the following order: character, logical, integer, real, and double precision. Within each block of data type the variables are usually listed in the following order: variables passed in the call sequence, local array variables, variables appearing in common blocks, and finally local scalar variables. In some of the sensitivity and integration subprograms and in the BLOCK DATA module this order is not followed strictly. Instead, we have placed together blocks that belong to the same program group; however, character variables, if any, are always listed first. For variables that appear in each common block we have, in general, used a separate data-type statement. For a given data type these declarations are listed in the same order as the common blocks. Also, in virtually all data-type declaration statements the variables are listed in alphabetical order.

The variable-type declaration statements are followed by DIMENSION statements. We first list variables included in the call sequence, then variables that appear in namelists, and finally local array variables. Following the DIMENSION statements, if any, all common blocks are listed in alphabetical order. Again in some subroutines this order is not strictly followed, and common blocks that belong to the same program group are placed together. However, within each such

TABLE 9.2.—ROUTINES WITH COMMON BLOCKS, SUBPROGRAMS, AND CALLING SUBPROGRAMS

Subprogram [Dummy procedure name] (Entry names)	Common blocks used			Subprograms called and referenced			Calling subprograms	
MAIN	COND LS0001 MISC OPTS2 SAVRAT SENVAR	DIRECT LS0002 NECC PRIN SENNOR SINT	GEAR9 LTUS ODECON PROP SENPAP	DIFFUN OUT2 SECCPU DFDP INTDY	KINP PEDERV WSR DMPRNT	OUT1 RXNTAB SENDDM SNSOUT		
CIMAGE	LTUS						KINP	
COMB	COND MISC POINTS	INDX NECC SPECES	LTUS OPTS2	ELEMNT SPOUT	EQLBRM	SECCPU	KINP	
CUBS (CINP)	AFUN	COND	LTUS	SPLINE			DIFFUN TINP	KINP
DIFFUN [F] (DIFFW, DIFF1)	COND GHSC ODECON PROP RRAT SPCONC TRAN	CUBASV LTUS OPTS1 RATLOG SABS SPEC2 TRAN2	CUBTMP NECC OPTS2 REAC2 SINT STCS ZERCON	CUBS	HETRAN	THRM	MAIN DMSTOD PREPJ	WSR LSODE STODE
ELEMNT	COND SPECES	INDX	MISC				COMB WSR	SHOK
EQLBRM	COND LTUS NECC SPECES	GHSC MATX OPTS2	INDX MISC POINTS	GAUS	MATRIX	THRM	COMB WSR	SHOCKS
GAUSS	COND	INDX	MATX				EQLBRM	WSR
HETRAN (HTWSR, VISCON)	COND NECC SPEC1 TRAN2	GHSC OPTS1 SPEC2	LTUS PDDTRM TRAN				DIFFUN	KINP
INIT	COND LTUS OPTS1 SINT	FAIR1 NECC OPTS2 SPEC1	FAIR2 ODECON SAVRAT SPEC2				KINP	
KINP (WSFLIP, RINP)	AFUN CUBTMP GHSC INERT2 LS0002 NECC OPTS2 PRIN2 REAC1 ROKET SAVVAR SPECES STCS TSTNOS XVST2	COND ELMNTS INDX KOUT1 LTUS ODECON PDDTRM PROP REAC2 RRAT SENVAR SPEC1 TCOF XVSA1	CUBASV FAIR1 INERT1 KOUT2 MISC OPTS1 PRIN RATLOG RMTHOK SABS SINT SPEC2 TRAN XVSA2	CIMAGE HETRAN THRM	COMB INIT TINP	CUBS SHOK SENSIN	MAIN	

TABLE 9.2.—Continued.

Subprogram [Dummy procedure name] (Entry names)	Common blocks used			Subprograms called and referenced			Calling subprograms	
MATRIX	COND MATX OPTS2	GHSC MISC POINTS	INDX NECC SPECES				EQLBRM	
OUT1	AFUN CUBTMP INERT2 LS0002 ODECON PRIN REAC2 SAVVAR SENPAR SPEC1 XVSA1 XVST2	COND FAIR2 KOUT1 LTUS OPTS1 PROP ROKET SENNAM SENNOR SENVAR SINT STCS TRAN XVSA2 XVST1	CUBASV GHSC KOUT2 NECC OPTS2 REAC1 RRAT SENNOR SINT TRAN XVST1				MAIN	
OUT2 (OUTSTR, OUT3)	AFUN GHSC KOUT2 ODECON PROP SABS SPCONC TRAN	COND INERT2 LTUS OPTS1 REAC2 SAVRAT SPEC1	GEAR9 KOUT1 NECC OPTS2 ROKET SINT SPEC2	RKTOUT	MAIN WSR			
PEDERV [JAC] (PDWSR)	COND MATX OPTS1 PROP SABS TCOF TRAN2	GHSC NECC OPTS2 REAC2 SPCONC TCOF2 ZERCON	LTUS ODECON PDDTRM RRAT STCS TRAN				WSR DMPJAC PREPJ	
RKTOUT	COND	LTUS	ROKET				OUT2	
RXNTAB	COND OPTS1 SPEC1	LTUS REAC2 SPEC2	ODECON SENPAR				MAIN	
SECCPU				System CPU clock	MAIN COMB SHOK WSR			
SHOCKS	COND LTUS POINTS	GHSC MISC SPECES	INDX NECC	EQLBRM	THRM	SHOK		
SHOK	COND NECC	LTUS POINTS	MISC SPECES	ELEMNT SPOUT	SECCPU	SHOCKS	KINP	
SPLINE							CUBS	
SPOUT (ECOUT, ESOUT, FSOUT)	AFUN GHSC LTUS OPTS2 SPEC1	COND INDX MISC POINTS	CUBASV KOUT1 NECC SPECES	THRM			COMB SHOK	

TABLE 9.2.—Continued.

Subprogram [Dummy procedure name] (Entry names)	Common blocks used			Subprograms called and referenced			Calling subprograms	
THRM	COND TCOF	GHSC TCOF2	LTUS				DIFFUN KINP SPOUT	EQLBRM SHOCKS WSR
TINP (TMPPAR)	COND LTUS PRIN SAVTMP XVST1	CUBTMP OPTS1 PRIN2 SAVVAR XVST2	KOUT1 OPTS2 SABS TSTNOS	CUBS			KINP	
WSOUT	COND MISC SPEC1	KOUT1 NECC SPEC2	LTUS PROP TRAN				WSR	
WSR	COND KOUT2 MISC PROP SPEC1	GHSC LTUS NECC REAC2 SPEC2	INDX MATX ODECON SPECES TRAN	DIFFUN GAUSS SECCPU	ELEMNT OUT2 THRM	EQLBRM PEDERV WSOUT	MAIN	
SENDDM	COND DIRECT SENSOL	DFDPA PRIN	DIRCT2 SENPAR	DIFFUN DMACHK	PEDERV DMINIT	DFDP LSODE	MAIN	
DFDP	COND NECC RRAT SENVAR	DFDPA ODECON SABS SPEC2	GHSC OPTS2 SENPAR				DMSTOD	
DMACHK	DIRECT	SENPAR		XERRWV			SENDDM	
DMINIT	DIRECT SENPAR	LS0001	PRIN				SENDDM	
DMINTR	DIRECT SENSOL	LS0001 SENVAR	SENPAR	XERRWV			SNSOUT LSODE	SNSPSC
DMPJAC				DGEFA	PEDERV		DMSTOD	
DMPRNT	DIRECT	EH0001					MAIN	
DMSTOD	DFDPA LS0001 SENPAR	DIRCT2 PRIN	DIRECT SAVRAT	DIFFUN DMPJAC	PEDERV SOLSY	DFDP XERRWV	LSODE	
SENSIN	COND OPTS2 SENPAR	LTUS REAC2 SENVAR	ODECON SENNAM SPEC1				KINP	
SNSOUT (SNSOT2)	DIRECT LTUS PRIN SENNOR SENVAR	GHSC NECC RRAT SENPAR SINT	LS0001 OPTS2 SENNAM SENSOL TSTNOS	DMINTR	SNSPSC	SNSTAB	MAIN	

TABLE 9.2.—Concluded.

Subprogram [Dummy procedure name] (Entry names)	Common blocks used			Subprograms called and referenced	Calling subprograms
VNORM					LSODE PREPJ STODE
XERRWV	EH0001				DMACHK DMINTR DMSTOD LSODE INTDY
XSETF	EH0001				
XSETUN	EH0001				
BLOCK DATA	DIRECT LS0001 OPTS1 TCOF2	EH0001 LTUS REAC1 TSTNOS	KOUT1 NECC SPEC1		

TABLE 9.3.—COMMON BLOCKS WITH VARIABLES AND SUBPROGRAMS

Common block	Variables (dimension)	Subprograms where used
AFUN	LSUBM ETA D VISC BETA SHOCK	CUBS KINP OUT1 OUT2 SPOUT
COND	SIGMA(50) T RHO V DVAR AREA MDOT P IVAR LS LSP1 LSP2 LSP3 NEXT FLPROB	MAIN COMB CUBS DIFFUN ELEMNT EQLBRM GAUSS HETRAN INIT KINP MATRIX OUT1 OUT2 PEDERV RKTOUT RXNTAB SHOCKS SHOK SPOUT THRM TINP WSOUT WSR SENDDM DFDP SENSIN LSODE INTDY PREPJ STODE
CUBASV	CUBX(100) CUBY(100) CUBM(100) CN(4) NTB ITPSZ IPRCOD	DIFFUN KINP OUT1 SPOUT
CUBTMP	CUBXT(100) CUBYT(100) CUBMT(100) CNTMP(4) NTTB ITTSZ ITRCOD	DIFFUN KINP OUT1 TINP
DFDPA	DFDPI(52,750)	SENDDM DFDP DMSTOD
DIRCT2	NFIRST LSCALL	SENDDM DMSTOD
DIRECT	DMHO DMELU(13) JDMOPT JDMJAC ISTRDM JDMJEV JDMADD JDMDEV IDIRC2 NDMSN IDMNR IDMJAC IDMY2 JDMIA NRSTOT NISTOT NSTDM NFEDM NJEDM	MAIN SENDDM DMACHK DMINIT DMINTR DMPRNT DMSTOD SNSOUT SNSPSC LSODE STODE BLOCK DATA
EH0001	MESFLG LUNIT	DMPRNT RSCOM SVCOM XERRWV XSETF XSETUN BLOCK DATA
ELMNTS	ELNAM(15)	KINP
FAIR1	FUEL	INIT KINP
FAIR2	SCC SCH SCOX ERATIO NOXRAT FLAIR ARAT CRAT AIRMW FRO2	INIT OUT1
GEAR9	HUSED NQUSED NSTEP NFE NJE VN PDVAR TDVAR	MAIN OUT2 INTDY STODE

TABLE 9.3.—Continued.

Common block	Variables (dimension)	Subprograms where used
GHSC	DLTX TXR GRT(50) HRT(50) SR(50) CPR(50) DCPR(50)	DIFFUN EQLBRM HETRAN KINP MATRIX OUT1 OUT2 PEDERV SHOCKS SPOUT THRM WSR DFDP SNSOUT SNSPSC
INDX	NLM IQ1 IQ2 IQ3 KMAT IMAT COMBUS HP TP CONVG	COMB ELEMNT EQLBRM GAUSS KINP MATRIX SHOCKS SPOUT WSR
INERT1	DISNM(50)	KINP
INERT2	NINERT	KINP OUT1 OUT2
KOUT1	UNIT TITLE(20) UNITI UNITO FPS SI CGS	KINP OUT1 OUT2 SPOUT TINP WSOUT BLOCK DATA
KOUT2	CONC EXCHR DELH(250) DBUGO	KINP OUT1 OUT2 WSR
LS0001	ROWND CONIT CRATE EL(13) ELCO(13,12) HOLD RC RMAX TESCO(3,12) ELO H HMIN HMXI HU TN UROUND ILLIN INIT LYH LEWT LACOR LSAVF LWM LIWM MXSTEP MXHNIL NHNIL NTREP NSLAST NYH IALTH IPUP LMAX MEO NQNYH NSTEPI IER JSTART KFLAG L METH MITER MAXORD N NQ NST NFE NJE NQU	MAIN DMINIT DMINTR DMSTOD SNSOUT SNSPSC LCODE INTDY PREPJ RSCOM SOLSY STODE SVCOM BLOCK DATA
LS0002	HINIT HMAX HMIN MAXORD MXSTEP MXHNIL	MAIN KINP OUT1
LTUS	LTHM LREAD LWRITE LDAT LTRAP LSCR NTHRD NBLANK NPHOTO	MAIN CIMAGE COMB CUBS DIFFUN EQLBRM HETRAN INIT KINP OUT1 OUT2 PEDERV RKTOUT RXNTAB SHOCKS SHOK SPOUT THRM TINP WSOUT WSR SENSIN SNSOUT SNSTAB INTDY BLOCK DATA
MATX	GA(51,52) GX(51)	EQLBRM GAUSS MATRIX PEDERV WSR
MISC	TT HR0 ENN SUMN ENNL PP CPR0 ELNO(15) NITER	MAIN COMB ELEMNT EQLBRM KINP MATRIX SHOCKS SHOK SPOUT WSOUT WSR
NECC	SSUM MIXMW RATM M2 GAMMA TCPR RERG RCAL	MAIN COMB DIFFUN EQLBRM HETRAN INIT KINP MATRIX OUT1 OUT2 PEDERV SHOCKS SHOK SPOUT WSOUT WSR DFDP SNSOUT SNSPSC BLOCK DATA
ODECON	NRS NRSP1 NRSP2 NSODE NSODP1 NTEQ NRHEQ NVEQ NODES SKIPTR TREQD RHREQD NOJTRH VREQD	MAIN DIFFUN INIT KINP OUT1 OUT2 PEDERV RXNTAB WSR DFDP SENSIN SNSPSC INTDY STODE
OPTS1	VERSI TIMEV VERSA AREAV	DIFFUN HETRAN INIT KINP OUT1 OUT2 PEDERV RXNTAB TINP INTDY STODE BLOCK DATA
OPTS2	RHOCON TASS RXORDR ACONST PCONST TGIVEN TCONST	MAIN COMB DIFFUN EQLBRM INIT KINP MATRIX OUT1 OUT2 PEDERV SPOUT TINP DFDP SENSIN SNSOUT
PDDTRM	PDDSIG(50) PDDT PDDRHO PDDV	HETRAN KINP PEDERV
POINTS	PMLOG DLVTP DLVPT HSUM CCPR GAMMAF WM	COMB EQLBRM MATRIX SHOCKS SHOK SPOUT

TABLE 9.3.—Continued.

Common block	Variables (dimension)	Subprograms where used
PRIN	CPRINT(100) NPRNTS CEND PSTAT PASSV PTEMP IPRINT	MAIN KINP OUT1 TINP SENDDM DMINIT DMSTOD SNSOUT SNSPSC
PRIN2	APRINT(100) PRINT(100) END	KINP TINP
PROP	EIN EOUT VOLUME SI SF RHOI GAMI WM DELMD DELT MASS MPR DOTMAX TMPMIN DTMAX RITE WSFLOW WELSTR WSRHTR	MAIN DIFFUN KINP OUT1 OUT2 PEDERV WSOUT WSR
RATLOG	DAFLOG(250)	DIFFUN KINP
REAC1	EQUAL	KINP OUT1 BLOCK DATA
REAC2	LSR(4,250) XX(250) DPX(2,250) MM(35) RATE(250) BRATE(250) LR LRTYPE(250) LR3RD(35) NS3RD(35) I3RD(10,35)	DIFFUN KINP OUT1 OUT2 PEDERV RXNTAB WSR SENSIN
RMTHOK	RMFINE THFINE	KINP
ROKET	PC ATHROT ROCKET	KINP OUT1 OUT2 RKTOUT
RRAT	A(250) N(250) EACT(250) M(10,35) NUM(250) NTBR ALLM1	DIFFUN KINP OUT1 PEDERV DFDP SNSOUT SNSPSC
SABS	S1 AA BB S2 DTERM TTERM DA DT MWARN	DIFFUN KINP OUT2 PEDERV TINP DFDP
SAVRAT	FF(53)	MAIN INIT OUT2 DMSTOD SNSPSC
SAVTMP	XTB(100) TTB(100) TMPTB(100) CX(4) CT(4) TPRINT(100)	TINP
SAVVAR	CXTB(100) CATB(100) CXTTB(100) CTMPTB(100) IPRSV(4) ITRSAV(2) HT(5) WSRHT(5) ACON PCON TCON	KINP OUT1 TINP
SENNAM	SNAMES(2,53)	OUT1 SENSIN SNSOUT SNSTAB
SENNOR	SIVNP(52) SRPNP(3) YINV(52)	MAIN OUT1 SNSOUT SNSPSC
SENPARR	SCIV(53,52) SCRPP(53,250,3) NPARR(3) IDPAR(2,52) NREAC(250) NRPOUT	MAIN OUT1 RXNTAB SENDDM DFDP DMACHK DMINIT DMINTR DMSTOD SENSIN SNSOUT SNSPSC SNSTAB
SENSOL	DYDY0(52,52) SENSE(53,250,3)	SENDDM DMINTR SNSOUT SNSPSC SNSTAB
SENVAR	TINY SENCAL SENSIV SENSRP SENSJ SENSJ SENSEJ SENSTD SENP SENOUT SNORDR	MAIN KINP OUT1 DFDP DMINTR SENSIN SNSOUT SNSPSC SNSTAB
SINT	TKSAVE EMAX ATOLSP MF MAXSTP NH AVH MMHG MOLEF	MAIN DIFFUN INIT KINP OUT1 OUT2 SNSOUT SNSPSC LODE
SPCONC	C(50) CSUM	DIFFUN OUT2 PEDERV
SPECES	EN(50) ENLN(50) DELN(50) ELSP(15,50)	COMB ELEMNT EQLBRM KINP MATRIX SHOCKS SHOK SPOUT WSR
SPEC1	DSNAM(3) DSPNM(50) EFFM BLANK HNU TAPEND TBSPNM(10,35)	HETTRAN INIT KINP OUT1 OUT2 RXNTAB SPOUT WSOUT WSR SENSIN BLOCK DATA
SPEC2	MW(50) W(50) STOIC(50,250) OMEGA(50,250)	DIFFUN HETTRAN INIT KINP OUT2 RXNTAB WSOUT WSR DFDP

TABLE 9.3.—Concluded.

Common block	Variables (dimension)	Subprograms where used
STCS	NSTOIC(4,250) NSPRP(2,250)	DIFFUN KINP OUT1 PEDERV
TCOF	TC(50,7,2) TLOW TMID THI TPREV	KINP PEDERV THRM
TCOF2	THCX(5) THDCX(3) THHCX(5) THSCX(5) KTHRM	PEDERV THRM BLOCK DATA
TRAN	DD HTRAN TWALL TOTMAS PQMRHO PQMT QDOTM OTTO STROKE RPM BORE QMREAD HTC(5) WSHTC(5) NTR VTC(4,50,2) ICV(50) VCSP(50,2) FDSQ2 IH2O KOUNT	DIFFUN HETRAN KINP OUT1 OUT2 PEDERV WSOUT WSR
TRAN2	GAMM1 GM1DG	DIFFUN HETRAN PEDERV
TSTNOS	SMEST TESTNO NLMAX LSMAX LRMAX LSRMAX TBRMAX TBSMAX NTBMAX NPMAX	KINP TINP SNSOUT SNSPSC BLOCK DATA
VELDOT	TOLD VDOT	INTDY STODE
XVSA1	XU AU(2)	KINP OUT1
XVSA2	XTB(100) ATB(100) CX(4)	KINP OUT1
XVST1	XTU TU	OUT1 TINP
XVST2	XTTB(100) TMPTB(100) CTMP(4)	KINP OUT1 TINP
ZERCON	FBRATE(2,250) LZEROC	DIFFUN PEDERV

set the common blocks are listed alphabetically. The common block list is followed by equivalence statements, if any, and then by any namelists. Within each namelist the variables are ordered alphabetically. However, the namelists themselves are listed in the order in which they are read. Following the namelists are DATA statements for local variables and finally any functions that are internal to the routine.

The remainder of this chapter describes the code. To facilitate description of the many functions and options built into it, we separate the code into seven major sections, where each section performs a different task. These sections are as follows: control, input/output, numerical integration procedure, sensitivity analysis, chemical equilibrium computations, incident shock computations, and perfectly stirred reactor computations. In addition, thermodynamic properties, heat transfer rates, and transport properties are computed in separate sections, by using the procedures described in chapter 8.

For each of the seven sections we first describe its function and then, because they have already been described in table 9.1, only list the subprograms used. The discussion is focused mainly on the special features and built-in options. However, we provide detailed flowcharts to explain the computational procedures; in addition, structural diagrams are

given for each section. We conclude this chapter with a brief discussion of the error messages included in the code.

9.2 Control

The control section consists of the MAIN program and, as discussed in later sections, to some extent the subroutines COMB, SHOK, SENDDM, and LSOE. The MAIN program provides for problem setup through a series of subroutine calls. It calls for input and output of the reaction mechanism and other problem options. It also calls the appropriate subroutine, WSR or SENDDM, depending on the problem type—perfectly stirred reactor (PSR) or kinetics/sensitivity analysis. For the latter type of problem the MAIN program also manages the calls for solution output and computes and prints the CPU times.

The structure of the LSENS code is given in figure 9.1, wherein a line connecting two routines indicates that the lower routine is called by the upper one and the names in parentheses are ENTRY names. This notation is used in all the structural diagrams presented in this chapter. The dashed lines connecting the routine LSOE with XERRWV and DMINTR indicate that other routines not shown in this figure are also called by LSOE. We do not include the other

9. Code Organization and Description

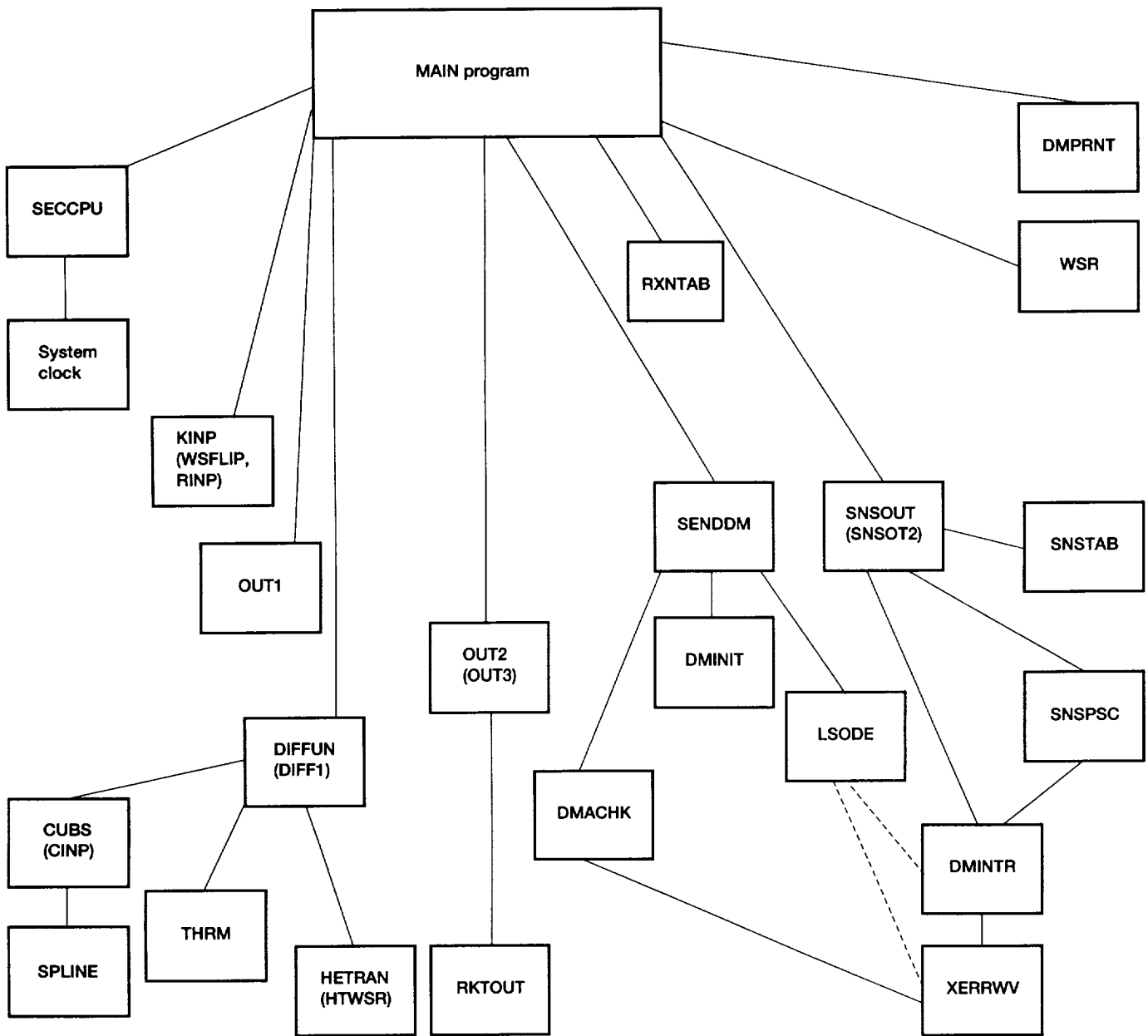


Figure 9.1.—Structure of LSENS code.

routines here because we take up the structure of the LSODE package in a later section (9.4). Figure 9.1 gives all the routines called by SENDDM because it provides some control of the integration. The subroutines called by DIFFUN are also included because of its importance.

A detailed flowchart of the MAIN program is given in figure 9.2. An important feature of LSENS is its capability to perform multiple cases in a single run. This option is most

useful when several runs have to be made with the same reaction mechanism but with different problem types or initial conditions. Another option that controls the integration concerns the printout of results. The solution is generated either at intervals of a prescribed number of integration steps or at the output stations $\xi_{out,1}, \xi_{out,2}, \dots$, where ξ is the integration variable (i.e., the independent variable for the ODE's).

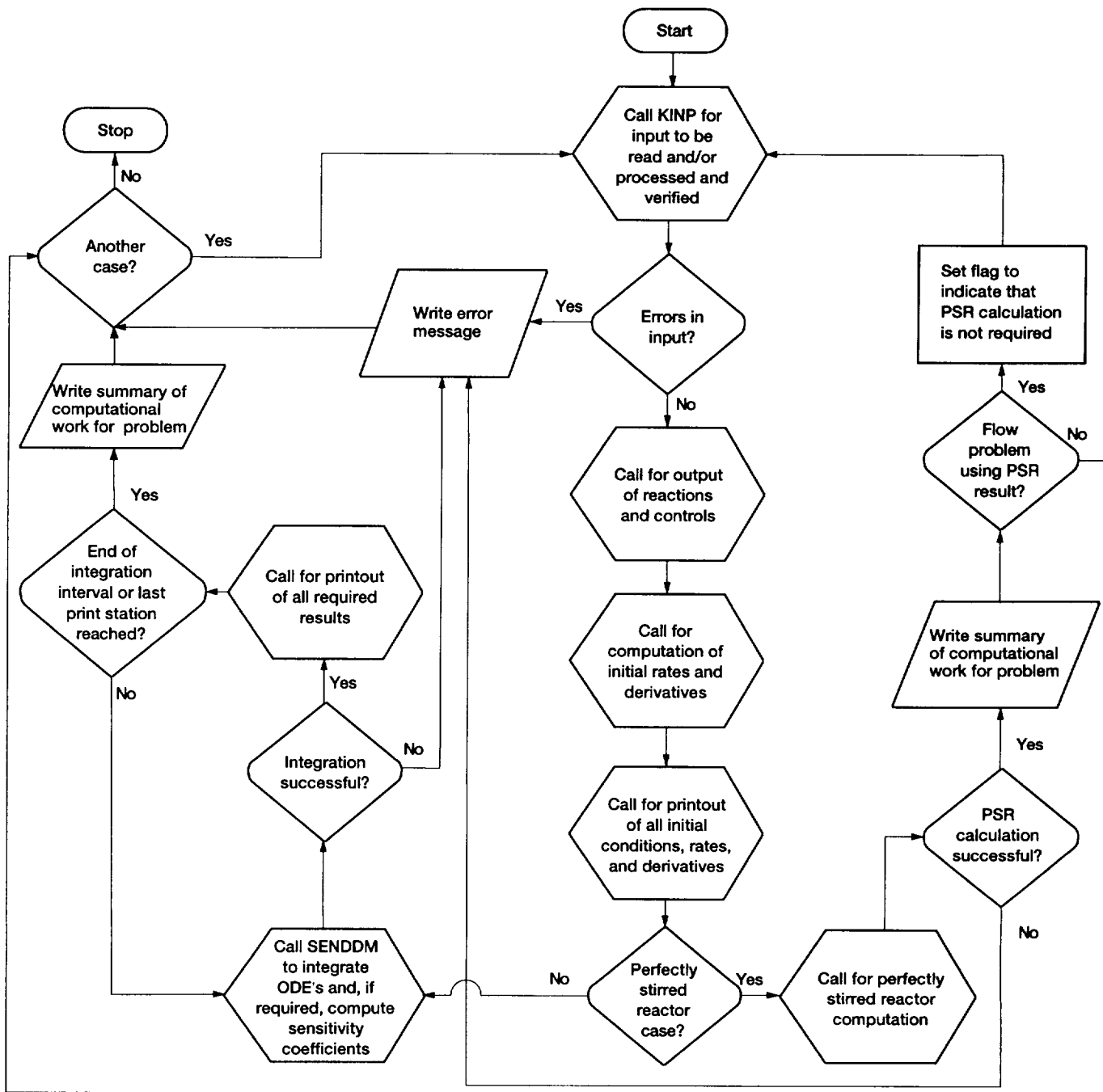


Figure 9.2.—Flowchart of MAIN program.

9.3 Input/Output

The input/output section is called for all problem types. It reads, processes, and checks the legality of all initial conditions and all input data, except those concerned with sensitivity analysis. It also prints all input data and initial conditions and prints and checks the validity of solutions generated by the numerical integration method. In addition,

an option is available to tabulate and print reaction numbers in the order of decreasing importance for each species and to print all nonzero production rates of every species by each reaction. The subprograms that make up this section are CIMAGE, INIT, KINP, OUT1, OUT2, RKTOUT, RXNTAB, TINP, and BLOCK DATA. The routines related to output for sensitivity analysis, equilibrium, incident shock, and PSR calculations are considered in later sections.

9. Code Organization and Description

The most important subroutine in this section is KINP, whose structure and flowchart are given in figures 9.3 and 9.4, respectively. This subroutine initializes various parameters, sets standard options, and reads all options and input data, except those specified for sensitivity analysis and the temperature profile for an assigned-temperature problem. The code has built-in standard choices for many of its options to minimize the amount of input data required. For the same reason, for a repeat case most of the data specified for the previous problem are saved.

For an assigned-temperature problem KINP calls TINP to process the temperature profile and other optional inputs. For example, LSENS includes an option whereby output can be required at specified values of the temperature, if it is assigned in tabular form as a monotonic function of the integration variable. In this situation TINP calls CUBS to compute corresponding values of the integration variable.

Another option that is included in KINP is that the rate coefficient for any third-body collisional reaction may be adjusted to account for the different efficiencies of different third-body species. The standard choice, which is automatically set in KINP, is a collisional efficiency of unity. Thus only collisional efficiencies that are different from unity need to be specified. For each third-body reaction KINP builds up a table of third-body species and their collisional efficiencies that are different from unity. Any species for which an efficiency value of unity is given is deleted from this table.

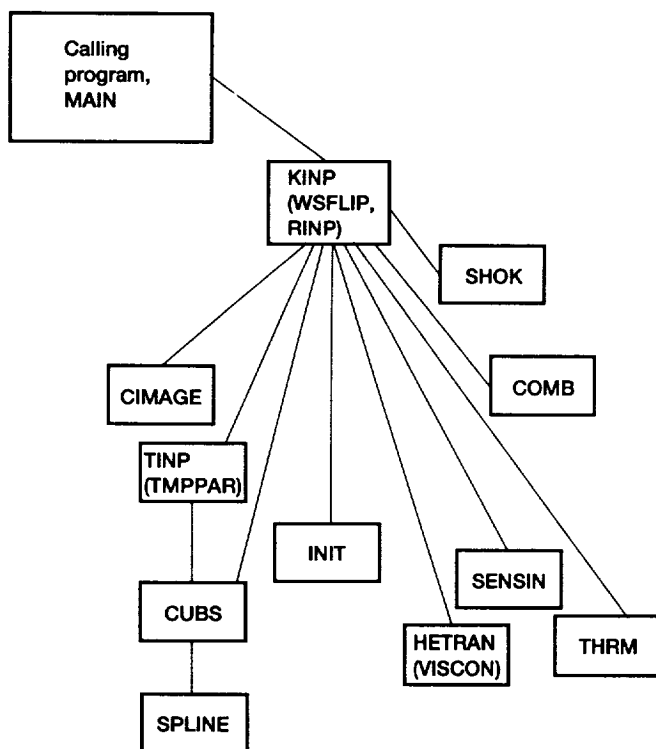


Figure 9.3.—Structure of subroutine KINP.

KINP automatically builds up lists of reacting species and their constituent elements (atoms, etc.) as it processes each reaction. Therefore the user need not specify separate lists of elements and reacting species. A list of inert species is, however, required. Also in specifying third-body species names there is no requirement that this species has already appeared in a reaction. However, after all reactions and inert species names have been processed, the routine verifies that every third-body species has indeed appeared either as a reacting or an inert species. In addition, the routine checks that the user has not specified a noncatalytic reacting species as inert.

KINP also confirms that third-body efficiencies have been specified only for a third-body reaction and that the efficiency list follows the reaction. If the list is inserted before the reaction, the routine will give an error message and stop execution. However, if a third-body reaction precedes the one for which the list is intended, such testing is impossible, unless a third-body efficiency list had already been processed for it.

KINP includes an option to check the legality of the reaction mechanism. When this option is selected, the code verifies that no reaction is duplicated and that each reaction satisfies charge and atom balance requirements. For example, reaction duplication may arise because the same reaction is written in different forms in different regions of the input file. However, we feel that such testing, which adds to the expense of running the code, is only necessary when the user is either developing or preparing a new mechanism. For this reason we have made it optional, and for a repeat case that uses exactly the same mechanism as the previous case the option is switched off by KINP (even if it had been selected for the previous problem). However, in order to avoid an error exit for apparently inexplicable reasons, for each repeat case KINP checks that the formerly specified mechanism was legal. If, for any reason it had been found to be illegal, a message stating this fact is printed and the execution is halted.

The same two actions (i.e., printing an error message and terminating execution) are taken by KINP for a repeat case if either illegal or insufficient thermodynamic data had been supplied for the previous problem.

The final test of the reaction mechanism ensures that no superfluous reaction has been included. Such a condition arises when some reactions contain species that can neither be formed nor destroyed by the mechanism and so will always have forward and reverse (and hence net) rates of zero. An example is the modeling of the H_2-O_2 system with an $H_2-O_2-N_2$ mechanism. This problem may also arise because some elements of reacting or inert species have not been initialized (to a nonzero concentration). The error is not considered fatal; a message describing the problem will be printed, but the execution will not be terminated.

Following this testing of the reaction mechanism, KINP identifies the problem type, checks the legality and sufficiency of all input data, and converts them to internal (cgs)

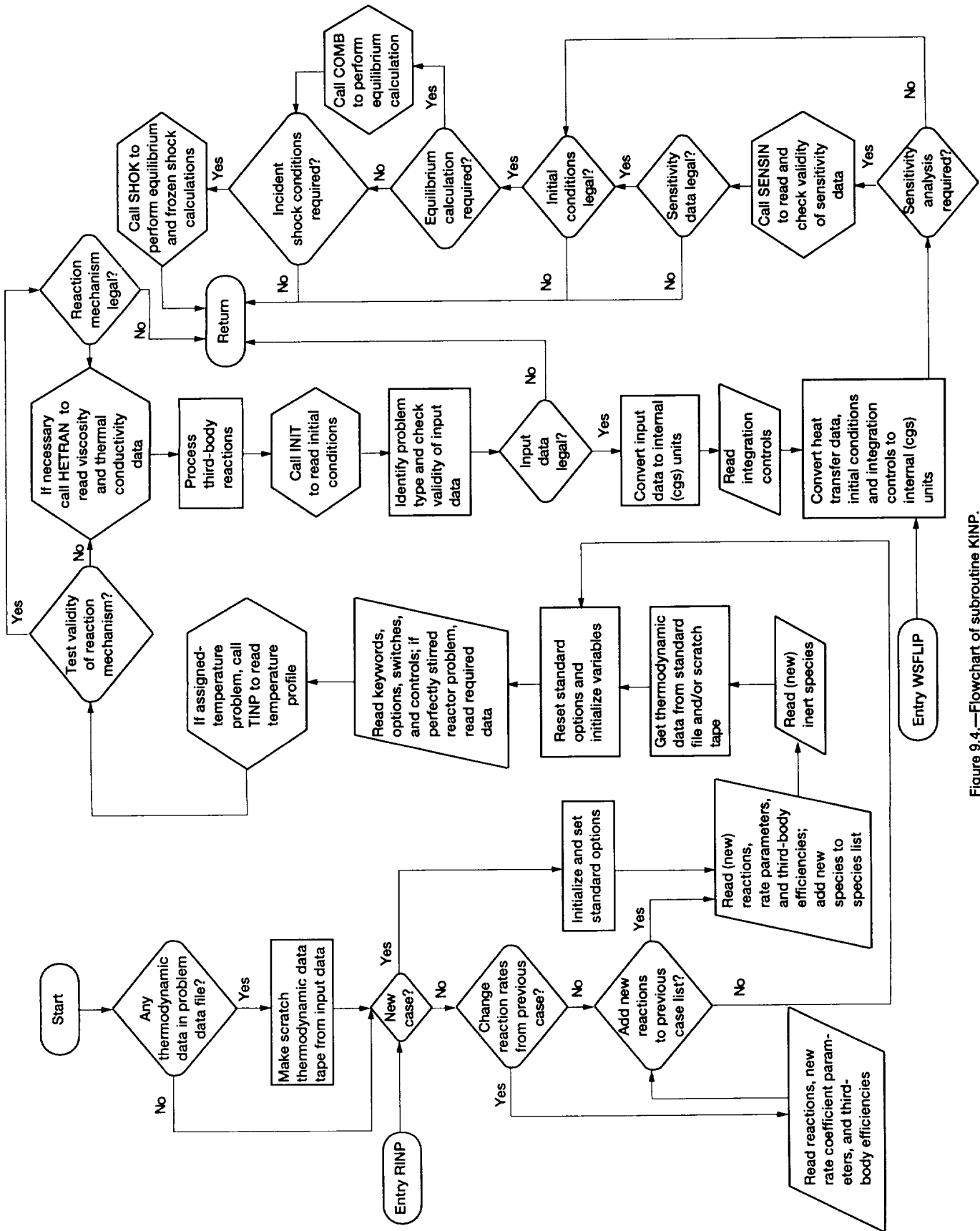


Figure 9.4.—Flowchart of subroutine KINP.

9. Code Organization and Description

units. A choice of three systems of units, cgs, U.S. customary, and SI, is provided for both input and output. The choices for input and output units are independent of one another, so that either the same or different units for input and output can be selected. Output may be required at specified values of the assigned variable (pressure or area) if it is given in tabular form as a monotonic function of the integration variable. In such a case KINP calls CUBS to compute corresponding values of the integration variable.

An important function of KINP is that for a kinetics-only problem (i.e., no sensitivity analysis) it sets the number N of ODE's that must be solved. The ODE set contains ODE's for only those variables that are required for a given problem and vary during the course of the integration. Thus we do not solve ODE's for variables that are not required (e.g., velocity for a static problem), that are specified (e.g., temperature for an assigned-temperature problem), and that are constant (e.g., inert species mole numbers). The ODE set therefore contains the minimum number of ODE's required to solve the problem. Not including the preceding variables increases the efficiency of the ODE solution for two reasons: (1) it avoids unnecessary calculations of derivatives and Jacobian matrix elements and (2) the system is smaller, thereby decreasing the cost of both lower-upper (LU)-decomposing (e.g., ref. 6) the iteration matrix (see eq. (9.3)) and solving for the corrections at each iteration.

If sensitivity analysis is required, KINP calls the routine SENSIN, which processes all options and input data relevant to sensitivity analysis and is described in section 9.5.

Finally KINP processes the initial conditions and, if necessary, converts them to internal (cgs) units. It also checks the legality and sufficiency of the initial conditions. Moreover, it checks that the problem is not overspecified. If a chemical equilibrium computation is required, KINP calls subroutine COMB, which manages such calculations. If postshock conditions are required, KINP calls subroutine SHOK, which manages these calculations.

9.4 Numerical Integration Procedure

LSENS uses the double-precision version (dated June 17, 1980) of the packaged code LSODE (refs. 20 to 22) to solve the stiff ODE's arising in combustion chemistry. LSODE includes a variable-step, variable-order implicit Adams method (suitable for nonstiff problems) of orders 1 to 12 and a variable-step, variable-order backward differentiation formula method (suitable for stiff problems) of orders 1 to 5. The user may, however, specify a smaller value than used in the code for the maximum order to be attempted on any step.

Irrespective of the solution method the code starts the integration with a first-order method and, as the integration proceeds, automatically adjusts the method order (and step size) for optimal efficiency while satisfying prescribed

accuracy requirements. Both integration methods employ a predictor-corrector scheme, wherein on each step $[\xi_{n-1}, \xi_n]$ an initial guess $\underline{Y}_n^{[0]}$ for the solution vector \underline{Y}_n at ξ_n is first produced and then the guess is improved upon by iteration. That is, improved estimates $\underline{Y}_n^{[m]}$ ($m = 1, 2, \dots$) are computed until the iteration converges. A standard explicit predictor formula, a Taylor series expansion method devised by Nordsieck (ref. 24), is used to generate $\underline{Y}_n^{[0]}$. A range of iteration techniques for correcting this estimate is included in LSODE. Both the basic integration method and the corrector iteration technique are selected by means of the method flag MF. By definition, MF has the two decimal digits METH and MITER, and

$$MF = 10 \times \text{METH} + \text{MITER} \quad (9.1)$$

In this equation the integers METH and MITER indicate, respectively, the integration method and the corrector iteration technique to be used for the problem. Table 9.4 summarizes the integration methods included in LSODE and the appropriate values for METH. The legal values for MITER and their meanings are given in table 9.5. The Jacobian matrix \mathbf{J} referred to in this table is an $N \times N$ matrix, with element J_{ij} defined as

$$J_{ij} = \partial f_i / \partial y_j, \quad i, j = 1, \dots, N \quad (9.2)$$

TABLE 9.4.—SUMMARY OF INTEGRATION METHODS INCLUDED IN LSODE AND CORRESPONDING VALUES OF METH

METH	Integration method
1	Variable-step, variable-order implicit Adams method of orders 1 to 12
2	Variable-step, variable-order implicit backward differentiation formula method of orders 1 to 5

TABLE 9.5.—CORRECTOR ITERATION TECHNIQUES AVAILABLE IN LSODE AND CORRESPONDING VALUES OF MITER

MITER	Corrector iteration technique
0	Functional iteration
1	Modified Newton iteration with user-supplied analytical Jacobian matrix
2	Modified Newton iteration with internally generated numerical Jacobian matrix
3	Modified Jacobi-Newton iteration with internally generated numerical Jacobian matrix
^{a,b} 4	Modified Newton iteration with user-supplied banded Jacobian matrix
^a 5	Modified Newton iteration with internally generated banded Jacobian matrix

^aThe user must supply the lower (ML) and upper (MU) half-bandwidths of the Jacobian matrix (ref. 22).

^bThis option should not be used with the present version of LSENS.

where y_i is the i th ($i = 1, \dots, N$) dependent variable and $f_i = dy_i/d\xi$.

The LSODE package consists of the main core integration routine, also called LSODE, and the 21 subprograms CFODE, DAXPY, DDOT, DGBFA, DGBSL, DGEFA, DGSLS, DSCAL, D1MACH, EWSET, IDAMAX, INTDY, PREPJ, RSCOM, SOLSY, STODE, SVCOM, VNORM, XERRWV, XSETF, and XSETUN. Of these subprograms LSENS makes no use of the routines RSCOM, SVCOM, XSETF, and XSETUN. The structure of the LSODE package is given in figure 9.5, wherein the dashed lines indicate the modifications made to this code to compute sensitivities and the names in brackets are dummy procedure names.

The main routine LSODE controls the integration and serves as an interface between the calling subprogram and the rest of the package. Its flowchart is given in figure 9.6, where ITASK and ISTATE are user-supplied indices that specify, respectively, the task to be performed and the state of the calculation, that is, if the call to LSODE is the first

one for the problem or if it is a continuation (ref. 22). It must be pointed out that the figure shows only those LSODE options used by LSENS. On return from LSODE the value of ISTATE indicates if the integration was performed successfully, and if not, the reason for failure. The variable TOUT is the next ξ value at which output is required. Finally JSTART is an internally defined variable used for communicating the state of the calculation with subroutine STODE.

An important feature of LSODE is that it will compute the step size to be attempted on the first step if the user chooses not to provide a value for it. Another useful feature is that different integration methods can be used in different sub-intervals of the problem. For example, the heat release period, especially the early part, is not stiff (refs. 25 to 30), and it may be more efficient to switch to a nonstiff method in this regime (ref. 31). The code LSENS does not exploit this feature because of the lack of reliable regime identification tests, and the same method is used for the whole problem.

The routine STODE advances the solution to the ODE's by a single integration step. In addition, it computes the method order and the step size that together maximize efficiency while maintaining prescribed accuracy. To increase the efficiency of the calculation procedure, the solution history, which is required by the multistep methods used in LSODE (ref. 22), is saved in the form suggested by Nordsieck (ref. 24). The $N \times (q_n + 1)$ Nordsieck history matrix z_{n-1} at ξ_{n-1} contains the numerical solution \underline{Y}_{n-1} and the q_n scaled derivatives $h_n^{(j)} \underline{Y}_{n-1}^{(j)} / j!$ ($j = 1, \dots, q_n$), where $h_n (= \xi_n - \xi_{n-1})$ and q_n are the current step size and method order, respectively, and $\underline{Y}^{(j)} = d^j \underline{Y} / d\xi^j$. The flowchart of STODE, again as used by LSENS, is presented in figure 9.7, which essentially illustrates how the history matrix is advanced over the step $[\xi_{n-1}, \xi_n]$. In this figure NCF is the number of corrector convergence failures on the current step, KFLAG is an internally defined flag used for communication with subroutine LSODE, $z_n^{[0]}$ is the predicted history matrix at ξ_n , and \mathbf{P} , an $N \times N$ iteration matrix that arises from Newton iteration, is given by

$$\mathbf{P} = \mathbf{I} - h\beta_0 \mathbf{J} \tag{9.3}$$

where \mathbf{I} is the $N \times N$ identity matrix and β_0 is a method coefficient that depends on the integration method and the method order (ref. 22). The vector $\underline{\ell}$ contains the integration method coefficients, and $\ell_0 (= \beta_0)$ is the zeroth component of $\underline{\ell}$. The integer counter IALTH indicates how many more steps are to be taken with the current step size H and method order NQ, and HMIN and HMAX are user-supplied minimum and maximum absolute values of the step size to be attempted on any step. The ratios RHDN, RHSM, and RHUP are factors by which H can be increased if the new method order is $NQ - 1$, NQ (the current value), and $NQ + 1$, respectively. R is the ratio of the step size to be attempted next to its current value, and RMAX is the maximum R allowed when a step size change is next considered. Finally NQMAX is the

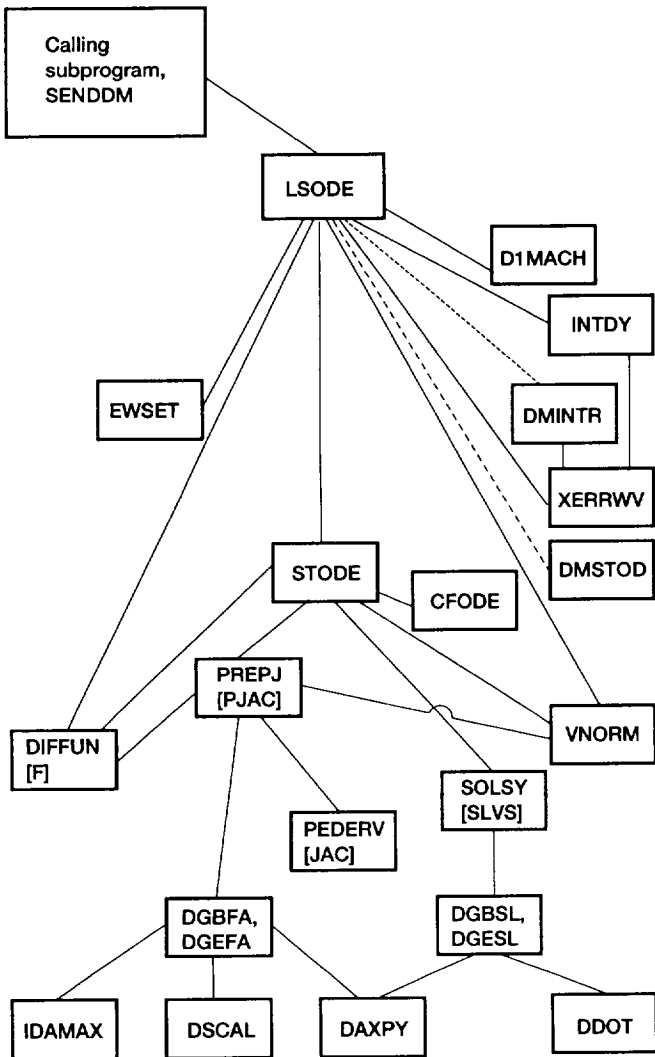


Figure 9.5.—Structure of LSODE package (adapted from reference 22).

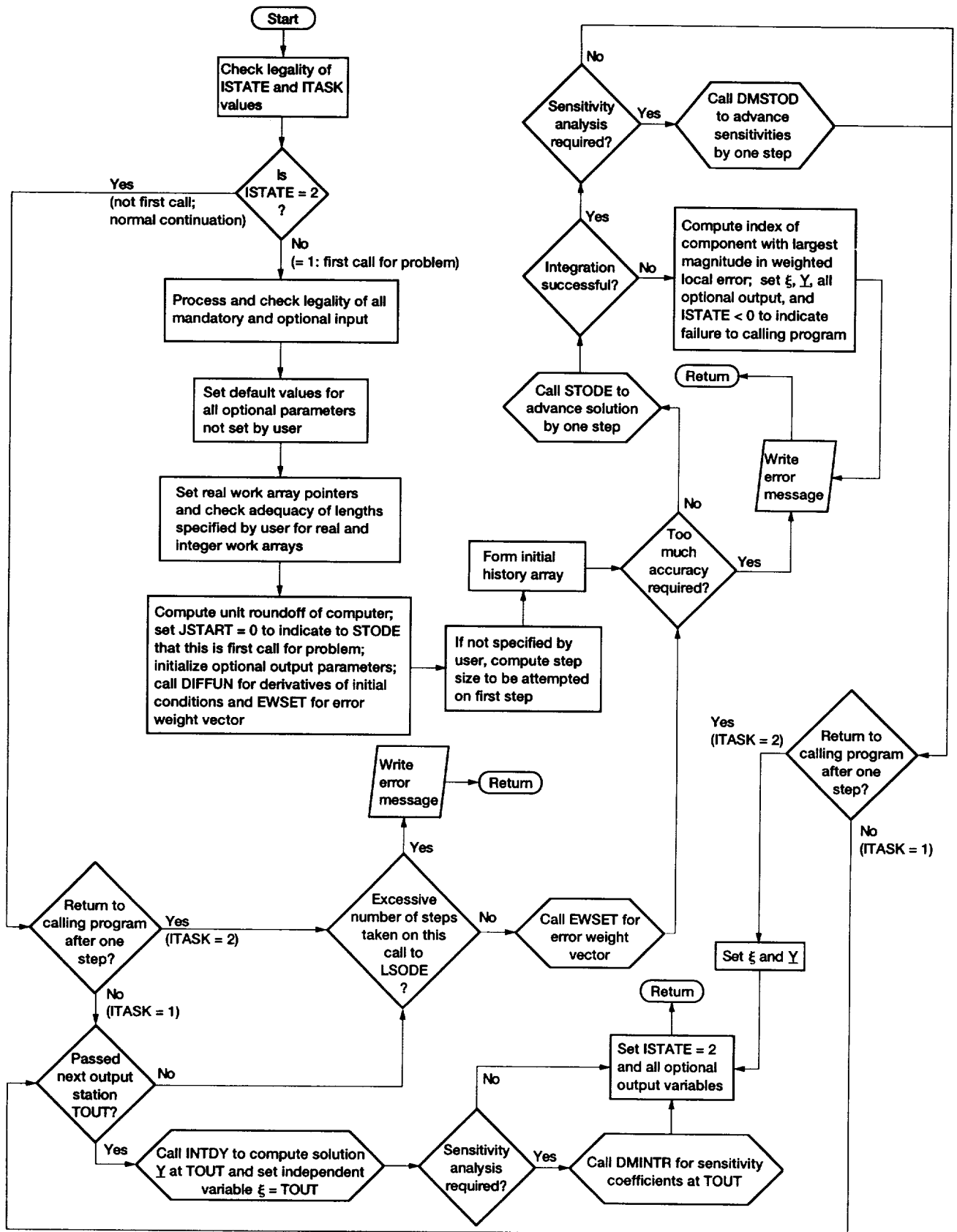


Figure 9.6.—Flowchart of subroutine LSODE, as used by LSENS (adapted from reference 22).

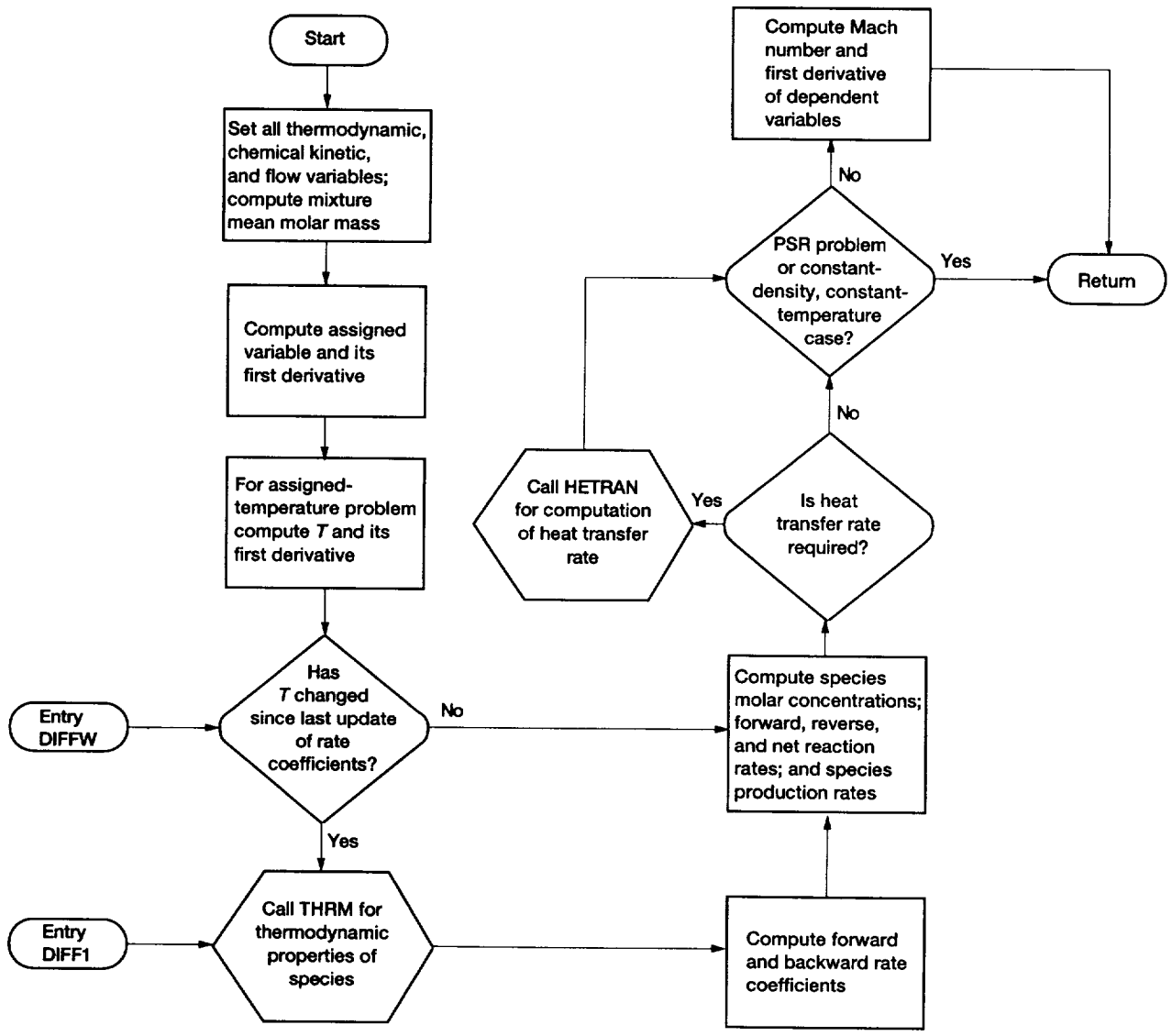


Figure 9.8.—Flowchart of subroutine DIFFUN.

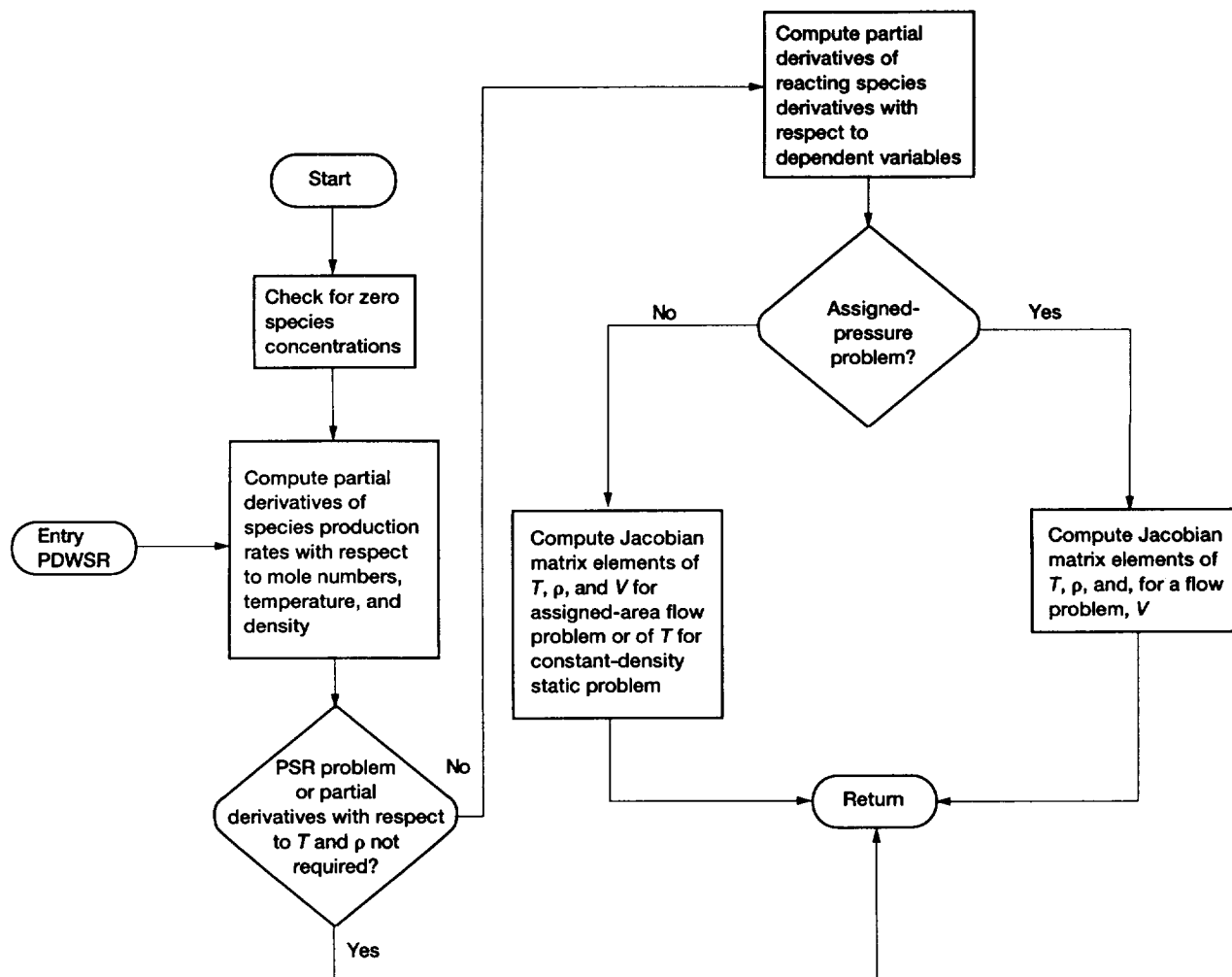


Figure 9.9.—Flowchart of subroutine PEDERV.

maximum method order that may be attempted on any step, and $\underline{\epsilon}_n (= h_n \dot{Y}_n - h_n \dot{Y}_n^{[0]})$ is proportional to the local truncation error vector.

The two user-supplied routines used by the integrator are DIFFUN, which computes the derivatives, and PEDERV, which computes the Jacobian matrix. The structure of DIFFUN is given in figure 9.1. Flowcharts for DIFFUN and PEDERV are presented in figures 9.8 and 9.9, respectively. In these figures ρ and V are the mixture density and velocity. To maintain the accuracy of the Jacobian matrix and to prevent overflow errors, PEDERV checks for species with zero concentration and includes special calculation procedures for such a situation.

9.5 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 ($i = 1, \dots, N$) dependent variable, and η_j is either an initial condition value or a rate coefficient parameter (i.e., A_j , n_j , E_j , or c_j , see eqs. (8.3) and (8.4)). The sensitivity analysis computations use the decoupled direct method (refs. 30, 32, and 33), as implemented by Dunker (ref. 34) for isothermal problems and extended by Radhakrishnan (ref. 33) to nonisothermal combustion kinetics. An important feature of LSENS is that it can

9. Code Organization and Description

be used to generate any number of sensitivity coefficients, from just one initial condition or one rate coefficient parameter of one reaction to the full set of all N initial conditions and all $3NR$ rate coefficient parameters, where NR is the total number of reactions. Finally the linear sensitivity coefficients of the temporal derivatives of the dependent variables (i.e., $\{\partial \dot{Y}_i / \partial \eta_j\}$) can also be computed.

This section of the program reads, processes, and checks the legality of all input data, solves for the sensitivity coefficients, and normalizes and prints them when output is required. Provision is made for the user to specify a cutoff level, $TINY$, for the normalized sensitivity coefficients. Any normalized sensitivity coefficient that is smaller in magni-

tude than $TINY$ is set equal to zero. For rate coefficient parameters an option to tabulate and print nonzero normalized sensitivity coefficients in decreasing magnitude is provided. The list is produced for each dependent variable. The corresponding reaction number is placed above each normalized sensitivity coefficient. Thus the user has the convenient option of obtaining a list of reaction numbers in order of decreasing importance for each variable. Finally, if required, this section computes, normalizes, prints, and if necessary, tabulates in order of decreasing importance, the sensitivity coefficients of the temporal derivatives of the variables.

The 12 subroutines that make up this section are SENDDM, DFDP, DMACHK, DMINIT, DMINTR,

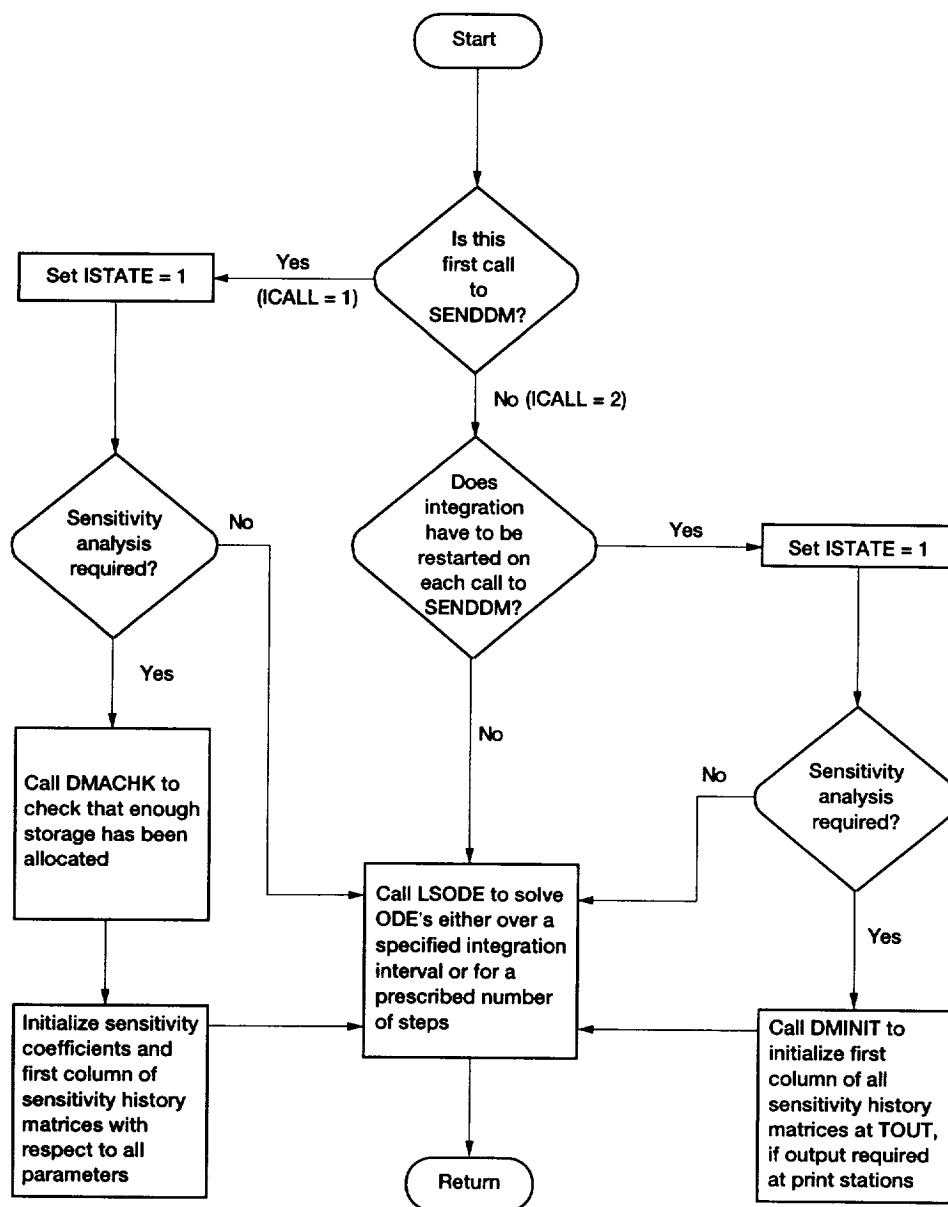


Figure 9.10.—Flowchart of subroutine SENDDM.

DMPJAC, DMPRNT, DMSTOD, SENSIN, SNSOUT, SNSPSC, and SNSTAB. Among these, six routines (SENDDM, DMACHK, DMINIT, DMPJAC, DMPRNT, and DMSTOD) were adapted from the May 1984 version of the code CHEMDDM (ref. 34). Other subprograms that are required for sensitivity analysis include DIFFUN, PEDERV, DAXPY, DDOT, DGEFA, DGESL, DSCAL, IDAMAX, SOLSY, XERRWV, and those used by DIFFUN.

Besides reading, processing, and checking the legality of all input, SENSIN sets the number of ODE's to be solved. In addition to the ODE's for nonconstant quantities SENSIN examines the initial condition parameters with respect to which sensitivities are required. For example, if sensitivity coefficients with respect to the initial temperature are required, the Jacobian matrix elements of all variables with respect to the temperature must be computed. Therefore, even if the temperature is constant, the routine includes it in the ODE list. Similar remarks apply to the density and inert species mole numbers.

The main subprogram in this section of the code is SENDDM, whose structure and flowchart are given in figures 9.1 and 9.10, respectively. The internally defined integer variable ICALL in figure 9.10 denotes if the call to SENDDM is the first one for the problem or a continuation. This routine sets the sensitivity arrays at the initial time and calls DMACHK, which checks that sufficient storage has been allocated for the problem. SENDDM also manages the call to LSODE. LSENS includes an option that dictates how the integration is to be continued after every printout of the solution: either normally, so that LSODE uses past solution values in further developing the solution, or by reinitializing LSODE so that it effectively solves a new problem after every printout. If the second option is selected, SENDDM calls DMINIT to initialize the sensitivity arrays and sets the appropriate index that causes LSODE to reinitialize the integration process.

The routine DMSTOD advances the sensitivity arrays by one time step. The integration method used to solve for the sensitivities is the backward differentiation formula, as implemented in LSODE. Predicted values are first generated by using the Taylor series expansion method devised by Nordsieck (ref. 24). They are then corrected with the backward differentiation formula method in conjunction with a single Newton-Raphson iteration, using an analytical Jacobian matrix. At each step DMSTOD uses exactly the same step size and method order as those used by STODE to integrate the model problem. As for the model problem the solution history for sensitivity coefficients is maintained in the Nordsieck history array. Thus for the j th parameter η_j the $N \times (q_n + 1)$ matrix $Z_{j,n}$ contains $S_{j,n}$, the sensitivity coefficient vector with respect to η_j , and its q_n scaled derivatives at t_n .

The structure of DMSTOD is shown in figure 9.11, where the dashed lines connect subroutines included in, or required by, the LSODE package and the names in brackets are dummy procedure names. This figure shows that the only

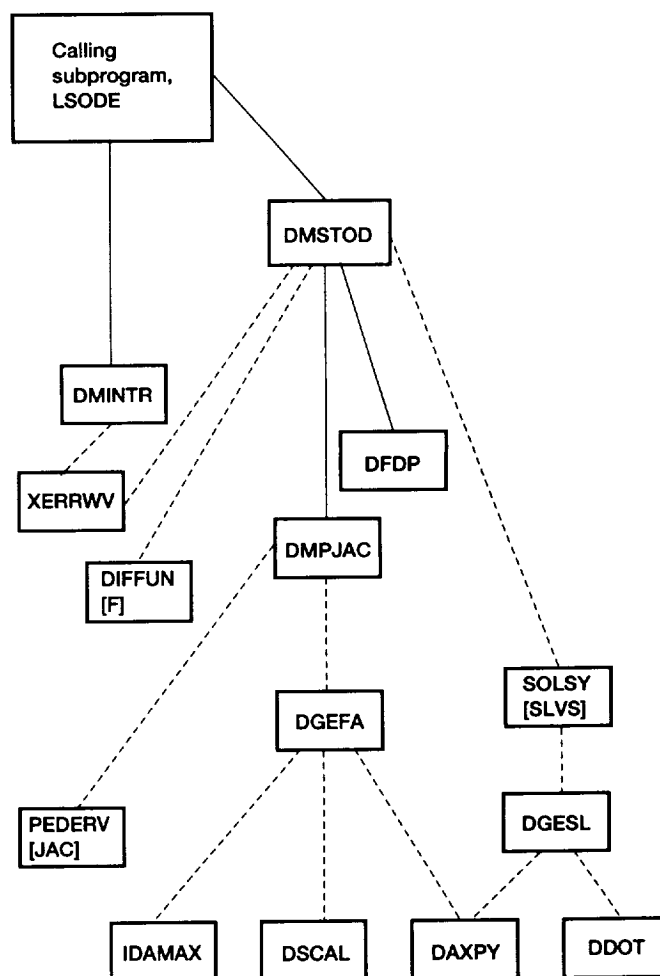


Figure 9.11.—Structure of subroutine DMSTOD.

additional subprogram needed by the decoupled direct method to compute sensitivity coefficients is DFDP. DMPJAC is a simplified version of PREPJ (table 9.1). The former routine assumes that a specific iteration method (Newton-Raphson with a user-supplied analytical Jacobian matrix) is used; the latter is more generalized. DMPJAC can be replaced with PREPJ, but to do so would reduce the efficiency marginally. DMINTR is based on INTDY (table 9.1) and computes the sensitivities and their derivatives at the output stations.

The flowchart of DMSTOD is presented in figure 9.12, which illustrates how the Nordsieck history matrices for the sensitivity coefficients are advanced over the step $[t_{n-1}, t_n]$. In this figure $Z_{j,n}^{[0]}$ is the predicted history matrix for the j th parameter at t_n , \mathbf{P} is given by equation (9.3), and the vector $Z_{j,n-1}(1)$ contains the scaled first derivative of $S_{j,n-1}$ (i.e., $h_n \dot{S}_{j,n-1}$). LSENS contains an option that controls how often the Jacobian matrix is updated when solving for the sensitivity coefficients: either on every step or only on those steps for which STODE performed this update for the model problem.

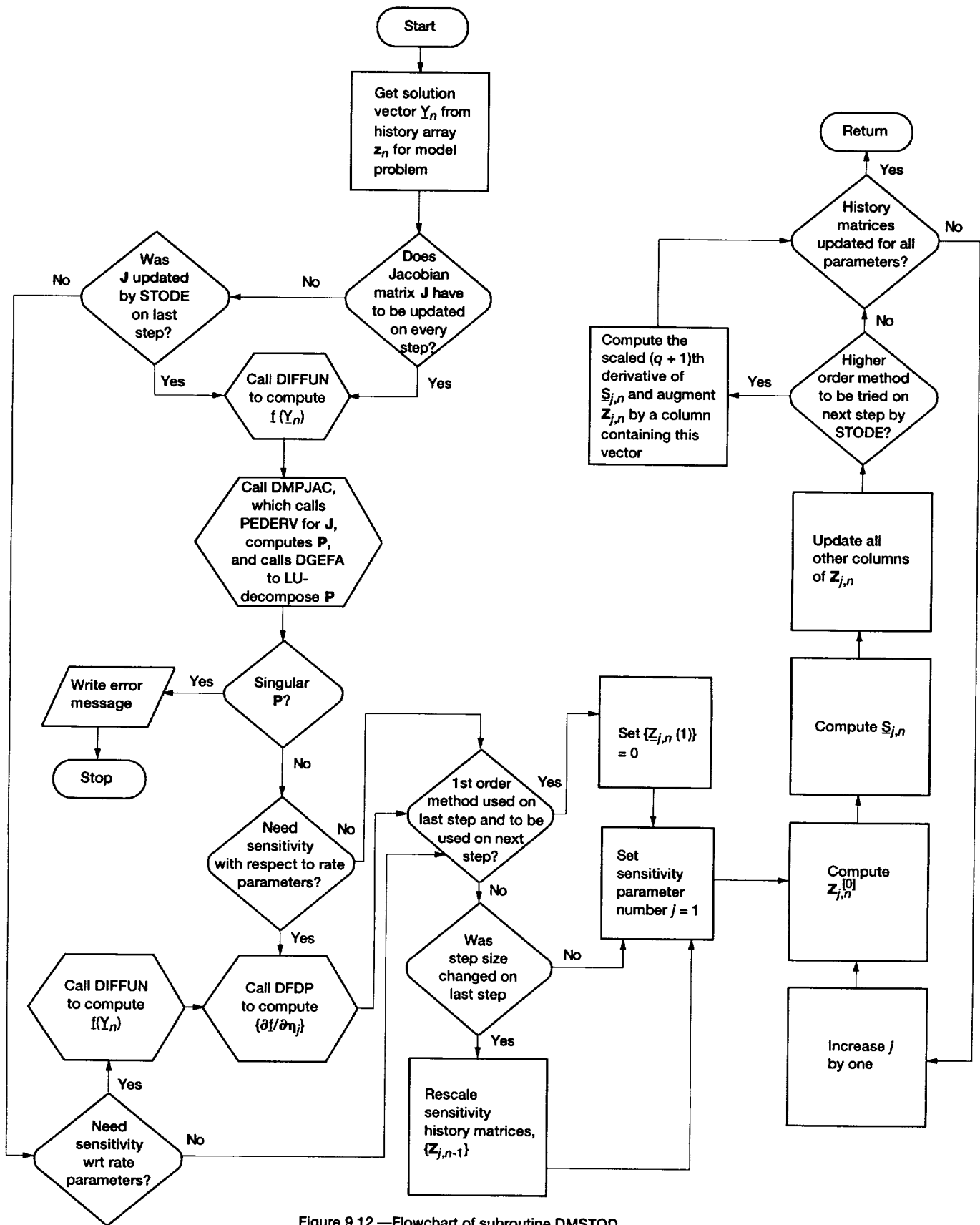


Figure 9.12.—Flowchart of subroutine DMSTOD.

9.6 Chemical Equilibrium Computations

The code has built-in procedures for computing the equilibrium composition for the following four assigned states: (1) temperature and pressure (TP), (2) mass-specific enthalpy and pressure (HP), (3) temperature and specific volume (TV), and (4) mass-specific internal energy and specific volume (UV). For cases 2 and 4 the equilibrium temperature is also determined. The code automatically performs the appropriate type of equilibrium calculation, as discussed here.

The main subprogram in this section is COMB, which manages the equilibrium computation, calls for output of results, and prints the computational work required. The other routines used in this section, whose structure is given in figure 9.13, are ELEMNT, EQLBRM, GAUSS, MATRIX, SECCPU, SPOUT, and THRM. COMB examines the variable TCONST, which is set in KINP and indicates whether the kinetics problem following the equilibrium calculation is at constant temperature, and sets the appropriate switch,

TP or HP, to indicate to EQLBRM whether an assigned-temperature equilibrium computation is required. It also initializes the estimates for the equilibrium mixture composition and, if necessary, the equilibrium temperature.

The equilibrium computations are performed in EQLBRM, which was adapted from the code CET (ref. 11). The routine selects the equilibrium problem type by examining the switches HP, TP, and RHOCON, which indicates if a constant-density kinetics problem follows the equilibrium computation. The equilibrium state is obtained by minimizing either the Gibbs or Helmholtz function. The resulting algebraic equations are solved by using a descent Newton-Raphson iteration method (refs. 10 and 11), which automatically limits the size of the corrections at each iteration to avoid convergence difficulties. Also, to prevent negative concentration and temperature, the code solves for the logarithm of the variables. A flowchart of the calculation procedure is given in figure 9.14, where v , T , and p are, respectively, the mixture mass-specific volume, temperature, and pressure.

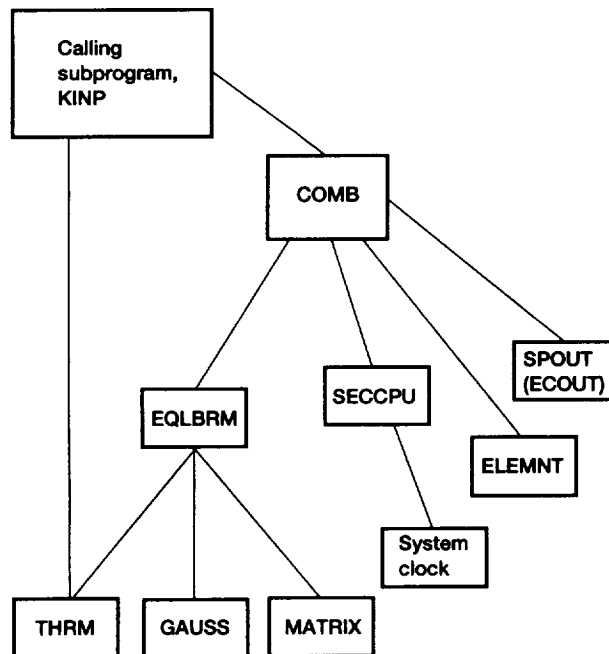


Figure 9.13.—Structure of chemical equilibrium section.

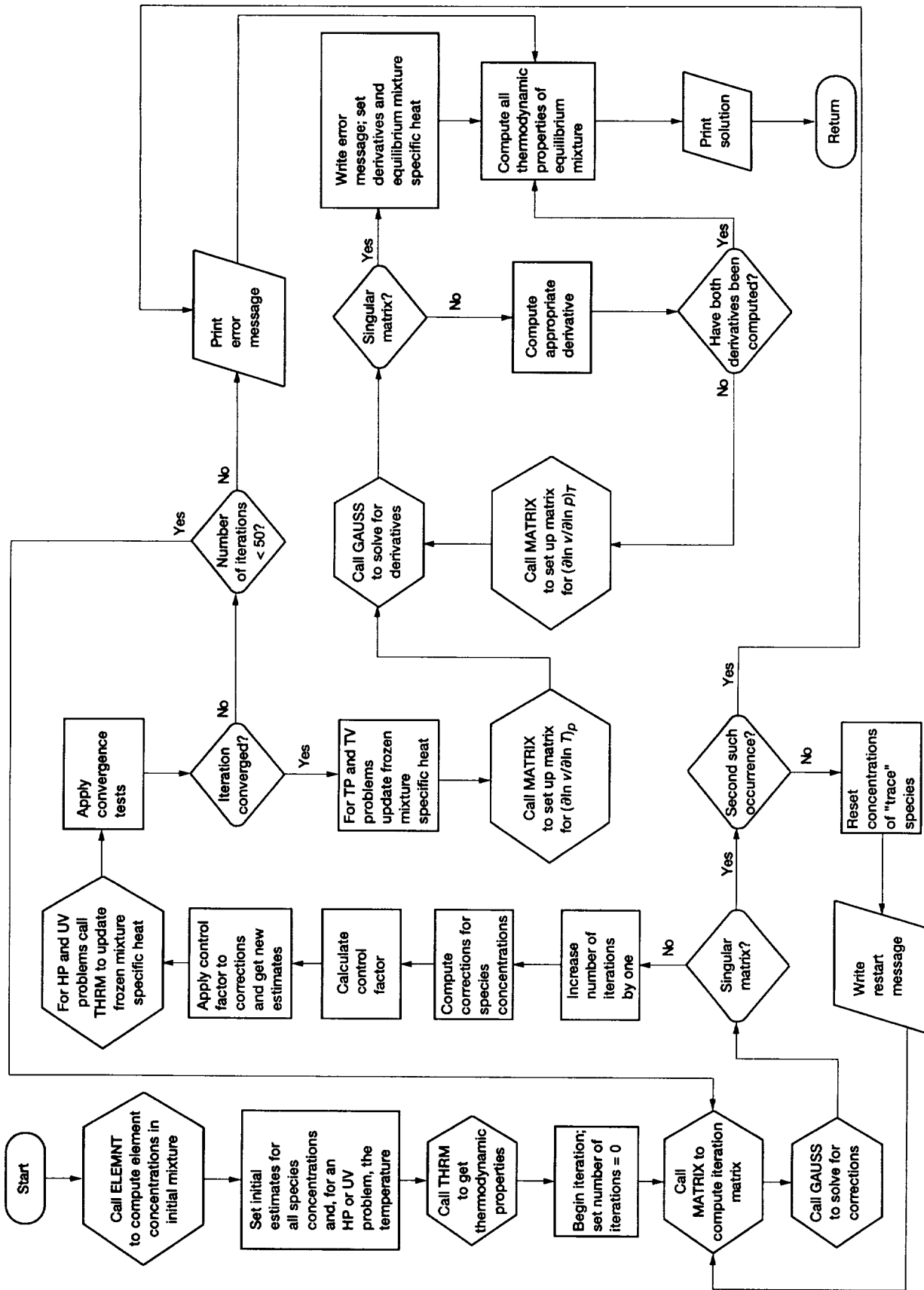


Figure 9.1.4.—Flowchart of equilibrium calculation procedure (subroutine EQLBRM).

9.7 Incident Shock Computations

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 when 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.

The main subprogram in this section of the code is SHOK, which manages the shock computation, calls for output of results, and prints the computational work required. It also sets the type of shock computation, equilibrium or frozen, to be performed. Figure 9.15 gives the structure of this section, which uses the routines ELEMNT, EQLBRM, SECCPU, SHOCKS, SPOUT, THRM, and those called by EQLBRM.

The routine SHOCKS, which was adapted from CET (ref. 11), sets initial estimates for both postshock conditions. It also computes both states by solving the mass, momentum, and energy conservation equations. A Newton-Raphson iteration procedure, which automatically limits the

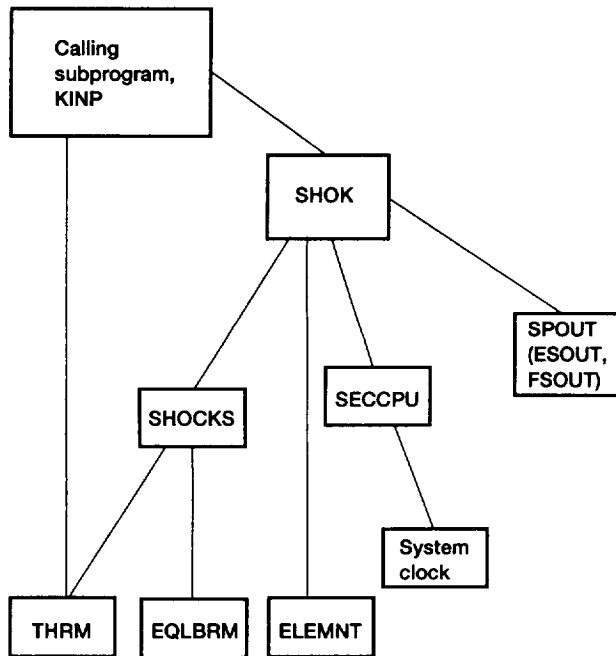


Figure 9.15.—Structure of incident shock section.

size of the corrections to minimize convergence difficulties, is used. In order to avoid negative variables during the solution procedure, the equations are cast in terms of the logarithm of the variables. Figure 9.16 presents the flowchart of SHOCKS. In this figure the internally defined logical variable EQL is used to denote the type of shock calculation, and L_m is a characteristic shock tube reaction length required for the post-shock kinetics problem. After the frozen shock conditions are successfully computed, L_m is calculated by SPOUT if the user has not provided a value for it.

Starting with the frozen shock state the code follows the progress of the chemical reaction in the shocked gas by integrating the ODE's 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 (refs. 35 to 37):

$$\frac{A(x)}{A_s} = \frac{1}{1 - (x/L_m)^\eta} \quad (9.4)$$

In this equation $A(x)$ is the area at distance x , A_s is the shock tube cross-sectional area, L_m is a characteristic length, and the exponent η is either 0.5 for a laminar boundary layer or 0.8 for a turbulent boundary layer.

The length L_m also depends on whether the boundary layer is laminar or turbulent. It may be specified by the user or is computed by the code as follows (refs. 36 and 37):

$$L_m = \left(\frac{d_H}{4\beta} \right)^{\frac{1}{\eta}} \left(\frac{p_1}{p_2} \right) \left(\frac{V_1}{V_1 - V_2} \right) \left(\mathcal{M}_1 \frac{p_1}{p_{st}} \frac{\rho_{st} c_{st}}{\mu_{st}} \right)^{\frac{1-\eta}{\eta}} \quad (9.5)$$

In this equation $d_H (= 4A_s/L_p)$, where L_p is the shock tube perimeter) is the shock tube hydraulic diameter; β is a boundary layer thickness parameter; $\mathcal{M} (= V/c)$, where c is the sonic velocity) is the Mach number; p_{st} is a standard pressure, for example, 1 atmosphere; and μ is the mixture dynamic viscosity. The subscripts 1 and 2 refer, respectively, to conditions upstream and downstream of the shock, using a coordinate system attached to the shock, which is therefore stationary. Thus the unshocked gas flows into the shock at velocity V_1 , and the shocked gas flows away from it at velocity V_2 . Finally the subscript “st” denotes standard conditions and the quantity $\rho_{st} c_{st} / \mu_{st}$ is assumed to be the same as $\rho_1 c_1 / \mu_1$.

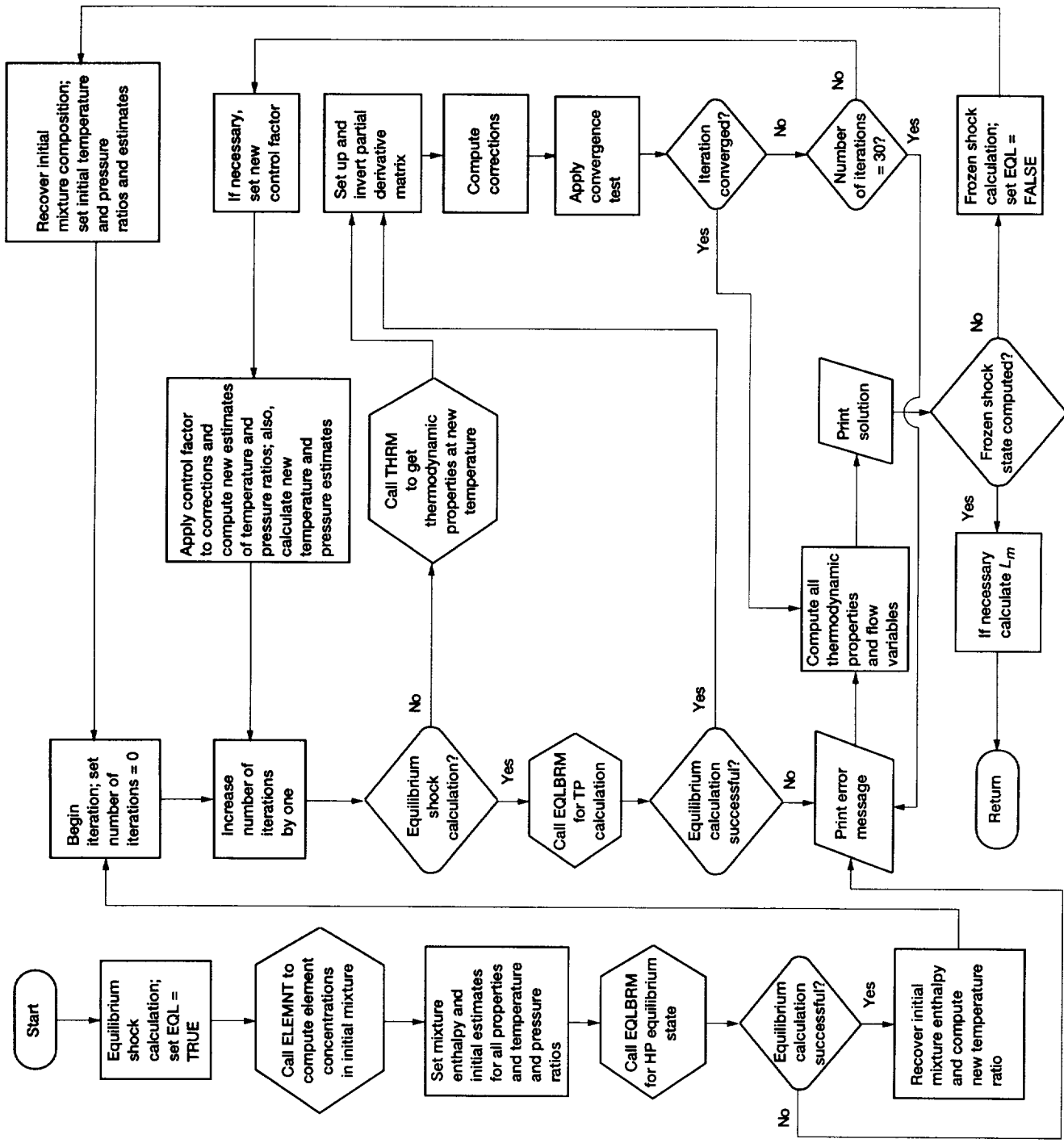


Figure 9.16.—Flowchart of incident shock calculation procedure (subroutine SHOCKS).

9.8 Perfectly Stirred Reactor Computations

Perfectly stirred reactor 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 the reactor exit. In the latter case the mass flow rate and the reactor exit mixture composition are computed. The problem type is identified by examining the user input parameters required for problem solution, and so a separate switch need not be set.

The main routine in this program section, whose structure is shown in figure 9.17, is WSR. The other routines required

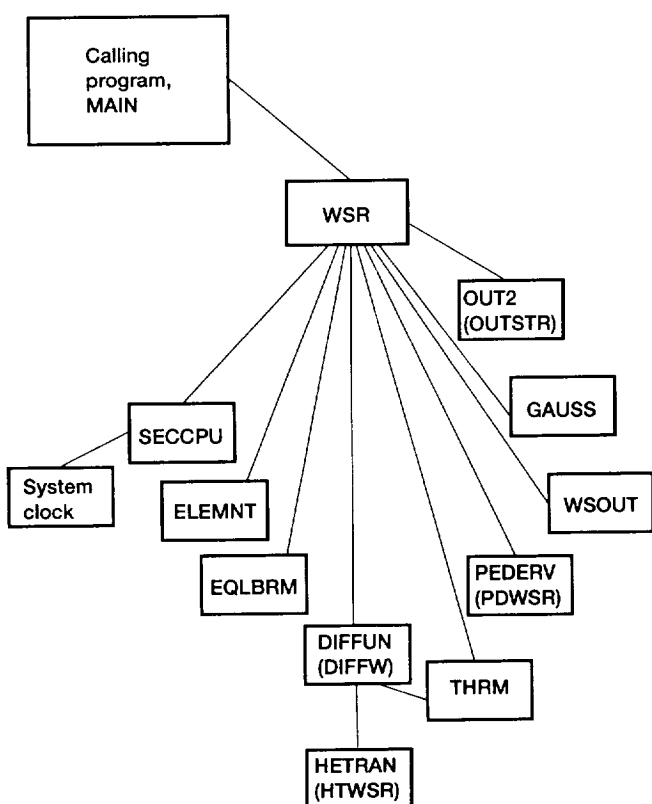


Figure 9.17.—Structure of perfectly stirred reactor section.

by this section are DIFFUN, ELEMNT, EQLBRM, GAUSS, HETRAN, OUT2, PEDERV, SECCPU, THRM, and WSOUT. The flowchart of WSR is given in figure 9.18, where \dot{m} , T , and τ_r are the mass flow rate, temperature, and residence time, respectively. Also \dot{m}_1 is the mass flow rate that is specified for the first solution of an assigned-mass-flow-rate problem, and T_1 and \dot{m}_0 are the prescribed temperature and mass flow rate to start iteration, respectively, for the first solution of an assigned-temperature problem. Finally T_{eq} is the equilibrium temperature for the reactor inlet state.

For both problem types WSR solves the governing nonlinear algebraic equations by using a Newton-Raphson iteration technique, which 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 equilibrium state, a series of perfectly stirred reactor computations 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 routine includes several tests and, when necessary, restarts of the calculation to ensure that the solution is physically meaningful. Finally it includes tests for possible blowout of the chemical reaction within the reactor.

9.9 Error Messages

The code contains many error messages—too numerous to list here. Every input parameter is tested for legality and consistency with the other input variables. If an illegal input parameter is discovered, a detailed message is printed. Each error message is self-explanatory and complete. It not only describes the mistake but tells the user how to fix the problem. During execution some tests are made to ensure that variables are within either given or reasonable 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.

Chapter 10

Code Implementation

The LSENS package may have to be modified to make it compatible with the user's computer system and computational requirements. The modifications are classified into two categories, (1) those that must be made before the present version of the code can be used and (2) those required to enhance the computational capabilities of the code. In this chapter we describe the first category of modifications. We also discuss certain optional changes that may be made. We then list parameters that place restrictions on the size of the problem that can be solved with the present version of the code. The code adjustments needed to alter the built-in values for these parameters (i.e., the second category of modifications) are, however, described in appendix C. We conclude this chapter with a brief note on restrictions that may have to be placed on the optimization level when compiling subroutine KINP (see table 9.1).

10.1 Computer Language

The code is compatible with most FORTRAN 77 compilers. However, it contains detailed instructions on how to convert it to FORTRAN 66, as discussed in chapter 9 (section 9.1).

10.2 Suppressing Underflow Error Messages

The MAIN program includes a system call to suppress underflow error messages. The form of this call used in the present version of LSENS is

```
CALL ERRSET (208, 256, -1, 1, 1)
```

where ERRSET is a VS FORTRAN-supported subroutine that enables the user to control execution when error conditions occur. The number 208 is an IBM-designated code for

underflow. On other systems the call may have to be changed or deleted. For example,

```
CALL ERRSET (74, .TRUE., .FALSE., .FALSE., .FALSE.)
```

is appropriate for VAX computers.

10.3 Function SECCPU

The function SECCPU, which computes the incremental central processing unit (CPU) time in seconds (see table 9.1), may have to be replaced. This function calls the system clock and is given for several common computing systems in appendix B.

10.4 Modifying Subroutine XERRWV

The routine XERRWV, which prints out error messages from the LSODE package and the sensitivity routines DMACHK, DMINTR, and DMSTOD (see table 9.1), is machine and language dependent. Therefore the data-type declaration for the argument MSG, which is a Hollerith literal or integer array containing the message to be printed, may have to be changed, as described in reference 22. However, the routine itself gives the necessary modifications for many machine environments.

10.5 BLOCK DATA Variables

The next modification concerns the values assigned to the variables SMEST, TESTNO, THCX(3), and THDCX(1) in the BLOCK DATA module. The double-precision variable SMEST (current value, 1.0D-35) is used in normalizing sensitivity coefficients. Any dependent variable Y_i that is smaller in magnitude than SMEST is set equal to SMEST to avoid

10. Code Implementation

computational difficulties caused by an excessively small (or zero) Y_i . The real variable TESTNO (current value, $-1.0E+35$), which is essentially $-\infty$, is the value used to initialize several variables, such as the print stations, that is, distinct points at which the solution is required. Both SMEST and TESTNO may be changed by the user. The double-precision variables THCX(3) (= 2/3) and THDCX(1) (= 4/3) are used to compute certain thermodynamic properties and may be changed to make them consistent with the precision of (i.e., number of decimal digits considered by) the machine.

All logical unit numbers for input and output are also set in the BLOCK DATA and may be changed by the user. The logical unit numbers used for input are as follows: First, a data file containing thermodynamic data for the species must be made available on unit LTHM, whose current value is 4. Second, data for the user's problem are read from unit LREAD, which is currently 5. This data type includes the chemical reaction mechanism, keywords identifying the problem to be solved, initial conditions, etc. Thermodynamic data may also be included in this file, in which case they are copied onto a separate scratch file associated with the logical unit LSCR, whose present setting is 9. Subsequently, these coefficients are read from unit LSCR. All other data are copied onto, and later read from, a separate scratch file, which must be made available on the logical unit LDAT, which is currently 7. Finally, if transport properties are required for the problem, the necessary data are read from the logical unit LTRAP (current value, 8).

Summary information regarding storage and work requirements for sensitivity analysis from the routine DMPRNT and all error messages from the routine XERRWV are sent to the logical unit LUNIT, which is currently 6. All other output information is directed to the logical unit LWRITE, whose current value is also 6.

10.6 Reducing Storage Requirement

The final code modification concerns computer memory requirement. If the sensitivity analysis option is not required and computer storage is of concern to the user, the lengths LRW and LIW of the real and integer work arrays RWORK and IWORK, respectively, may be decreased to the values indicated in the code. The dimensions of RWORK and IWORK must also be decreased to the values given in the code. The storage requirement may be further reduced by replacing all sensitivity subroutines except SENDDM (see

chapter 9) and the common blocks SENNAM, SENNOR, SENPAR, and SENSOL with dummy versions or, if allowed by the loader, even eliminating them. If SENPAR is either deleted or replaced with a dummy version, the EQUIVALENCE statement in subroutine RXNTAB must be deleted.

The present version of the code makes no use of the routines RSCOM, SVCOM, XSETF, and XSETUN (see table 9.1), which may all be deleted if the user intends to use LSENS solely as a black box that provides solutions to kinetics and sensitivity analysis problems. In addition, if the iteration techniques given by MITER = 4 and MITER = 5 (see table 9.5) will not be used, the routines DGBFA and DGBSL may be replaced with dummy routines or even eliminated, if allowed by the loader. Finally, if parameters related to problem size are smaller than the built-in values (see section 10.7), the modifications described in appendix C may be made to decrease the storage requirement.

10.7 Code Limitations

The present version of LSENS allows for a maximum number of 50 (reacting plus inert) species participating in a maximum number of 250 reactions, of which up to 35 may be of the third-body collisional type. The maximum number of elements (atoms, etc.) that can be used for any problem is 15. However, a species may contain no more than four different elements. For any general third-body reaction, third-body collisional efficiencies different from unity can be specified for a maximum number of 10 species. As described in chapter 11, one or more of the three quantities pressure, area, and temperature can be specified for the complete problem by means of a table of values. If this option is used, up to 100 values can be given. Finally, the maximum number of print stations is 100.

10.8 Compiler Optimization Level

On some computer systems (e.g., Sun SPARCstation 1) the size of subroutine KINP may prevent the use of high optimization levels. If any compilation difficulty is encountered, this subroutine should be compiled separately with a reduced optimization level that is acceptable to the system. All other subprograms can, however, be compiled with any optimization level.

Chapter 11

Description of Code Usage

Two input data files are normally required to execute the code. The first one, which we call the standard thermodynamic data file, must contain the chemical symbol and composition, molar mass, and thermodynamic data for each species. The second file, which we call the problem data file, must give information about, and data required by, the problem (or problems) to be solved. If desired, the user may include in this data file thermodynamic data for any species. A third input file, containing transport properties data, is required for certain computations. This file will be called the transport properties data file.

In this chapter we describe code usage and the input data files and provide a guide to their preparation. We first discuss the thermodynamic and transport properties data information that must be supplied. We then describe the preparation of the problem data file for a single case. Finally, we explain the construction of the data file required to solve multiple problems in a single run.

11.1 Thermodynamic Data Information

The thermodynamic data for the species may be supplied through the standard thermodynamic data file and/or the problem data file. The user informs the code where to get the thermodynamic data by means of the keyword TAPE or CARD in columns 1 to 4 of the first line of the problem data file. If data for all species are to be taken from the standard thermodynamic data file, the keyword TAPE must be used. A standard file, containing data from the most recent thermodynamic data base of the CET code (ref. 11) for many species in the C-H-N-O system, is supplied with LSENS. Figure 11.1 shows the beginning of this file. The first line contains the three temperatures TLOW, TMID, and THIGH, which together define two temperature ranges (300 to 1000 K and 1000 to 5000 K in fig. 11.1) for two sets of the seven coefficients defined in equations (8.32) to (8.34). The format used to read this line is 3F10.3. For each species the data are written in a group of four lines. For convenience each line con-

tains its number in column 80. The first line contains the following identification information: species name, thermodynamic data source and date when compiled, composition and phase of the species, temperature range over which the thermodynamic data are valid, and molar mass of the species. The format used to read this line is A8, 16X, 4(A2, F3.0), 21X, E13.6. On the next three lines the coefficients for the high-temperature range, followed by those for the low-temperature range, are listed. The format used to read these lines is 5E15.8, /, 5E15.8, /, 4E15.8. For some species the fourth line contains a fifth number, which is the heat of formation at 25 °C divided by the universal gas constant and has units of Kelvin. The end of the standard file is indicated by a line containing the word END in columns 1 to 3.

If thermodynamic data for some (or all) species are to be selected from the problem data file, the keyword CARD should be used. The thermodynamic data must then be placed immediately after the first line. The format for this information is identical to that of the standard thermodynamic data file. That is, the second line of the problem data file must contain the three temperatures TLOW, TMID, and THIGH. Next thermodynamic data must be given for all species for which the user wishes to provide this information in the problem data file. For each such species, irrespective of the case (or cases) in which it appears, the four lines of data described previously must be supplied. Finally the end of the thermodynamic data is indicated by a line containing the word END in columns 1 to 3.

If data are given for any species in both the standard thermodynamic data file and the problem data file, the latter information will be used and the former ignored. Thus the user has a convenient way of temporarily changing any thermodynamic data or adding new species without modifying the standard file.

The thermodynamic data information, that is, either the keyword TAPE or the keyword CARD followed by the required data, is given only once, at the beginning of the problem data file, irrespective of the number of problems contained therein. Therefore, if any thermodynamic data are to

```

300.000 1000.000 5000.000
AR L 5/66AR 1.00 0.00 0.00 0.G 300.000 5000.000 39.94800 1
0.25000000E 01 0.00000000 0.00000000 0.00000000 0.00000000 2
-0.74537502E 03 0.43660006E 01 0.25000000E 01 0.00000000 0.00000000 3
0.00000000 0.00000000-0.74537498E 03 0.43660006E 01 0.00000000 4
BR J 6/74BR 1. 0. 0. 0.G 300.000 5000.000 79.909 1
0.20843207E 01 0.71949483E-03-0.27419924E-06 0.42422650E-10-0.23791570E-14 2
0.12858837E 05 0.90838003E 01 0.24611551E 01 0.33319275E-03-0.10080655E-05 3
0.12262126E-08-0.44283510E-12 0.12711920E 05 0.69494733E 01 4
BR2 J12/61BR 2.0 0.0 0.0 0.G 300.000 5000.000 159.818 1
0.44479495E 01 0.10051208E-03-0.16393816E-07 0.22685621E-11-0.10236774E-15 2
0.23659941E 04 0.40888431E 01 0.38469580E 01 0.26111841E-02-0.40034147E-05 3
0.28120689E-08-0.73256202E-12 0.24846984E 04 0.69696985E 01 4
C J 3/78C 1. 0. 0. 0.G 300.000 5000.000 12.01100 1
0.25769424E 01-0.13903944E-03 0.69481807E-07-0.67414021E-11-0.43389004E-16 2
0.85425220E 05 0.43358122E 01 0.25279476E 01-0.12519400E-03 0.22544496E-06 3
-0.18489024E-09 0.57291741E-13 0.85448374E 05 0.46274790E 01 0.00000000 4
CH J12/67C 1.H 1.0 0.0 0.G 300.000 5000.000 13.01890 1
0.22673116E 01 0.22043000E-02-0.62250191E-06 0.69689940E-10-0.21274952E-14 2
0.70838037E 05 0.87889352E 01 0.35632752E 01-0.20031372E-03-0.40129814E-06 3
0.18226922E-08-0.86768311E-12 0.70405506E 05 0.17628023E 01 0.00000000 4
CH2 J12/72C 1.H 2. 0. 0.G 300.000 5000.000 14.02680 1
0.27525479E 01 0.39782047E-02-0.14921731E-05 0.25956899E-09-0.17110673E-13 2
0.45547759E 05 0.66534799E 01 0.35883347E 01 0.21724137E-02-0.13323408E-05 3
0.19469445E-08-0.89431394E-12 0.45315188E 05 0.22627869E 01 0.00000000 4
CH20 J 3/61C 1.H 2.0 1.0 0.G 300.000 5000.000 30.02620 1
0.28364249E 01 0.68605298E-02-0.26882647E-05 0.47971258E-09-0.32118406E-13 2
-0.15236031E 05 0.78531169E 01 0.37963783E 01-0.25701785E-02 0.18548815E-04 3
-0.17869177E-07 0.55504451E-11-0.15088947E 05 0.47548163E 01 0.00000000 4
CH202 L 5/80C 1.H 2.0 2. 0.G 300.000 5000.000 46.02559 1
0.57878771E 01 0.75539909E-02-0.30995161E-05 0.54494809E-09-0.34704210E-13 2
-0.48191230E 05-0.65299015E 01 0.21183796E 01 0.11175469E-01 0.26270773E-05 3
-0.81816403E-08 0.30133404E-11-0.46669293E 05 0.14480175E 02 0.00000000 4
CH3 J 6/69C 1.H 3.0 0.0 0.G 300.000 5000.000 15.03470 1
0.28400327E 01 0.60869086E-02-0.21740338E-05 0.36042576E-09-0.22725300E-13 2
0.16449813E 05 0.55056751E 01 0.34666350E 01 0.38301845E-02 0.10116802E-05 3
-0.18859236E-08 0.66803182E-12 0.16313104E 05 0.24172192E 01 0.00000000 4
CH20H BUR 84C 1.H 3.0 1. 0.G 300.000 5000.000 31.03410 1
0.46129885E 01 0.62971413E-02-0.20029483E-05 0.27889868E-09-0.13731448E-13 2
-0.38908596E 04 0.52852684E 00 0.33168221E 01 0.66920184E-02 0.28148397E-05 3
-0.57237202E-08 0.20724056E-11-0.33146321E 04 0.81576948E 01-0.20128906E 04 4
CH30 L 6/80C 1.H 3.0 1. 0.G 300.000 5000.000 31.03410 1
0.37590771E 01 0.78945048E-02-0.26710486E-05 0.39814507E-09-0.21447125E-13 2
0.13208081E 03 0.29936295E 01 0.21097078E 01 0.71927756E-02 0.53939157E-05 3
-0.74311082E-08 0.20939578E-11 0.97822803E 03 0.13137219E 02 0.00000000 4
CH4 L 5/84C 1.H 4. 0. 0.G 300.000 5000.000 16.04260 1
0.21737833E 01 0.89936592E-02-0.27855467E-05 0.39775117E-09-0.19976425E-13 2
-0.10216566E 05 0.70773563E 01 0.29428148E 01 0.25153728E-02 0.79085839E-05 3
-0.47495483E-08 0.14244910E-12-0.10056824E 05 0.45714579E 01-0.90051691E 04 4
CH30H L 4/80C 1.H 4.0 1. 0.G 300.000 5000.000 32.04200 1
0.40334730E 01 0.93680508E-02-0.30449373E-05 0.43456994E-09-0.22136329E-13 2
-0.26159859E 05 0.23539820E 01 0.26587849E 01 0.73515214E-02 0.71443337E-05 3
-0.87661114E-08 0.23805116E-11-0.25353684E 05 0.11238121E 02 0.00000000 4
CN J 6/69C 1.N 1.0 0.0 0.G 300.000 5000.000 26.01770 1
0.36036285E 01 0.33644390E-03 0.10028933E-06-0.16318166E-10-0.36286722E-15 2
0.51159833E 05 0.35454505E 01 0.37386307E 01-0.19239224E-02 0.47035189E-05 3
-0.31113000E-08 0.61675318E-12 0.51270927E 05 0.34490218E 01 0.00000000 4
CNN J 6/66C 1.N 2.00 0.00 0.G 300.000 5000.000 40.02440 1
0.48209077E 01 0.24790014E-02-0.94644109E-06 0.16548764E-09-0.10899129E-13 2
0.68685948E 05-0.48484039E 00 0.35077779E 01 0.72023958E-02-0.75574589E-05 3
0.42979217E-08-0.94257935E-12 0.68994281E 05 0.60234964E 01 0.00000000 4
END

```

Figure 11.1.—Illustration of structure of standard thermodynamic data file.

be included in this file, they should be placed here, regardless of the case (or cases) for which they are intended. Thus the same thermodynamic data (i.e., those given in the standard thermodynamic data file and/or the problem data file) will be used for all problems.

11.2 Transport Properties Data

Transport properties data are required only if heat transfer rates are to be computed by using the built-in heat transfer correlations (see eqs. (8.24) and (8.29)). This information must be supplied in a separate data file and cannot be included in the problem data file. A transport properties data file is provided with the code. This file contains coefficients for curve-fitted equations giving viscosity (in grams per centimeter per second $\times 10^6$) and thermal conductivity (in calories per centimeter per second per Kelvin $\times 10^6$) as a function of temperature for 19 species (see eqs. (8.37) and (8.38)) and is listed in figure 11.2. Each line contains data for one species and one transport property in the following order: species

name, the four coefficients, and the transport property. The format for reading each line is 1X, A8, 4X, 4E15.6, 3X, A4. The end of the file is indicated by a line containing the word LAST in columns 1 to 4.

11.3 Problem Data File for Single Case

Following either the keyword TAPE or the end of the thermodynamic data the problem data file must contain information required to execute the first (or only) case. The various types of input data needed to execute a single case, in the order that they must be placed after the thermodynamic data information, are as follows:

- (1) Title line
- (2) Chemical reactions and collisional efficiencies
- (3) Inert species list
- (4) Integration and assigned variables, units, and fuel name line

H2	0.687200E 00	-0.617320E 00	-0.111490E 03	0.577240E 00	VISC
H2	0.116129E 01	0.469043E 03	-0.551496E 05	-0.149041E 01	COND
CH3OH	0.641455E 00	-0.211775E 03	0.125265E 05	0.150983E 01	VISC
CH3OH	0.793792E 00	-0.487550E 03	0.322097E 05	0.646522E 00	COND
CH4	0.600440E 00	-0.817476E 02	0.165196E 04	0.154710E 01	VISC
CH4	0.853201E 00	-0.288931E 03	0.193692E 05	0.355349E 00	COND
C2H2	0.579032E 00	-0.152664E 03	0.628889E 04	0.177748E 01	VISC
C2H2	0.666734E 00	-0.330498E 03	0.165066E 05	0.118596E 01	COND
C2H4	0.578808E 00	-0.148526E 03	0.598107E 04	0.175623E 01	VISC
C2H4	0.674458E 00	-0.506382E 03	0.309615E 05	0.150224E 01	COND
C3H8	0.578645E 00	-0.156438E 03	0.658750E 04	0.156557E 01	VISC
C3H8	0.638223E 00	-0.615207E 03	0.349529E 05	0.183162E 01	COND
C6H6	0.611472E 00	-0.214866E 03	0.120814E 05	0.144145E 01	VISC
C6H6	0.549565E 00	-0.765765E 03	0.437054E 05	0.222155E 01	COND
NO	0.646504E 00	0.388567E 01	-0.858737E 04	0.165204E 01	VISC
NO	0.614175E 00	-0.260319E 03	0.360173E 05	0.110633E 01	COND
N2	0.633933E 00	-0.119153E 02	-0.471490E 04	0.166605E 01	VISC
N2	0.686393E 00	-0.161885E 03	0.236277E 05	0.482688E 00	COND
AR	0.590770E 00	-0.925770E 02	0.299040E 04	0.233077E 01	VISC
AR	0.641113E 00	-0.670283E 01	-0.730294E 04	0.207105E 00	COND
O	0.763927E 00	0.569723E 02	-0.345462E 04	0.822320E 00	VISC
O	0.776744E 00	0.803466E 02	-0.349619E 04	-0.481407E -01	COND
H	0.860209E 00	0.498177E 02	-0.542523E 04	-0.102874E 01	VISC
H	0.869708E 00	0.670316E 02	-0.828927E 04	0.891354E 00	COND
H2O	0.756380E 00	-0.301892E 03	0.187539E 05	0.987951E 00	VISC
H2O	0.123241E 01	-0.163667E 03	0.141541E 05	-0.291285E 01	COND
NH3	0.596434E 00	-0.314839E 03	0.271390E 05	0.197019E 01	VISC
NH3	0.774738E 00	-0.627003E 03	0.664059E 05	0.974813E 00	COND
NO2	0.603384E 00	-0.988315E 02	-0.969625E 03	0.200464E 01	VISC
NO2	0.504502E 00	-0.475725E 03	0.415525E 05	0.201241E 01	COND
OH	0.597497E 00	-0.365001E 03	0.351359E 05	0.217217E 01	VISC
OH	0.812287E 00	-0.264930E 03	0.343595E 05	0.105187E 00	COND
CO	0.779825E 00	0.193974E 03	-0.317845E 05	0.436375E 00	VISC
CO	0.800546E 00	-0.237400E 02	0.804541E 04	-0.468414E 00	COND
CO2	0.440370E 00	-0.288400E 03	0.193120E 05	0.324659E 01	VISC
CO2	0.603518E 00	-0.438483E 03	0.322949E 05	0.134023E 01	COND
O2	0.659260E 00	0.941422E 00	-0.711780E 04	0.164432E 01	VISC
O2	0.478050E 00	-0.452958E 03	0.579015E 05	0.228192E 01	COND
LAST					

Figure 11.2.—Transport properties data file supplied with LSENS.

11. Description of Code Usage

- (5) Problem data, related options, heat transfer data, and output controls (namelist PROB)
- (6) Assigned temperature profile (namelist TMPDAT) (optional)
- (7) Perfectly stirred reactor problem data (namelist WSPROB) (optional)
- (8) Initial conditions (and namelist START)
- (9) Local error tolerances and other integration controls (namelist SOLVER)
- (10) Sensitivity analysis data (and namelist SENRXN) (optional)
- (11) FINIS line

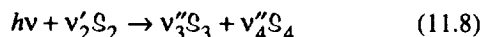
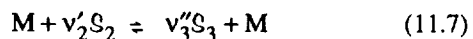
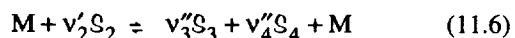
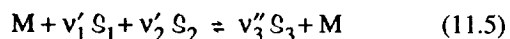
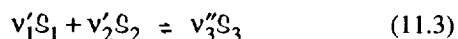
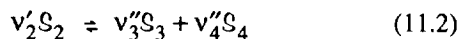
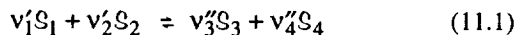
For each of these data types we describe the variables that have to be set and, where appropriate, give both the values to be assigned and the default values. Figure 11.3 gives the flow diagram of the necessary input, including available options.

11.3.1 Title Line

The first line after the thermodynamic data information is a title message, with a maximum length of 80 characters. This input line, which is read with the format 20A4, is provided as a convenient means of identifying the problem. If a title is not needed, a blank line must be inserted here.

11.3.2 Chemical Reactions and Collisional Efficiencies

For all problems the user must provide the chemical reaction mechanism and the rate coefficient data. The mechanism may contain a maximum of 250 reactions involving as many as 50 species. Any collisional reaction, including the spontaneous activation or deactivation of excited-state species, can be used. However, a maximum of only two different reactant and two different product species may be specified for a reaction. Therefore the following types of reactions are allowed:



In these equations 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 of species i , the species M represents a general third-body collision partner, and the quantity $h\nu$ represents a single quantum of radiation. The single arrow in equation (11.8) indicates that the photochemical reaction is irreversible. The double arrows in the other reactions, equations (11.1) to (11.7), mean that they can be either reversible or irreversible.

Following the title line the chemical reactions must be listed, one per line. The line has fields for the stoichiometric coefficients and names of two reactant and two product species. A general third-body species is denoted by the symbol M and a quantum of radiation by HNU . The symbol for the species electron is E in the standard thermodynamic data file supplied with LSENS. The reaction line also has fields for indicating a reversible or irreversible reaction and the special rate coefficient expression, equation (8.4). Values for the rate coefficient parameters A_j , n_j , and either E_j or c_j (see eqs. (8.3) and (8.4)) must be specified here. These parameters must be given in cgs units, that is, moles, centimeters, seconds, and calories. The reactions may be placed in any order, regardless of type. The format of each reaction line is 2(F3.1, A8, 1X), A1, 2(F3.1, A8, 1X), A1, E10.4, 2F10.4, as shown in table 11.1. If a stoichiometric coefficient is not listed, that is, the corresponding columns are left blank, a value of unity will be used.

If numerical instability or other difficulty occurs while processing the input data, the user should check the reaction mechanism, especially the three rate coefficient parameters, to ensure that every variable has been placed in its correct field. The image produced of the input data file (see chapter 12, section 12.1, for details) will assist the user in performing this task. These checks should also be made if any problems, such as excessive computational work or questionable results (e.g., exceedingly long or short induction delay or heat release time) occur.

The reaction mechanism may contain up to 35 general third-body collisional reactions, that is, one with M as a reactant and/or a product species. For any such reaction collisional efficiencies different from 1.0 may be specified for as many as 10 species as follows: A line with the word THIRDBODY in columns 4 to 12 is placed immediately after the collisional reaction for which third-body efficiencies are to be considered. This line is followed by one or more lines, each containing names and collisional efficiencies of up to four species (for which the efficiencies are different from unity). The A_j factor for any reaction of this type is for a standard collision partner, usually nitrogen or an inert gas such as argon. The format used to read each line is 4(A8, 2X, F10.3), corresponding to the variables in table 11.2.

The end of the third-body collisional efficiency list is indicated by either a blank line or the word END in the first three columns of a species field. If exactly four species are listed

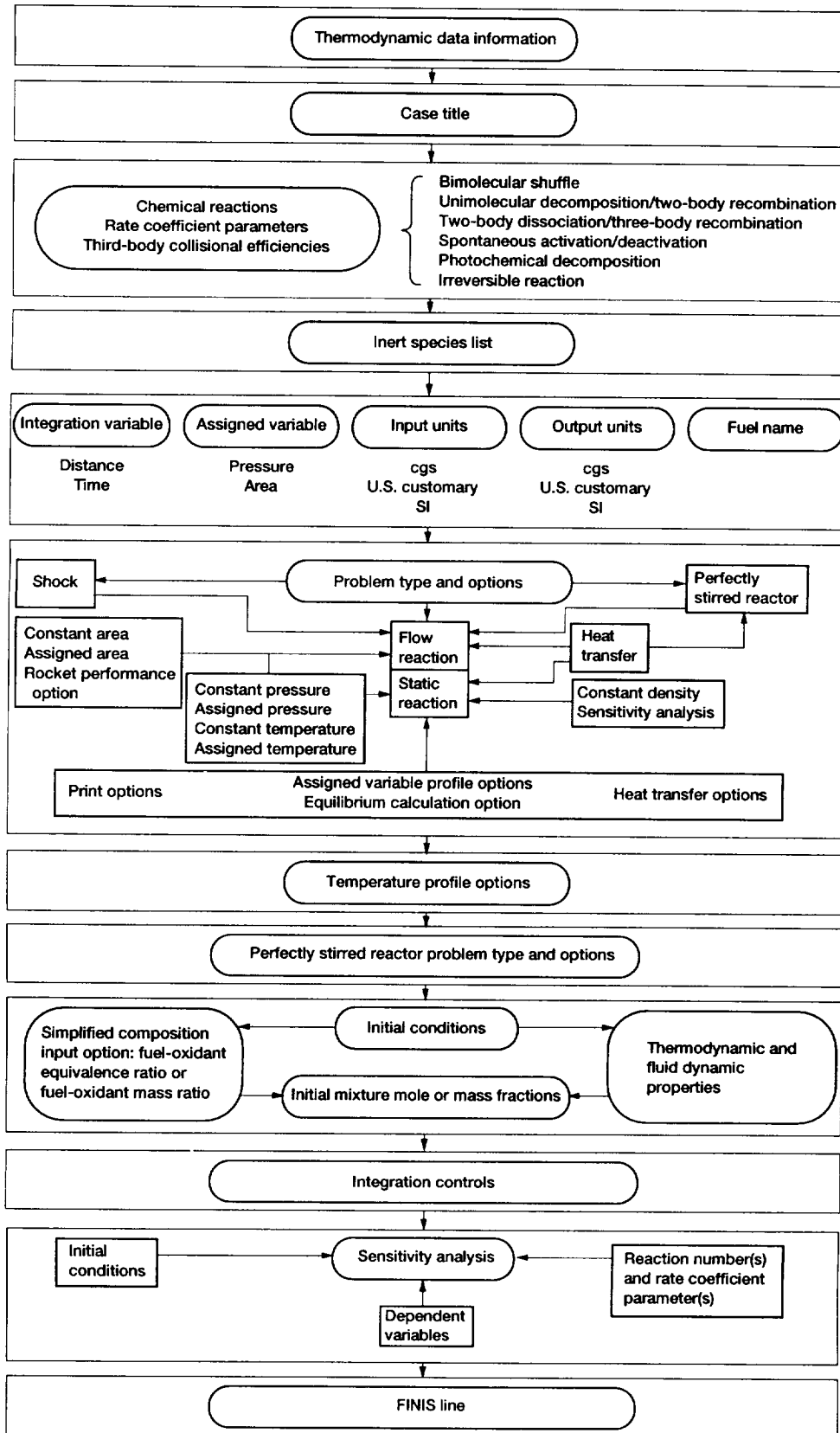


Figure 11.3.—Structure of problem data file for single (or first) case.

TABLE 11.1.—FORMAT OF REACTION LINE

Columns	Variable type	Format	Content and explanation
1-3	Real	F3.1	Stoichiometric coefficient of first reactant if two reactants are present; default value, 1.0
4-11	Character	A8	(a) Name of first reactant if two reactants are present (b) Letter M in column 4 if a collisional decomposition reaction (c) HNU in columns 4 to 6 if a photochemical decomposition reaction
13-15	Real	F3.1	Stoichiometric coefficient of second reactant (only reactant if a decomposition reaction); default value, 1.0
16-23	Character	A8	Name of second (or only) reactant
25	Character	A1	Equal to symbol (=) if reaction is reversible; any other symbol if it is irreversible
26-28	Real	F3.1	Stoichiometric coefficient of first product (only product if a recombination reaction); default value, 1.0
29-36	Character	A8	Name of first (or only) product
38-40	Real	F3.1	Stoichiometric coefficient of second product if two products are present; default value, 1.0
41-48	Character	A8	(a) Name of second product if two products are present (b) Letter M in column 41 if a collisional recombination reaction
50	Character	A1	Leave blank for equation (8.3); write "S" if equation (8.4) is used for k_j
51-60	Double precision	E10.4	A_j for equation (8.3) or (8.4)
61-70	Double precision	F10.4	n_j for equation (8.3) or (8.4)
71-80	Double precision	F10.4	E_j for equation (8.3) or c_j for equation (8.4)

TABLE 11.2.—FORMAT OF THIRD-BODY COLLISIONAL EFFICIENCY LINE

Columns	Variable type	Format	Content
1–8	Character	A8	Species name
11–20	Real	F10.3	Collisional efficiency
21–28	Character	A8	Species name
31–40	Real	F10.3	Collisional efficiency
41–48	Character	A8	Species name
51–60	Real	F10.3	Collisional efficiency
61–68	Character	A8	Species name
71–80	Real	F10.3	Collisional efficiency

on the last line, an additional line that either contains the word END in columns 1 to 3 or is blank must be added.

The end of the reaction list is denoted by either a blank line or the word END in columns 4 to 6.

11.3.3 Inert Species List

Following the reaction list, the names of any inert species, that is, species present but not reacting, must be given, eight per line, starting in columns 1, 11, 21, 31, 41, 51, 61, and 71. The format for each line is 8(A8, 2X). The end of the inert species list is indicated by a species field that either is blank or contains the word END in the first three columns. If the last line in this list contains exactly eight species, the next line should either be blank or have the word END in columns 1 to 3, as a more positive indicator of the end of the list. If no inert species is to be considered for the problem, the reaction list must be followed by a line that either contains the word END in columns 1 to 3 or is blank.

11.3.4 Integration and Assigned Variables, Units, and Fuel Name Line

The line following the inert species list specifies the independent variable for the problem and, as described here, the variable, if any, for which the user intends to assign a profile. This line also contains the input units, the output units, and possibly the fuel name.

The independent variable for the ordinary differential equations describing reaction chemistry depends on the problem. This variable, which we call the integration variable, can be either time or distance for any flow problem, including that following an incident shock or a perfectly stirred reactor computation. For a static problem, however, only time may be used.

To solve a static reaction problem, the user must provide either the pressure p or density ρ for the entire integration interval. Of the two, only pressure may be assigned as a func-

tion of time—density can be prescribed only as constant. For a flow problem the solution for either pressure p or area A must be supplied. In this case either quantity may be described by a function of either time or distance. The variable, pressure or area, for which a nonconstant profile can be given is called the assigned variable. Its name, PRESSURE or AREA, must be included on this line, even if it is specified as constant. If neither of the two variables is assigned (e.g., a constant-density problem), the corresponding field must be blank. The same is true when the assigned variable is described by a built-in function (e.g., a flow problem following an incident shock computation).

Either the same or different units may be selected for input and output. Table 11.3 lists the input units for each of the three systems, cgs, U.S. customary, and SI, built into the code.

The initial composition of the reacting mixture is usually specified by means of mole or mass fractions. However, for a general hydrocarbon fuel the user has the option of specifying this information by the simple means of either the fuel-oxidant equivalence ratio or the fuel-oxidant mass ratio. If the user intends to use the simplified-composition input option, the name of the fuel must be entered on this data line. The fuel name may be given even if this option is not used.

The integration and assigned variables, units, and fuel name line therefore contains only character variables, and its format is 4(A4, 6X), A8. The keywords and variable name that must be included here are given in table 11.4, with the default option, if any, underlined. If the fuel name is not required and default options are to be used for all other variables, this line may be blank.

11.3.5 Problem Data, Related Options, Heat Transfer Data, and Output Controls (Namelist PROB)

Next additional information regarding the problem to be solved is given in a namelist called PROB. It contains the logical variables that designate the problem type. The choices

TABLE 11.3.—UNITS FOR INPUT VARIABLES IN THE THREE SYSTEMS ALLOWED BY CODE

Variable	Units ^a		
	cgs (internal)	U.S customary	SI
Mass	g	lb	kg
Length	cm	ft	m
Time	s	s	s
Density	g/cm ³	lb/ft ³	kg/m ³
Pressure ^{b,c}	atm	lbf/ft ²	N/m ²
Area ^c	cm ²	ft ²	m ²
Volume	cm ³	ft ³	m ³
Temperature	K	°R	K
Velocity	cm/s	ft/s	m/s
Mass flow rate	g/s	lb/s	kg/s
Mixture composition	Mole or mass fraction	Mole or mass fraction	Mole or mass fraction

^aActivation energy must be in calories per mole, and the preexponential factor in cgs units.

^bPressure may also be specified in torr (millimeters of mercury) for all three systems.

^cFor rocket performance calculations the combustion chamber pressure and the nozzle throat area must be given in pounds per square inch absolute and square inches, respectively.

TABLE 11.4.—DESCRIPTION OF INTEGRATION AND ASSIGNED VARIABLES, UNITS, AND FUEL NAME LINE

Columns	Content	Explanation
1-4	<u>TIME</u>	Time is integration (i.e., independent) variable
1-8	<u>DISTANCE</u>	Distance is integration variable
11-18	<u>PRESSURE</u>	Pressure profile is assigned
11-14	<u>AREA</u>	Area profile is assigned (Leave blank for constant-density problem, shock kinetics problem, and perfectly stirred reactor problem with no flow problem following it)
21-23	<u>CGS</u>	Input is in cgs (internal) units
21-23	<u>FPS</u>	Input is in U.S. Customary units
21-22	<u>SI</u>	Input is in SI units
31-33	<u>CGS</u>	Output required in cgs (internal) units
31-33	<u>FPS</u>	Output required in U.S. Customary units
31-32	<u>SI</u>	Output required in SI units
41-48	Fuel name	Name of fuel if simplified-composition input option will be used to specify initial mixture composition

are static or flow kinetics, shock kinetics, perfectly stirred reactor (PSR) kinetics, and PSR-plus-plug-flow reaction kinetics. The namelist also contains logical variables that select options for static and flow problems. The options related to problem type include constant pressure, constant density, constant temperature, assigned temperature, constant area, heat transfer, heat transfer model, equilibrium reaction, and sensitivity analysis. Pressure and area profiles are also specified in PROB. Other options related to the testing of kinetic mechanism legality and the type of output generated are selected in this namelist. Finally, printout information is generally given here.

Because of the many options built into the code, PROB contains a large number of variables, which, together with their explanations, are listed in table 11.5 (default values are underlined). To assist the user in preparing this namelist (i.e., in deciding what variables have to be set), we consider each problem type and option in a separate subsection. The user has to read only the subsections relevant to the problem type and options selected.

11.3.5.1 Problem types and related options. — We now describe how to select each problem type and the related options, except heat transfer, for static and flow kinetics problems.

11.3.5.1.1 Static problem: For static problems the calculation may be performed either at constant density or with a user-supplied pressure-versus-time profile. A *constant-density static problem* is indicated by setting the logical variable RHOCON equal to TRUE.

An *assigned-pressure static problem* is selected by means of the keyword PRESSURE on the integration and assigned variables, units, and fuel name line, as described in section 11.3.4. The pressure-versus-time p - t profile can be specified by using one of the following two methods:

- (1) In tabular form as a function of time
- (2) As a polynomial function of up to third degree

$$p = CT0 + CT1 * t + CT2 * t^2 + CT3 * t^3 \quad (11.9)$$

To use method (1), the pressure values must be set in the array ATB, and corresponding time values in the array TTB. Method (2) is selected by giving nonzero values for one or more of the four constants CT0, CT1, CT2, and CT3. A *constant-pressure calculation* can be performed by setting a nonzero value for only CT0 (= the desired pressure). An alternative method of specifying a constant-pressure problem is to set the logical variable PCON equal to TRUE. The pressure will then be held constant at its user-supplied initial value (see section 11.3.8.1). The combination of the switch PCON equal to TRUE and a nonzero value for (only) CT0 can also be used.

11.3.5.1.2 One-dimensional flow problem: Either pressure or area may be assigned for a plug-flow (i.e., no back mixing) reaction problem. An *assigned-pressure flow problem* is

selected by means of the keyword PRESSURE on the integration and assigned variables, units, and fuel name line, as described in section 11.3.4. The pressure can be given as a function of either time t or distance x , independently of the choice of integration variable. One of the following two methods can be used to specify p :

- (1) In tabular form as a function of either time or distance
- (2) As a polynomial function of either time

$$p = CT0 + CT1 * t + CT2 * t^2 + CT3 * t^3 \quad (11.10)$$

or distance

$$p = CX0 + CX1 * x + CX2 * x^2 + CX3 * x^3 \quad (11.11)$$

To use method (1), the pressure values must be set in the array ATB, and either the corresponding time values must be set in the array TTB or the corresponding distance values in the array XTB. In order to specify a p - t profile by method (2), nonzero values must be given for one or more of the four coefficients CT0, CT1, CT2, and CT3. A p - x profile is given by assigning nonzero values for one or more of the four coefficients CX0, CX1, CX2, and CX3.

A *constant-pressure calculation* can be performed by setting a nonzero value for only CT0 (or CX0), which must be set equal to the desired pressure. An alternative method of specifying a constant-pressure problem is to set the logical variable PCON equal to TRUE. The pressure will then be held constant at its user-supplied initial value (see section 11.3.8.1). The combination of the switch PCON equal to TRUE and a nonzero value for (only) CT0 or CX0 will also work.

An *assigned-area flow problem* is selected by means of the keyword AREA on the integration and assigned variables, units, and fuel name line, as described in section 11.3.4. The area can be given as a function of either time t or distance x independently of the choice of integration variable. One of the following two methods can be used to specify A :

- (1) In tabular form as a function of either time or distance
- (2) As a polynomial function of either time

$$A = CT0 + CT1 * t + CT2 * t^2 + CT3 * t^3 \quad (11.12)$$

or distance

$$A = CX0 + CX1 * x + CX2 * x^2 + CX3 * x^3 \quad (11.13)$$

To select method (1), the area values must be set in the array ATB, and either the corresponding time values must be set

TABLE 11.5.—DESCRIPTION OF NAMELIST PROB

Variable name	Variable type	Value	Explanation
ACON	Logical	TRUE <u>FALSE</u>	Area is assigned as constant Area may vary for problem
APRINT	Real	----	Array of areas or pressures at which output is desired; maximum number of print stations, 100
ATB	Real	----	Array of areas or pressures when specifying assigned variable in tabular form; maximum number of values, 100
ATHROT	Real	----	Rocket nozzle throat area in square inches
BETA	Real	----	Dimensionless boundary layer parameter; used to compute LSUBM for shock kinetics problem
BORE	Real	----	Diameter of piston used for computing heat transfer coefficient when OTTO = TRUE; default value, 9 cm
COMBUS	Logical	TRUE <u>FALSE</u>	Equilibrium computation is to be performed Equilibrium computation is not required
CONC	Logical	<u>TRUE</u> FALSE	Mixture composition is to be printed out as moles per unit volume Mixture composition is to be printed out as mass fractions
CT0, CT1, CT2, CT3	Real	----	Coefficients in assigned variable AVAR versus time t equation: $AVAR = CT0 + CT1*t + CT2*t^2 + CT3*t^3$
CX0, CX1, CX2, CX3	Real	----	Coefficients in assigned variable AVAR versus distance x equation: $AVAR = CX0 + CX1*x + CX2*x^2 + CX3*x^3$
D	Real	----	Hydraulic diameter of shock tube in centimeters; used to compute LSUBM for shock kinetics problem
DBUGO	Logical	TRUE <u>FALSE</u>	Additional intermediate output is to be printed Only standard output is to be printed
END	Real	----	Value of independent variable at which integration is to be terminated; integration is stopped when solution has been advanced to independent variable value \geq END
ETA	Real	----	Dimensionless exponent in special area equation for shock kinetics problem
EXCHR	Logical	TRUE <u>FALSE</u>	Print energy exchange rate for each reaction Print net reaction conversion rate for each reaction
HTRAN	Logical	TRUE <u>FALSE</u>	Heat transfer between reacting system and environment is to be considered Heat transfer rate not required: either temperature is assigned, or reacting system is adiabatic
HT0, HT1, HT2, HT3, HT4	Real	----	Coefficients in polynomial heat transfer rate \dot{Q} (or \dot{Q}') equation as function of temperature T : \dot{Q} (or \dot{Q}') = $HT0 + HT1*T + HT2*T^2 + HT3*T^3 + HT4*T^4$
IPRINT	Integer	----	Number of integration steps between output of results; default value, 5
LSUBM	Real	----	Characteristic length in centimeters for shock tube reaction

TABLE 11.5.—Continued.

Variable name	Variable type	Value	Explanation
ORDER	Logical	TRUE	Ordered list of nonzero formation rates of all species due to each reaction is to be generated
		<u>FALSE</u>	List of species formation rates by individual reactions is not required
OTTO	Logical	TRUE	Built-in correlations are to be used to compute internal combustion engine heat transfer rate for assigned-pressure static problem
		<u>FALSE</u>	Built-in correlations are not needed for heat transfer computation
PC	Real	----	Rocket chamber pressure in pounds per square inch absolute
PCON	Logical	TRUE	Pressure is assigned as constant for static or flow problem
		<u>FALSE</u>	Pressure may change during the reaction
PRINT	Real	----	Array of integration variable values at which output is desired; maximum number allowed, 100
QMREAD	Logical	<u>TRUE</u>	Heat transfer rate is given by polynomial function of temperature
		<u>FALSE</u>	Heat transfer correlations built into code are to be used for heat transfer rate computations
RHOCON	Logical	TRUE	Density (volume) is to be held constant for static problem
		<u>FALSE</u>	Density is allowed to vary
ROCKET	Logical	TRUE	Rocket performance parameters are to be computed for flow problem
		<u>FALSE</u>	Rocket performance need not be computed
RPM	Real	----	Engine speed in revolutions per minute; used for heat transfer computation when OTTO = TRUE; default value, 1000
RXNTST	Logical	TRUE	Reaction mechanism must be checked for duplicate reactions and other inconsistencies
		<u>FALSE</u>	Reaction mechanism legality need not be checked
SENCAL	Logical	TRUE	Sensitivity coefficient computations are required
		<u>FALSE</u>	Sensitivity analysis is not needed
SENSTD	Logical	TRUE	Sensitivity coefficients of temporal derivatives of dependent variables are required
		<u>FALSE</u>	Sensitivity coefficients of temporal derivatives of dependent variables are not required
SHOCK	Logical	TRUE	Incident shock kinetics computation is to be performed
		<u>FALSE</u>	Incident shock computation is not needed
STROKE	Real	----	Length of piston used for computing heat transfer coefficient when OTTO = TRUE; default value, 8 cm
TASS	Logical	TRUE	Assigned-temperature calculation is to be performed; namelist TMPDAT must follow PROB (see section 11.3.6)
		<u>FALSE</u>	Temperature profile will not be assigned

TABLE 11.5.—Concluded.

Variable name	Variable type	Value	Explanation
TCON	Logical	TRUE <u>FALSE</u>	Temperature is to be held constant for static or flow problem Temperature may change during the reaction
TINY	Double precision	----	Smallest nonzero value (in magnitude) allowed for normalized sensitivity coefficients; default value, 0.0
TOTMAS	Real	----	Total reacting mass for static problem with heat transfer rate given by polynomial equation
TTB	Real	----	Array of times when specifying area or pressure in tabular form as function of time; values correspond to those in the ATB array; maximum number of points, 100
TWALL	Real	----	Wall temperature used with heat transfer correlation equations
VISC	Real	----	Standard state dynamic viscosity in grams per centimeter per second; used to compute LSUBM for shock kinetics problem
WELSTR	Logical	TRUE <u>FALSE</u>	Perfectly stirred reactor (PSR) computation is to be performed; namelist WSPROB must follow PROB (see section 11.3.7) PSR computation is not required
WSFLOW	Logical	TRUE <u>FALSE</u>	Flow problem computation is to be performed after PSR problem, using PSR results for composition, temperature, density, and mass flow rate; namelist WSPROB must follow PROB (see section 11.3.7) Flow problem does not follow PSR problem
XTB	Real	----	Array of distances if pressure or area is specified in tabular form as function of distance; values correspond to those in the ATB array; maximum number of points, 100

in the array TTB or the corresponding distance values in the array XTB. In order to specify an $A-t$ profile by method (2), nonzero values must be given for one or more of the four coefficients CT0, CT1, CT2, and CT3. An $A-x$ profile is given by assigning nonzero values for one or more of the four coefficients CX0, CX1, CX2, and CX3.

A *constant-area calculation* can be performed by setting a nonzero value for only CT0 (or CX0), which must be set equal to the desired area. An alternative method of specifying a constant-area problem is to set the logical variable ACON equal to TRUE. The area will then be held constant at its user-supplied initial value (see section 11.3.8.1). The combination of the switch ACON equal to TRUE and a nonzero value for (only) CT0 or CX0 is also valid.

11.3.5.1.3 Constant-temperature problem: For either a static or flow problem, including that following a PSR calculation, a constant-temperature calculation can be performed by setting the logical variable TCON equal to TRUE. The temperature will then be held constant at its user-supplied initial value (see section 11.3.8.1). This option cannot be used for the flow problem following an incident shock calculation.

Another method of specifying a constant-temperature calculation is described next.

11.3.5.1.4 Assigned-temperature problem: For a static or flow problem, except that following either an incident shock or PSR calculation, the temperature profile can be given as described in section 11.3.6. To select this option, the logical variable TASS must be set equal to TRUE. The option can also be used to perform a constant-temperature calculation (see section 11.3.6 for details).

11.3.5.1.5 Sensitivity analysis: This option can be selected for a static problem only. In order to indicate that sensitivity coefficients are to be computed, the logical variable SENCAL must be set equal to TRUE. If sensitivity coefficients of the temporal derivatives of the dependent variables are also required, the logical variable SENSTD should be set equal to TRUE. Another variable that both controls the sensitivity computations and is set in PROB is TINY (default value, 0.0), which is the magnitude of the smallest normalized sensitivity coefficient that can be considered significant. Any normalized coefficient that is smaller in magnitude than TINY will be set equal to zero.

11.3.5.1.6 Chemical equilibrium computations: To select this option, the logical variable COMBUS must be set equal to TRUE. The code assumes that a chemical equilibrium computation is being performed only to obtain additional information concerning a kinetics problem. Therefore a legal kinetics problem must be specified. Also, because the species list is constructed from the reaction mechanism, the latter must contain all species that are important to the equilibrium problem.

Depending on the kinetics problem specified, the code will automatically compute the equilibrium results for one of the following four assigned states: temperature and pressure (TP), specific enthalpy and pressure (HP), temperature and specific volume or density (TV), and specific internal energy and specific volume or density (UV). An assigned volume calculation (TV or UV) is executed for a constant-density kinetics problem (i.e., when the logical variable RHOCON is set equal to TRUE). For all other kinetics problem types an assigned-pressure calculation (TP or HP) is done. In either case an assigned-temperature equilibrium solution (TP or TV) is produced for a constant-temperature kinetics problem. For a nonconstant-temperature kinetics problem either an HP or a UV computation is executed, depending on the value of RHOCON.

If the user is solely interested in the results of the chemical equilibrium computation, the reaction mechanism and rate coefficient parameters are unimportant. Hence any mechanism that contains all required species may be used, nonzero values specified for only the preexponential factors, and all reactions made irreversible to increase the efficiency of the kinetics computation. Indeed the reactions do not even have to satisfy atom and charge balance requirements. However, in this case, an error exit may occur during the kinetics computation. Also reaction mechanism legality testing (see section 11.3.5.3) must not be required. Finally, in the interest of efficiency, only one print station (see section 11.3.5.4), very close to the initial value of the integration variable, should be specified.

11.3.5.1.7 Shock tube reaction: In order to compute the equilibrium and frozen conditions behind an incident shock wave, the logical variable SHOCK must be set equal to TRUE. The incident shock computations are followed by a one-dimensional flow problem with area assigned as a function of distance. In particular, the area profile is given by the special function, equation (9.4). The exponent ETA ($= \eta$) in this equation must be specified. The user must also give either the characteristic length LSUBM ($= L_m$) or the three quantities shock tube hydraulic diameter D ($= d_H$), unshocked gas dynamic viscosity VISC ($= \mu_1$), and the boundary layer thickness parameter BETA ($= \beta$) so that the code can compute L_m (see eq. (9.5)). Finally, because the species list is constructed from the reaction mechanism, the latter must contain all species important to the shock computations.

If the user is solely interested in studying postshock conditions, the reaction mechanism and rate coefficient parameters are unimportant. Hence any mechanism that contains all required species may be used, nonzero values specified for only the preexponential factors, and all reactions made irreversible to increase the efficiency of the kinetics computation. Indeed, the reactions do not even have to satisfy atom and charge balance requirements. However, in this case, an error exit may occur during the kinetics computation. Also reaction mechanism legality testing (see section 11.3.5.3) must not be required. Finally, in the interest of efficiency, only one print station (see section 11.3.5.4), very close to the initial value of the integration variable, should be specified.

11.3.5.1.8 Perfectly stirred reactor (PSR) problem: For solving a PSR problem the logical variable WELSTR should be set equal to TRUE. The type of problem to be solved and the required variables are specified elsewhere, as described in section 11.3.7.

11.3.5.1.9 Combined PSR-and-plug-flow problem: A plug-flow calculation may be performed after a PSR problem in a single computer 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 logical variable WSFLOW must be set equal to TRUE. The variables required for the PSR computation must be given elsewhere, as described in section 11.3.7. For the flow problem the assigned variable (pressure or area) and its profile must be specified, as described in section 11.3.5.1.2.

11.3.5.1.10 Summary: The variables that must be set in PROB for the different problem types and options considered here are summarized in table 11.6. For a flow problem following a PSR problem either pressure or area may be assigned by means of the profiles (and variables) given in this table. However, temperature may be assigned only as constant and that too only by means of the flag TCON: the option TASS = TRUE cannot be used. But pressure or area may be assigned as constant in any of the ways described in section 11.3.5.1.2. Finally temperature cannot be assigned for the flow problem following a shock computation.

11.3.5.2 Heat transfer models.—Any static or flow problem may be nonadiabatic, with an assigned heat transfer rate. The logical variable HTRAN must be set equal to TRUE to indicate that the heat transfer rate is required. The heat transfer rate \dot{Q} for a static problem or the heat transfer rate per unit length in the flow direction \dot{Q}' for a flow problem can be specified by one of the following two models:

- (1) As a polynomial function of up to fourth degree in temperature T :

$$\dot{Q} \text{ (or } \dot{Q}') = HT0 + HT1 * T + HT2 * T^2 + HT3 * T^3 + HT4 * T^4 \quad (11.14)$$

TABLE 11.6.—VARIABLES THAT MUST BE SET IN NAMELIST PROB FOR DIFFERENT PROBLEM TYPES AND RELATED OPTIONS^a BUILT INTO CODE

Problem type or option	Variables that must be set in namelist PROB
Constant-density static	RHOCON (= TRUE)
Constant-pressure static	PCON (= TRUE) and/or CT0
Assigned-pressure static, with pressure specified: As polynomial function of time In tabular form as function of time	One or more of CT0, CT1, CT2, and CT3 ATB and TTB
Constant-pressure flow	PCON (= TRUE) and/or one of CT0 and CX0
Constant-area flow	ACON (= TRUE) and/or one of CT0 and CX0
Assigned-pressure or assigned-area flow, with assigned variable specified: As polynomial function of time In tabular form as function of time As polynomial function of distance In tabular form as function of distance	One or more of CT0, CT1, CT2, and CT3 ATB and TTB One or more of CX0, CX1, CX2, and CX3 ATB and XTB
Constant-temperature static or flow ^{b,c}	TCON (= TRUE)
Assigned-temperature static or flow ^d	TASS (= TRUE)
Sensitivity analysis ^e	SENCAL (= TRUE); and possibly SENSTD (= TRUE) and TINY
Chemical equilibrium	COMBUS (= TRUE)
Incident shock	ETA; either LSUBM or all three of BETA, D, and VISC; and SHOCK (= TRUE)
Perfectly stirred reactor	WELSTR (= TRUE)
Perfectly stirred reactor plus plug flow	WSFLOW (= TRUE); variables that depend on the assigned variable and its profile, as described above; and for constant temperature flow TCON (= TRUE)

^aExcept those related to heat transfer, which are given in table 11.7.

^bAn alternative method of specifying a constant-temperature problem is to set TASS = TRUE in PROB and the appropriate variable in namelist TMPDAT (see table 11.10).

^cThis option cannot be used for a flow problem following an incident shock computation.

^dThis option cannot be used for a flow problem following either an incident shock or a PSR computation.

^eThis option can be selected for static problems only.

- (2) As built-in standard heat transfer correlations (see eqs. (8.22) to (8.31))

Only the second heat transfer model may be specified for the assigned-pressure, variable-volume, Otto-cycle engine static problem. All other static problems may use only the first model. For flow problems, however, either model may be selected.

To select the first heat transfer model, which is the default option corresponding to QMREAD = TRUE (see table 11.5), nonzero values must be given for one or more of the five coefficients HT0, HT1, HT2, HT3, and HT4 in equation (11.14). In addition, for a static problem the total mixture mass TOTMAS must be supplied.

To use the second heat transfer model, QMREAD must be set equal to FALSE and the transport properties data file made available. The wall temperature TWALL (= T_w in eq. (8.22)) must also be specified. Finally for the Otto-cycle model the variable OTTO must be set equal to TRUE and values given for engine speed RPM (in revolutions per minute) and cylinder bore BORE and stroke STROKE.

In summary, if heat transfer is to be considered, the required information depends on both the problem type and the heat transfer model. Table 11.7 lists the variables that must be specified for heat transfer computations. The default value for QMREAD is TRUE, so that this quantity need not be set if the heat transfer rate is given as a polynomial func-

tion of temperature. However, we recommend supplying QMREAD if the problem data file contains several problems, and we have therefore included it in table 11.7.

11.3.5.3 Reaction mechanism tests.—The logical variable RXNTST controls whether the legality of the reaction mechanism is tested. Setting RXNTST equal to TRUE tells the code to check that no reaction is duplicated and that each reaction satisfies charge and atom balance requirements. We strongly recommend that this option be used when either developing or preparing a new mechanism.

11.3.5.4 Print stations.—The user has a choice of three options (in namelist PROB) for specifying when the solution is to be generated. The results may be printed at designated values of either the integration variable ξ or the assigned variable. The third option is to print the solution at intervals of a prescribed number of integration steps. For the flow problem behind an incident shock wave, only the first or third option may be used. For all other flow problems, including that following a PSR problem, any option may be selected. The same is true for any static kinetics-only or kinetics-plus-sensitivity problem.

To use the first option, the output stations—the ξ values at which the solution is required—must be entered into the array PRINT. The output stations must be listed in order of increasing ξ . Any output station that is either less than or equal to the previous one will be ignored. The same is true of any output station that is not greater than the user-supplied initial value of ξ .

TABLE 11.7.—VARIABLES THAT MUST BE SET IN NAMELIST PROB FOR HEAT TRANSFER COMPUTATIONS

Problem type	Heat transfer model	Variables that must be set in namelist PROB
Constant-density static	Polynomial function of temperature ^a	HTRAN (= TRUE); one or more of HT0, HT1, HT2, HT3, and HT4; QMREAD (= TRUE) ^b ; and TOTMAS
Assigned-pressure static	Polynomial function of temperature	HTRAN (= TRUE); one or more of HT0, HT1, HT2, HT3, and HT4; QMREAD (= TRUE) ^b ; and TOTMAS
	Built-in correlations	BORE; HTRAN (= TRUE); OTTO (= TRUE); QMREAD (= FALSE); RPM; STROKE; and TWALL
Assigned-pressure or assigned-area flow	Polynomial function of temperature	HTRAN (= TRUE); one or more of HT0, HT1, HT2, HT3, and HT4; and QMREAD (= TRUE) ^b
	Built-in correlations	HTRAN (= TRUE); QMREAD (= FALSE); and TWALL

^aThis is the only heat transfer model allowed for constant-density static problems.

^bThe default value for QMREAD is TRUE.

11. Description of Code Usage

The second option may be selected only if the assigned variable is specified in tabular form and that too only as a monotonic function of ξ . To use this option, the assigned-variable values at which output is required must be entered into the array APRINT. Also they must be ordered such that the resultant ξ values at which output is produced increase. That is, if the values in ATB increase monotonically, so should the APRINT values, and vice versa.

To use the third option, values must be given for the integer variable IPRINT (= the number of integration steps between output of results) and the real variable END (= the ξ value at which the integration is to be terminated). The integration is stopped on the first step for which $\xi \geq \text{END}$, and the results computed on this step are also printed.

The above three methods are summarized in table 11.8, which also lists the variables that must be set in PROB. A fourth method, described in section 11.3.6 may be used for a certain type of assigned-temperature problem. If this option will be used, data concerning output frequency must not be given in PROB.

11.3.5.5 Output information specification.—As described in chapter 12, for each problem type certain standard information is generated automatically at each print station. Also some optional quantities can be obtained either in place of, or in addition to, standard output. The optional output information that is produced depends on the values assigned for the logical variables CONC, DBUGO, EXCHR, ORDER, and ROCKET (see table 11.5). All variables, except ROCKET, may be specified for any static or flow problem, including that following a shock computation or a PSR problem. For static and postshock flow problems, however, ROCKET is

not relevant and must not be used. None of the variables, except DBUGO will have any impact on the information produced for sensitivity analysis, equilibrium, shock, and PSR computations.

The mixture composition is printed out as mole fractions and moles per unit volume. If the user prefers mass fractions to molar concentrations, CONC should be set equal to FALSE. (Mole fractions will, however, always be given.)

Setting DBUGO equal to TRUE will result in the output of several intermediate quantities that are useful for debugging. This option may also be used with a PSR calculation for the same purpose. The variables that are actually printed are listed in chapter 12 (see table 12.2), which may be consulted for details.

Another standard output quantity that can be replaced by an optional one is the net reaction conversion rate, equation (8.56), which is the net reaction rate divided by the square of the density and is printed for each reaction. It is provided as a convenient means of comparing net reaction rates. In its place the net energy exchange rate, equation (12.13), which is proportional to the heat release rate from a reaction, can be obtained by setting EXCHR equal to TRUE. This quantity is useful in certain situations, such as identifying reactions that are important to an ignition or heat release process.

A more definitive idea of the relative importance of reactions to species formation rates can be obtained by means of the option ORDER. If this logical variable is set equal to TRUE, an ordered list of nonzero net formation rates due to each reaction, equation (8.58), will be given for every species.

During a nozzle expansion process the rocket performance parameters specific impulse, vacuum specific impulse, area ratio, characteristic velocity, and thrust coefficient, equations

TABLE 11.8.—OPTIONS THAT SPECIFY WHEN OUTPUT IS TO BE GENERATED AND VARIABLES THAT MUST BE SET IN NAMELIST PROB FOR EACH OPTION

When is output required? ^a	Variables that must be set in namelist PROB
At specified values of the integration variable	PRINT
At specified values of the assigned variable (area or pressure) ^b	APRINT
At intervals of a specified number of integration steps ^c	IPRINT and END

^aSee section 11.3.6 for a fourth method of specifying print stations.

^bIn this case the assigned variable must be specified in tabular form, and that too only as a monotonic function of the integration variable. Also, values must be entered into APRINT in the same order (i.e., increasing or decreasing values) as in ATB.

^cIn this case output is also produced after the problem is completed (i.e., at the value ($\geq \text{END}$) of the independent variable reached on the final integration step).

(12.14) to (12.18), can be computed by setting ROCKET equal to TRUE. If this option is selected, the nozzle throat area ATHROT (in square inches) and the rocket combustion chamber pressure PC (in pounds per square inch absolute) must be supplied.

In order to follow the kinetic process in the flow through a converging-diverging nozzle with the present version of LSENS, the following three steps should be taken:

(1) Initial conditions for the subsonic flow problem in the converging portion of the nozzle (i.e., upstream of the throat) should be obtained by performing equilibrium calculations with a code such as CET (ref. 11).

(2) The kinetically limited pressure profile in this region of the nozzle is assumed to be the same as the equilibrium pressure profile, which is therefore assigned for the flow calculation. Starting with equilibrium conditions at some convenient point, the computation should be carried through the throat until the Mach number just exceeds 1.05.

(3) Finally the resulting conditions from step (2) provide the initial state for the flow problem in the diverging section, whose area profile must be assigned.

For high chamber pressures (e.g., ≥ 60 psia (414 kPa)) this procedure can be speeded up by eliminating step (2); that is, by assuming equilibrium flow up to a point just downstream of the throat (e.g., where the Mach number just exceeds 1.05). The equilibrium results at this point provide the necessary initial conditions for the supersonic assigned-area kinetic computation in the diverging section. The validity of this procedure is quite good, except for very low chamber pressures. For a given chamber pressure the nozzle position where the results of the kinetically limited and equilibrium computations begin to differ significantly from one another can only be determined by comparing the two.

11.3.6 Assigned-Temperature Profile (Namelist TMPDAT)

For an assigned-temperature problem (i.e., TASS = TRUE in PROB) the required data should be set in a namelist called TMPDAT, which must follow PROB. In addition to the temperature profile the namelist contains an option for specifying when output is to be generated. Table 11.9 gives the complete list, and description, of the variables in TMPDAT.

TABLE 11.9.—DESCRIPTION OF NAMELIST TMPDAT

Variable name	Variable type	Explanation
CT0, CT1, CT2, CT3	Real	Coefficients in polynomial equation for temperature T as function of time t : $T = CT0 + CT1*t + CT2*t^2 + CT3*t^3$
CX0, CX1, CX2, CX3	Real	Coefficients in polynomial equation for temperature T as function of distance x : $T = CX0 + CX1*x + CX2*x^2 + CX3*x^3$
TMPTB	Real	Array of temperatures when assigning temperature in tabular form as function of time or distance; maximum number of values, 100
TPRINT	Real	Array of temperatures at which printout is desired; maximum number of print stations, 100 ^a
TTB	Real	Array of times when specifying temperature table as function of time; values correspond to those in TMPTB; maximum number of points, 100
XTB	Real	Array of distances when specifying temperature table as function of distance; values correspond to those in TMPTB; maximum number of points, 100

^aIn this case the temperature must be specified in tabular form, and that too only as a monotonic function of the integration variable. Also, values must be entered into TPRINT in the same order (i.e., increasing or decreasing values) as in TMPTB.

11. Description of Code Usage

For a *flow problem* the temperature can be given as a function of either time or distance independently of the choices of the integration variable and the independent variable for the assigned variable. One of the following two methods can be used to specify the temperature T :

- (1) In tabular form as a function of either time or distance
- (2) As a polynomial function of either time t

$$T = CT0 + CT1 * t + CT2 * t^2 + CT3 * t^3 \quad (11.15)$$

or distance x

$$T = CX0 + CX1 * x + CX2 * x^2 + CX3 * x^3 \quad (11.16)$$

To select method (1), the temperature values must be set in the array TMPTB, and either the corresponding time values must be set in the array TTB or the corresponding distance values in the array XTB. In order to specify a T - t profile by method (2), nonzero values must be given for one or more of the four coefficients CT0, CT1, CT2, and CT3. A T - x profile is given by assigning nonzero values for one or more of the four coefficients CX0, CX1, CX2, and CX3.

For a *static problem* either of the two methods may be used, but temperature must be specified as a function of time. The procedure for selecting the T - t profile is exactly the same as that described for a flow problem.

A *constant-temperature calculation* can be performed by setting a nonzero value for only (1) CT0 for a static problem and (2) CT0 or CX0 for a flow problem. The appropriate variable must be set equal to the desired temperature.

The user can require printout of the solution at specified values of the temperature, but only if it is assigned in tabular form and that too only as a monotonic function of the integration variable. To use this option, the temperature values at which output is required must be entered into the array TPRINT. Also they must be ordered such that the resultant ξ

values at which output is produced increase. That is, if the values in TMPTB increase monotonically, so should the TPRINT values, and vice versa. Finally no print station information must have been given in PROB (see section 11.3.5.4 and table 11.8).

The variables that must be set in TMPDAT depend on whether temperature is a function of time or distance and on the form of the temperature profile. Table 11.10 summarizes the information required for each of the options built into the code.

11.3.7 Perfectly Stirred Reactor Problem Data (Namelist WSPROB)

For a PSR problem (i.e., WELSTR = TRUE and/or WSFLOW = TRUE in PROB) the required data are given in a namelist called WSPROB, which must follow PROB. The type of PSR computation—assigned mass flow rate or assigned temperature—is specified here. In either case a sequence of PSR solutions is generated until the desired condition is reached. The required data for each problem type and heat transfer and output control information are given in this namelist. Table 11.11 gives the complete list of variables in WSPROB and their descriptions, with default values underlined.

To select an *assigned-mass-flow-rate problem*, the variable DELMD must be set equal to the desired mass flow rate increment between successive solutions. Also DOTMAX, the desired mass flow rate through the reactor, must be given.

To select an *assigned-temperature problem*, the variable DELT, the temperature decrement between successive solutions, must be set. (The first solution is attempted with a temperature of $TEQ - DELT$, where TEQ is the equilibrium temperature for the reactor inlet state.) The user must also assign a value for either DTMAX, the maximum temperature decrement, or TMPMIN, the desired temperature at the reac-

TABLE 11.10.—VARIABLES THAT MUST BE SET IN NAMELIST TMPDAT FOR ASSIGNED-TEMPERATURE PROBLEM

Temperature profile	Temperature profile type	Variables that must be set in namelist TMPDAT
Temperature versus time	Constant ^a	CT0
	Polynomial	One or more of CT0, CT1, CT2, and CT3
	Tabular	TMPTB and TTB
Temperature versus distance	Constant ^a	CX0
	Polynomial	One or more of CX0, CX1, CX2, and CX3
	Tabular	TMPTB and XTB

^aSee table 11.6 for an alternative method of specifying a constant-temperature problem.

TABLE 11.11.—DESCRIPTION OF NAMELIST WSPROB

Variable name	Variable type	Value	Explanation
DELMD	Real	----	Mass flow rate increment for assigned-mass-flow-rate problem
DELTD	Real	----	Temperature decrement (absolute value) for assigned-temperature problem
DOTMAX	Real	----	Desired mass flow rate for assigned-mass-flow-rate problem
DTMAX	Real	----	Maximum temperature decrement desired for assigned-temperature problem; temperature at reactor exit will be set equal to (TEQ - DTMAX), where TEQ is the equilibrium temperature for the inlet state
MPR	Integer	----	Control variable for printout: printout occurs after first and, thereafter, every MPRth convergence; default value, 1
RITE	Logical	TRUE <u>FALSE</u>	Extra debug output will be printed on every iteration Only standard output will be printed
TMPMIN	Real	----	Desired temperature at reactor exit for assigned-temperature problem
VOLUME	Real	----	Volume of reactor
WSRHTR	Logical	TRUE <u>FALSE</u>	Reactor is not adiabatic and heat transfer rate is given by polynomial function of temperature (see below) Reactor is adiabatic
WSRHT0, WSRHT1, WSRHT2, WSRHT3, WSRHT4	Real	----	Coefficients in heat transfer rate \dot{Q} versus temperature T expression: $\dot{Q} = \text{WSRHT0} + \text{WSRHT1} * T + \text{WSRHT2} * T^2 + \text{WSRHT3} * T^3 + \text{WSRHT4} * T^4$

tor exit. If DTMAX is given, the reactor exit temperature will be set equal to TEQ - DTMAX. If both DTMAX and TMPMIN are specified, the value min(TMPMIN, TEQ - DTMAX) will be assigned for the exit temperature.

For both problem types the variable VOLUME must give the reactor volume. If the problem is not adiabatic and heat transfer must be considered, the logical variable WSRHTR should be set equal to TRUE. Also nonzero values must be supplied for one or more of the five coefficients in the following equation for the heat transfer rate \dot{Q} as a polynomial function of temperature T :

$$\dot{Q} = \text{WSRHT0} + \text{WSRHT1} * T + \text{WSRHT2} * T^2 + \text{WSRHT3} * T^3 + \text{WSRHT4} * T^4 \quad (11.17)$$

Finally WSPROB contains the variables MPR and RITE, which control, respectively, the frequency and detail of output generated during the PSR calculation. Results of the converged solution are printed after the first and thereafter after every MPRth convergence. If the logical variable RITE is set equal to TRUE, reaction rate information is printed after every iteration. This information may be useful for debugging and is described in chapter 12 (see section 12.5.2 and table 12.2). Additional intermediate information may be obtained by setting DBUGO equal to TRUE in PROB (see section 11.3.5.5).

The problem type (i.e., assigned mass flow rate or assigned temperature) dictates what variables must be set in WSPROB. The data required for both problem types are summarized in table 11.12. The code identifies the problem by examining DELMD and DELTD; therefore nonzero values

TABLE 11.12.—VARIABLES THAT MUST BE SET IN
 NAMELIST WSPROB FOR PERFECTLY STIRRED
 REACTOR PROBLEM^a

Problem type	Variables that must be set in namelist WSPROB ^b
Assigned mass flow rate	DELMD; DOTMAX; MPR; VOLUME; and for problem with heat transfer WSRHTR (= TRUE) and one or more of WSRHT0, WSRHT1, WSRHT2, WSRHT3, and WSRHT4
Assigned temperature	DELTA; DTMAX and/or TMPMIN; MPR; VOLUME; and for problem with heat transfer WSRHTR (= TRUE) and one or more of WSRHT0, WSRHT1, WSRHT2, WSRHT3, and WSRHT4

^aDetailed output will be produced at every iteration if the logical variable DBUGO is set equal to TRUE in namelist PROB (see section 11.3.5.5).

^bReaction rate information will be printed at every iteration if the logical variable RITE is set equal to TRUE.

should not be given for both variables: to do so will result in an error exit.

In the event of convergence to a false solution, the code increases the mass flow rate as part of the restart process. The user may therefore have to increase the original value of DOTMAX significantly to obtain a realistic solution under some conditions. For the same reason the reactor volume and the initial mass flow rate (see section 11.3.8) may also have to be adjusted.

11.3.8 Initial Conditions (and Namelist START)

For all problems the initial (or reactor inlet) state of the mixture must be provided. Unless already assigned in PROB and/or TMPDAT the required thermodynamic and fluid dynamic variables must be initialized in a namelist called START, which will follow PROB, TMPDAT, or WSPROB, depending on the problem type. Time and distance may also be initialized in START. Finally, if the simplified-composition input option is used, the necessary variables are set here; otherwise, the mixture composition must follow START. In the latter case the composition may be input as either mole or mass fractions. The logical variable MOLEF in START tells the code which composition units are used, as explained in table 11.13 (standard options are underlined).

The *simplified-composition input option* is selected by giving either the fuel-oxidant equivalence ratio ERATIO or the fuel-oxidant mass ratio FLAIR. If nonzero values are specified for both ERATIO and FLAIR, only the former will be used and the latter ignored. To use the first method, the stoi-

chiometric coefficients of carbon (SCC), hydrogen (SCH), and oxygen (SCOX) in the fuel must be given. All three variables are initialized to zero, and so the user has to set only the nonzero coefficients. If the oxidant composition is different from that built into the code (see table 8.1), the user must also supply ARAT, CRAT, and NOXRAT, which are the molar ratios of argon, carbon dioxide, and nitrogen to oxygen, respectively. (For example, if the oxidant is pure oxygen, ARAT = CRAT = NOXRAT = 0.0.) The second method requires a value for only FLAIR, unless the oxidant composition is different from the standard one, in which case the oxidant ratios ARAT, CRAT, and NOXRAT must also be given, as well as the oxidant molar mass AIRMW. (For example, if the oxidant is pure oxygen, AIRMW = 31.994.)

Irrespective of the method used, MOLEF must not be set equal to FALSE, and the integration and assigned variables, units, and fuel name line must contain the fuel name (see section 11.3.4). For each of the two methods the variables that must be set in START are summarized in table 11.14.

For *any kinetics problem*, including that following an incident shock or PSR computation, time and distance may be initialized by entering the desired values into TIME and X, respectively. Both variables are initialized to zero before START is read, and so the user needs to specify only the nonzero ones.

Other quantities that have to be initialized in START depend on the problem type and options selected. For convenience, we consider each problem type in a separate subsection and provide a table summarizing the required input information in START.

TABLE 11.13.—DESCRIPTION OF NAMELIST START

Variable name	Variable type	Value	Explanation
AIRMW	Real	----	Molar mass of oxidant; default value, 28.9644
AREA	Double precision	----	Initial area
ARAT	Real	----	Ar/O ₂ mole ratio in oxidant; default value, 0.044487
CRAT	Real	----	CO ₂ /O ₂ mole ratio in oxidant; default value, 0.001432
ERATIO	Real	----	Fuel-oxidant equivalence ratio
FLAIR	Real	----	Fuel-oxidant mass ratio
MACH	Real	----	Initial Mach number or Mach number of gas flowing into incident shock
MDOT	Double precision	----	Mass flow rate for flow problem. For PSR problem mass flow rate either assigned for first solution or to start iteration for assigned-temperature calculation
MMHG	Logical	TRUE <u>FALSE</u>	Pressure P (see below) is given in torr (millimeters of mercury) Pressure P is given in user's chosen input units
MOLEF	Logical	<u>TRUE</u> FALSE	Initial (or reactor inlet) mixture composition will be given in mole fractions Initial (or reactor inlet) mixture composition will be given in mass fractions
NOXRAT	Real	----	N ₂ /O ₂ mole ratio in oxidant; default value, 3.72735
P	Double precision	----	Initial pressure or unshocked gas pressure or reactor pressure
RHO	Double precision	----	Initial density or unshocked gas density or reactor inlet density
SCC	Real	----	Stoichiometric coefficient of carbon in fuel; default value, 0.0
SCH	Real	----	Stoichiometric coefficient of hydrogen in fuel; default value, 0.0
SCOX	Real	----	Stoichiometric coefficient of oxygen in fuel; default value, 0.0
T	Double precision	----	Initial temperature or unshocked gas temperature or reactor inlet temperature
TIME	Double precision	----	Initial time
V	Double precision	----	Initial velocity or shock velocity
X	Double precision	----	Initial distance

TABLE 11.14.—VARIABLES THAT MUST BE SET IN
NAMELIST START IF SIMPLIFIED-COMPOSITION INPUT
OPTION IS USED TO SPECIFY INITIAL MIXTURE
COMPOSITION^a

Quantity used to specify initial mixture composition	Variables that must be set in namelist START ^b
Fuel-oxidant equivalence ratio	ERATIO; SCC; SCH; SCOX; and if oxidant composition is different from the standard one, ARAT, CRAT, and NOXRAT
Fuel-oxidant mass ratio	FLAIR; and if oxidant composition is different from the standard one, AIRMW, ARAT, CRAT, and NOXRAT

^aThe fuel name must be given on the integration and assigned variables, units, and fuel name line (see section 11.3.4).

^bMOLEF must not be set equal to FALSE in START.

11.3.8.1 Static and flow kinetics problems.—We first consider the temperature, which is initialized in START if, and only if, it has not been assigned in TMPDAT (see sections 11.3.5.1.4 and 11.3.6). In this case (i.e., TASS = FALSE), the initial (or constant) temperature must be entered into the variable T. Note that T must be specified even for a constant-temperature calculation selected by means of the flag TCON if a temperature value is not given in TMPDAT. However, if a temperature profile, constant or otherwise, has been prescribed in TMPDAT, T must not be set: to do so will cause an error exit. A simple rule that summarizes these instructions is that namelist TMPDAT and the variable T in START cannot both appear in the problem data file for any problem, irrespective of the value of TCON.

The remaining variables whose initial values must also be set in START are as follows: For a *constant-density static problem* either pressure P or density RHO must be given.

For an *assigned-pressure static problem* no additional variable needs to be set if the pressure profile has been furnished in PROB. However, if the flag PCON had been used in PROB to denote constant pressure, but its value is not given there, either pressure P or density RHO must be set.

For an *assigned-pressure flow problem* with a pressure profile supplied in PROB two of the following three variables must be given: (1) mass flow rate MDOT; (2) area AREA; and (3) either velocity V or Mach number MACH. If the flag PCON had been used in PROB to denote constant pressure but its value is not given there, either pressure P or density RHO must also be set. An alternative that will work in this case is to set all three variables: (1) MDOT; (2) AREA; and (3) either V or MACH.

For an *assigned-area flow problem* with area profile provided in PROB, the user must set two of the following three variables: (1) either pressure P or density RHO; (2) mass flow rate MDOT; and (3) either velocity V or Mach number MACH. If the flag ACON had been used in PROB to denote constant area but its value is not given there, AREA must also be set. Alternatively values may be assigned for the three variables: (1) either P or RHO; (2) MDOT; and (3) either V or MACH.

11.3.8.2 Incident shock problem.—For this problem type values must be given for the following variables: unshocked gas temperature T and either unshocked gas pressure P or unshocked gas density RHO. Also either the variable V must be set equal to the shock velocity or a value assigned for MACH, the ratio of the shock velocity to the sonic velocity in the unshocked gas (i.e., the quantity \mathcal{M}_1 in eq. (9.5)).

11.3.8.3 PSR problem.—For both problem types the reactor inlet temperature T and either reactor pressure P or reactor inlet density RHO must be supplied. The user must also set MDOT, which is the mass flow rate either for the first solution of an assigned-mass-flow-rate problem or to start the iteration for the first solution of an assigned-temperature problem.

11.3.8.4 PSR-plus-plug-flow problem.—For the PSR calculation the user must set T equal to the reactor inlet temperature and MDOT equal to the mass flow rate that is either the first solution of an assigned-mass-flow-rate problem or the initial guess for the first solution of an assigned-temperature problem.

The flow problem following the PSR calculation dictates what other variables must be set in START. For an *assigned-*

pressure flow problem the user must supply a value for one of the following three variables: (1) initial area AREA; (2) initial velocity V; and (3) initial Mach number MACH. In addition, if a pressure profile has not been prescribed in PROB, either reactor pressure P or reactor inlet density RHO must be set. This situation arises when a constant-pressure flow problem is selected through the flag PCON but a pressure value is not given in PROB.

For an *assigned-area flow problem* either reactor pressure P or reactor inlet density RHO must be specified. In addition, if a constant-area flow problem has been specified through the flag ACON but an area value is not given in PROB, the user must set one of the three variables: (1) initial area AREA; (2) initial velocity V; and (3) initial Mach number MACH.

11.3.8.5 Summary.—The thermodynamic and fluid dynamic variables that must be initialized in START depend on the problem type, related options, and information given in PROB and TMPDAT. For each problem type and related option we list these variables in table 11.15. Finally, if the pressure is required, P may be given in either the input units selected by the user or torr (millimeters of mercury). In the latter case the logical variable MMHG must be set equal to TRUE in START.

11.3.8.6 Initial mixture composition.—Immediately following START, the initial (or reactor inlet) mixture composition is specified by listing the species names and their initial (or reactor inlet) mole or mass fractions, depending on the value assigned to MOLEF. The default composition, corresponding to MOLEF = TRUE, is mole fraction. For each species the data are given on a new line, which is read by using the format A8, 2X, F15.13. The species name is written, starting in column 1, exactly as in the chemical reactions. The species mole or mass fraction is written in columns 11 to 25, in either fixed- or floating-point format. All mole and mass fractions are initialized to zero, and so only species with non-zero initial concentration need to be included in the composition list. The end of this list is signaled by a line that either is blank or contains the word END in columns 1 to 3.

Even if the simplified-composition input option is used, the list of species names and mole fractions may be given. This feature enables the user to include some trace species in the initial mixture. However, the sums of the mole and mass fractions of all species must not exceed unity by more than 10^{-4} and 5×10^{-5} , respectively. Finally, if the initial mixture consists of only the fuel and standard oxidant species (i.e., oxygen, nitrogen, argon, and carbon dioxide, see table 8.1), the line following START must either be blank or contain the word END in columns 1 to 3.

11.3.9 Local Error Tolerances and Other Integration Controls (Namelist SOLVER)

All controls for the LSODE package (refs. 20 to 22), which is used to solve the governing ordinary differential

equations, are read in from a namelist called SOLVER, which must follow the initial composition list. For a perfectly stirred reactor problem without a flow problem following it, this namelist is not required, but its inclusion will not cause an error exit. SOLVER contains variables that select the integration and corrector iteration methods. Local error tolerances and other control variables are also set here. The complete list of variables in SOLVER, with brief explanations, is given in table 11.16.

The only two parameters that the user will usually set in SOLVER are EMAX, the local relative error tolerance for all variables, and ATOSLP, the local absolute error tolerance for reacting species mole numbers. The two tolerances control both the speed and accuracy of the integration and must therefore be chosen carefully; see, for example, reference 38, which examines the accuracy of several solvers, including LSODE. The default values for EMAX and ATOLSP favor accuracy over computational efficiency.

Pure relative error control is obtained by setting ATOLSP = 0.0; EMAX is then a measure of the number of accurate significant figures in the numerical solution. Setting EMAX = 0.0 gives pure absolute error control; ATOLSP is then a measure of the largest number that may be neglected. Now for a typical chemical reaction problem the solution components vary widely in magnitude, so that relative error control is appropriate. However, it is undefined when one or more solution components vanish and therefore cannot be used. (For example, in a typical chemical kinetics problem several of the species have zero initial concentration.) Hence mixed relative/absolute error control is used in LSENS, and nonzero values must generally be specified for both EMAX and ATOLSP. However, for a given EMAX we recommend using an ATOLSP that produces mostly relative error control for mole numbers greater than a minimum value, σ_{\min} , that is not considered significant. Such error control can be obtained by setting $ATOLSP < \sigma_{\min} EMAX$. The default ratio ($= 10^{-9}$) for ATOLSP/EMAX provides this error control for mole fractions greater than approximately 0.1 ppm (for a mixture with molar mass of order $O(100)$).

In addition to the ATOLSP/EMAX ratio the actual values for the two variables must be selected with care. Now LSODE controls only an estimate of the local error—the error committed on a single step, starting with data regarded as exact—whereas the quantity that is generally of interest to the user is the global error, or the deviation of the numerical solution from the exact solution. The global error is neither measured nor controlled by the code. Therefore the user is advised to be conservative in selecting EMAX and ATOLSP. However, the computational work generally increases when more accuracy is requested. Some experimentation may be necessary to optimize the tolerance parameters, that is, to find values that produce sufficiently accurate solutions while minimizing execution time. The global errors in solutions generated with particular values for the local error tolerances may be estimated by comparing them with results produced

TABLE 11.15.—SUMMARY OF THERMODYNAMIC AND FLUID DYNAMIC VARIABLES
WHOSE INITIAL VALUES MUST BE SET IN NAMELIST START

Problem type	Assigned variable specified in namelist PROB?	Variables whose initial values must be set in namelist START
Constant-density static	No ^a	TIME ^b ; P or RHO; and T ^c
Constant-pressure static	No ^d	TIME ^b ; P or RHO; and T ^c
	Yes	TIME ^b and T ^c
Assigned-pressure static	Yes	TIME ^b and T ^c
Constant-pressure flow	No ^{d,e}	TIME ^b ; X ^b ; P or RHO; two of MDOT, AREA, V (or MACH); and T ^c
	Yes	TIME ^b ; X ^b ; two of MDOT, AREA, V (or MACH); and T ^c
Assigned-pressure flow	Yes	TIME ^b ; X ^b , two of MDOT, AREA, V (or MACH); and T ^c
Constant-area flow	No ^f	TIME ^b ; X ^b ; AREA; two of P (or RHO), MDOT, V (or MACH); and T ^c
	Yes	TIME ^b ; X ^b ; two of P (or RHO), MDOT, V (or MACH); and T ^c
Assigned-area flow	Yes	TIME ^b ; X ^b ; two of P (or RHO), MDOT, V (or MACH); and T ^c
Incident shock ^g	Yes (see section 11.3.5.1.7)	TIME ^b ; X ^b ; P (or RHO); V (or MACH); and T
Perfectly stirred reactor ^h	None required	P (or RHO); MDOT; and T
Perfectly stirred reactor plus plug flow ^h	See section 11.3.5.1.9	TIME ^b ; X ^b ; P (or RHO) ⁱ ; MDOT; T; and AREA ^j

^aDensity cannot be assigned in PROB.

^bInitial values for time and distance need to be given only if they are different from zero.

^cNot needed if temperature is assigned in namelist TMPDAT (see section 11.3.6).

^dThis combination can be obtained only by setting PCON = TRUE in PROB.

^eAlternatively, initial values may be set for TIME^b; X^b; MDOT; AREA; V (or MACH); and T^c.

^fThis combination can be obtained only by setting ACON = TRUE in PROB. In this case initial values may be assigned for the following variables instead of those given above: TIME^b; X^b; P (or RHO); MDOT; V (or MACH); and T^c.

^gInitial values are for preshock state.

^hThe thermodynamic properties are for the inlet state.

ⁱRequired if area is the assigned variable or a constant-pressure flow problem is specified by means of the PCON switch and pressure is not supplied in PROB.

^jRequired if pressure is the assigned variable or a constant-area flow problem is specified through the option ACON and area is not set in PROB. In either case the initial value of V (or MACH) may be supplied instead of AREA.

TABLE 11.16.—DESCRIPTION OF NAMELIST SOLVER

Variable name	Variable type	Explanation
ATOLSP	Real	Local absolute error tolerance for reacting species; default value, 10^{-14}
EMAX	Real	Local relative error tolerance for all variables; default value, 10^{-5}
HINIT	Real	Step size to be attempted on the first step; computed by LSODE, unless specified here
HMAX	Real	Maximum step size allowed for problem; default value, ∞
HMIN	Real	Minimum step size allowed for problem; default value, 0
MAXORD	Integer	Maximum method order to be used by integrator; default value, 5, for MF = 21
MAXSTP	Integer	Maximum number of steps allowed for problem; default value, 2000
MF	Integer	Method flag; default value, 21, for “stiff” integrator
MXHNIL	Integer	Maximum number of messages printed per problem warning that step size is getting too small; default value, 10
MXSTEP	Integer	Maximum number of (internally defined) steps allowed during one call to LSODE; default value, 500

with smaller tolerances. In reducing the tolerances both EMAX and ATOLSP must be scaled down uniformly.

Another variable that can considerably influence both the accuracy and efficiency of the integrator is the method flag MF, which selects both the integration and corrector iteration methods (see eq. (9.1)). The default value, MF = 21, corresponds to the “stiff” integrator and the Newton-Raphson iteration technique using an analytical Jacobian matrix. For normal code usage we cannot recommend the use of any other MF. The method flag MF = 22 may, however, be used to check the accuracy of the MF = 21 results and, in particular, of the analytical Jacobian matrix (ref. 22). Users wishing to study the effects of other MF values on solution accuracy and efficiency are urged to consult appendix C, where any required coding changes are described. Note, in particular, that if the choice MF = 14 or MF = 24 is made without modifying the code, numerical instability may result. Finally, if the sensitivity analysis option is selected, only MF = 21 is allowed—any other MF will cause an error exit.

Two other variables that the user may either have or wish to adjust are MAXSTP, the maximum number of integration steps allowed for the problem, and MXSTEP, the maximum number of integration steps allowed during one call to LSODE. Exceeding MAXSTP steps without successfully completing the problem is a fatal error, and the execution will be terminated. Exceeding MXSTEP steps during one call to LSODE is, however, not fatal: an error message will be printed and the execution continued normally.

Either of the two problems might be the result of an inappropriate MF. The use of MF = 10 or MF = 20 is one example. These difficulties may also be caused by very stringent accuracy requirements or a very long integration interval during which the system does not equilibrate. The user should also check the problem data file, especially the reaction mechanism, for errors, as described in section 11.3.2.

Either difficulty may also be caused if pure relative error control (i.e., ATOLSP = 0.0) has been selected but most, or all, of the solution components are very small but nonzero.

11. Description of Code Usage

Finally, to track accurately very rapid reactions, such as during heat release, the integrator may be forced to select such small step sizes that these difficulties occur. If the step size becomes too small for the computer, an error message stating this fact is printed. The warning message is printed a maximum number of MXHNIL (default value, 10) times per problem. The user may change this maximum number by setting the integer variable MXHNIL equal to the desired value.

The remaining variables, HINIT, HMAX, HMIN, and MAXORD, allow the user to change several integration controls, which are usually set by LSODE. They are placed in SOLVER solely for the convenience of users who are familiar with LSODE, especially reference 22.

If default values are to be specified for all variables, a dummy line simply indicating the beginning and ending of the namelist may be used.

11.3.10 Sensitivity Analysis Data (and Namelist SENRXN)

If sensitivity analysis is required, the information that tells the code what sensitivity coefficients are to be computed is placed after namelist SOLVER. The information is of three types:

- (1) Names of dependent variables whose sensitivities are desired
- (2) Names of variables with respect to whose initial values sensitivity coefficients are required
- (3) List of reaction numbers with respect to whose rate coefficient parameters sensitivity coefficients are needed and logical variables identifying the parameters.

Type (1) data must always be supplied, but either one or both of data types (2) and (3) may be given. The data types may be given in any order. Each data type is preceded by a keyword line that identifies it.

Type (1) information (i.e., the list of dependent variables) is identified by a line containing the keyword SENSVAR in columns 1 to 7. The variable names are then listed in the next one or more lines, with each line containing up to a maximum of eight names. The format used to read each line is 8(A8, 2X); therefore the variable names must start in columns 1, 11, 21, 31, 41, 51, 61, and 71. In order to request sensitivity coefficients for any reacting species mole number, its name must be given. The species name must be written exactly as in the chemical reactions. The name ALLSP simplifies input by producing the sensitivities of all reacting species mole numbers. The names TEMP, DENSITY, and PRESSURE are required for sensitivity coefficients of temperature, density, and pressure, respectively. If the dependent variable list includes a variable that is constant for the problem (e.g., an inert species mole number), its name will be deleted from the list and sensitivities (which are all constant) of this vari-

able will not be generated. Finally the word ALL in the first three columns of a variable field can be used to generate sensitivities of all nonconstant variables—all reacting species mole numbers and any or all of the quantities temperature, density, and pressure that vary during the problem. The end of the dependent variable list is indicated by either a blank line or the word END in the first three columns of any variable field. If the last line of the list contains exactly eight names, the next line must either be blank or contain the word END in columns 1 to 3.

If initial condition sensitivities are to be computed, the required data must be preceded by a line containing the word INIT in columns 1 to 4. These data, which are names of variables with respect to whose initial values sensitivity coefficients are needed, are written in exactly the same way as the dependent variable names. This list may include names of inert species and other quantities that are constant. Thus the variable name ALLSP will produce sensitivities with respect to the initial mole numbers of all reacting and all inert species. Similarly the name ALL will generate initial value sensitivities for all variables. But because pressure is not an internal variable, sensitivity coefficients with respect to its initial value cannot be requested. However, the normalized sensitivity coefficients with respect to the initial pressure are equal to, and therefore can be obtained from, those with respect to the initial density. Finally, if temperature is assigned as a nonconstant function of time, sensitivities with respect to its initial value cannot be computed.

Type (3) data must be included if sensitivity coefficients with respect to rate coefficient parameters are required. These data, identified by a line containing the keyword REAC in columns 1 to 4, are specified in a namelist called SENRXN, which must follow the keyword line. This namelist contains logical variables for selecting the rate coefficient parameters with respect to which sensitivity coefficients are required. Reactions for which sensitivity analysis is to be performed and the output format for these sensitivity coefficients are also selected here. The namelist variables and their descriptions are given in table 11.17, with the default values underlined.

For a nonconstant-temperature problem any of the logical variables SENSJ, SENSNJ, and SENSEJ may be set equal to TRUE. That is, sensitivity coefficients with respect to any of the rate coefficient parameters A_j , n_j , and E_j or c_j can be computed. However, for a constant-temperature problem the normalized sensitivity coefficients with respect to all rate coefficient parameters are equal to one another (and equal to the normalized sensitivity coefficients with respect to the rate constant, see chapter 4 of part I). Hence sensitivity coefficients with respect to only A_j are produced, regardless of the rate coefficient parameters specified. The reaction numbers for which sensitivity coefficients are to be computed must be entered into the array RXNUM. The logical variable ALLRXN may be set equal to TRUE and RXNUM omitted, if sensitivities are required for all reactions.

TABLE 11.17.—DESCRIPTION OF NAMED LIST SENRXN

Variable name	Variable type	Value	Explanation
ALLRXN	Logical	TRUE	Sensitivities must be computed for all reactions
		<u>FALSE</u>	Sensitivities are required for only reactions listed in RXNUM
ORDER	Logical	TRUE	Nonzero normalized sensitivity coefficients and reaction numbers are to be listed in decreasing order of importance for each rate coefficient parameter
		<u>FALSE</u>	Reaction importance list is not required
OUTPUT	Logical	<u>TRUE</u>	Complete tables of normalized sensitivity coefficients are to be printed for each rate coefficient parameter
		FALSE	Only reaction importance list is to be printed
RXNUM	Real	----	Reaction numbers with respect to whose rate coefficient parameters sensitivities are required; not needed if ALLRXN is set equal to TRUE
SENSAJ	Logical	TRUE	Sensitivities with respect to rate coefficient parameter A_j are required
		<u>FALSE</u>	Sensitivities with respect to A_j are not required
SENSEJ	Logical	TRUE	Sensitivities with respect to rate coefficient parameter E_j (or c_j) are required
		<u>FALSE</u>	Sensitivities with respect to E_j (or c_j) are not required
SENSNJ	Logical	TRUE	Sensitivities with respect to rate coefficient parameter n_j are required
		<u>FALSE</u>	Sensitivities with respect to n_j are not required

As described in chapter 12 (see section 12.3), sensitivity coefficients with respect to rate coefficient parameters can be printed in two ways, indexed by either reaction number or dependent variable name, with the nonzero normalized sensitivity coefficients listed, along with the reaction numbers in decreasing order of importance. The first type of output is the default format; it can be switched off by setting the logical variable OUTPUT equal to FALSE. The second type of output can be obtained by setting the logical variable ORDER equal to TRUE. If the user sets OUTPUT = FALSE but fails to set ORDER = TRUE, the code automatically performs the latter operation so that some output is generated.

11.3.11 FINIS Line

The last line of the problem data file must contain the word FINIS in columns 1 to 5. This line will follow the initial composition list or namelist SOLVER or the sensitivity analysis data, depending on the problem type and whether the sensitivity analysis option is selected.

11.4 Problem Data File for Multiple Cases

If several problems are to be solved during one run, the first case data must follow exactly the same format given in section 11.3 for a single case. For the second and subsequent cases the required data, which must be placed immediately after the first case data, depend on the information, if any, that can be used from the previous case (or cases). The code was designed to minimize the amount of input data, especially when they are the same for two or more consecutive cases. The data that are saved from any one case depend on the value assigned in the next case data for an option called ACTION. The main purpose of providing this option was to eliminate typing in a reaction mechanism when it either is identical, or can be obtained by adding reactions, to that used for the previous case. We now describe, in the order that they must appear in the problem data file, the information that must be provided for each new problem. A summary of the necessary input is given in figure 11.4.

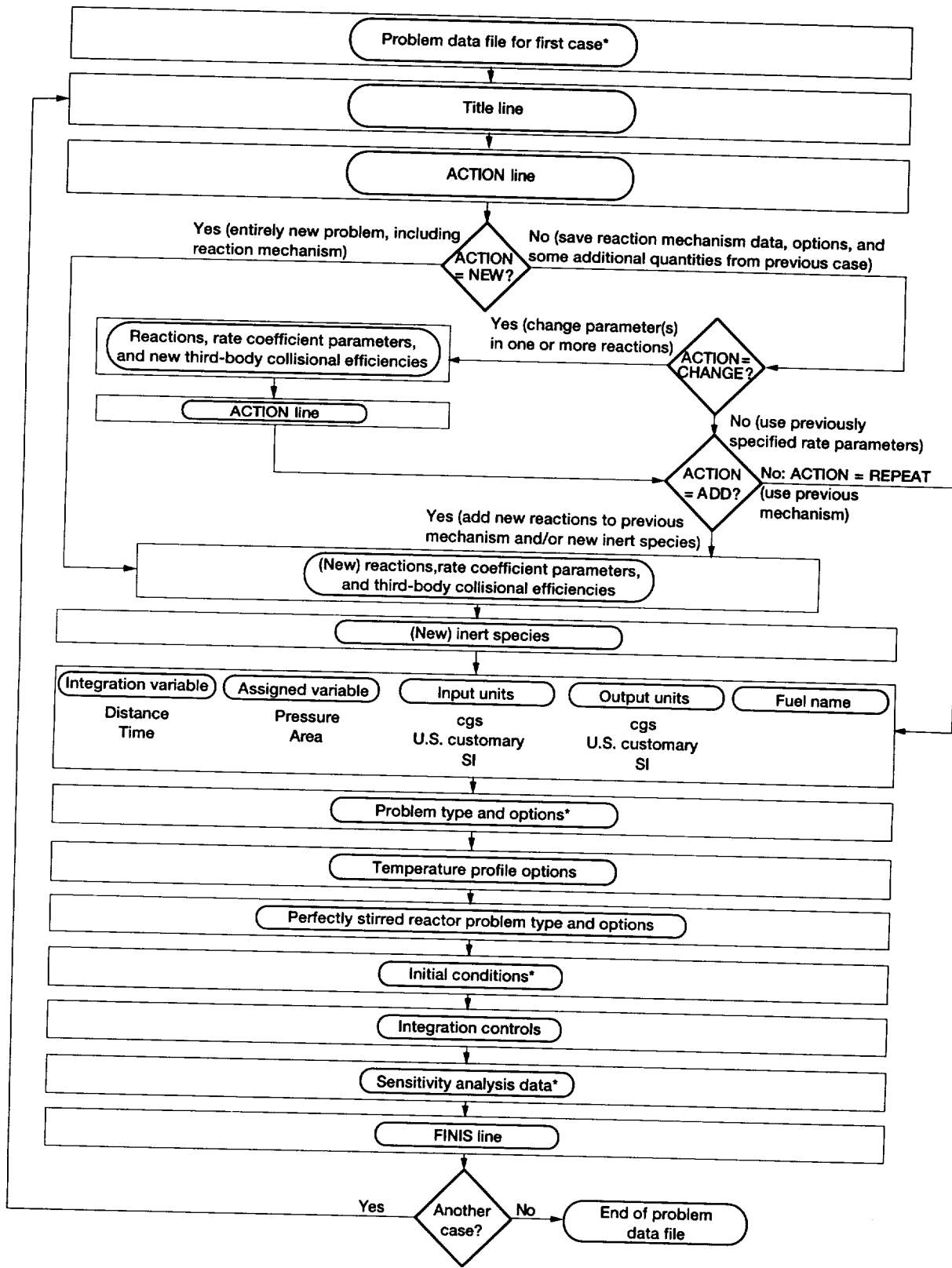


Figure 11.4.—Structure of problem data file for multiple cases. (Details of items marked with asterisk are given in figure 11.3.)

11.4.1 Title Line

The first line following the FINIS line for the previous case contains the title message described in section 11.3.1 and may be blank.

11.4.2 ACTION Line

The next line of the data set must contain the value for the ACTION option, which controls what data, if any, are saved from the previous case. The possible values are NEW, CHANGE, ADD, and REPEAT. One of the four words must be written on this line, starting in column 1.

The word NEW tells the code not to save any data from the previous problem. This option should be used when the reaction mechanism is different from that previously read in and cannot be constructed from it by adding reactions. In this case, all options and variables related to problem specification and control are reinitialized. The remainder of the problem data must be prepared by using exactly the same format as that given in sections 11.3.2 to 11.3.11 for a single case.

If any other value (i.e., CHANGE, ADD, or REPEAT) is assigned for ACTION, the reaction mechanism, the rate coefficient parameters, the third-body efficiencies, and the inert species from the previous case are saved, as are the options ACON, COMBUS, CONC, EXCHR, HTRAN, ORDER, OTTO, PCON, QMREAD, ROCKET, RHOCON, SENCAL, SENSTD, SHOCK, TASS, TCON, WELSTR, WSFLOW, and WSRHTR. In addition, any of the following quantities that were required for the previous problem are saved: ATHROT, BORE, END, HT0, HT1, HT2, HT3, HT4, IPRINT, PC, RPM, STROKE, TINY, TOTMAS, and TWALL; BETA, D, ETA, LSUBM, and VISC; DELMD, DELT, DOTMAX, DTMAX, MPR, TMPMIN, VOLUME, WSRHT0, WSRHT1, WSRHT2, WSRHT3, and WSRHT4; profiles for assigned variable (pressure or area) and temperature; and the print station list. (Note that if a value for any variable not needed for the problem is specified, this variable will not be saved.)

None of the above quantities needs to be reset unless either a new value is required or a different option is selected. In the latter case the previously specified option must be switched off. For example, if the previous problem required the computation of heat transfer rates and so HTRAN had been set equal to TRUE in PROB and the present case is adiabatic, HTRAN must now be set equal to FALSE in PROB. Similarly, if a different heat transfer model is to be used, QMREAD must be reset accordingly, and the data required by the new model supplied. Another illustrative example is as follows: Suppose the previous case had involved a perfectly stirred reactor problem followed by a plug-flow problem, and therefore the variable WSFLOW (and maybe, WELSTR) had been set equal to TRUE in PROB. If the present case does not require a flow problem calculation, WSFLOW must be

set equal to FALSE in PROB, but WELSTR need not be given. Of course, if a totally different problem type is to be considered, both WELSTR and WSFLOW must be set equal to FALSE, and the appropriate option switch supplied.

If a different problem type is specified, any option selected previously that might either be in conflict with, or is not allowed by, the new problem must be switched off; otherwise, an error exit will occur. An example is sensitivity analysis, which can be performed only for a static problem. Thus, if a static problem for which sensitivity analysis had been selected by setting SENCAL equal to TRUE in PROB is followed by a flow problem, SENCAL must now be set equal to FALSE in PROB.

The CHANGE option allows the user to change any rate coefficient parameter and/or any third-body efficiency for any reaction previously given. This option may also be used to change the reversibility of a reaction and/or the form of the rate coefficient expression. In any case the reactions whose parameters are to be modified must be listed, one per line, immediately after the ACTION line. Each reaction must be given exactly as before, using the format given in section 11.3.2. In addition, values must be specified for all three rate coefficient parameters, even those with the same values.

If collisional efficiencies are to be changed for a third-body collisional reaction, lines containing the word THIRDBODY and the species names and efficiencies must follow the reaction, as described in section 11.3.2. Only species whose efficiencies are different from those previously specified need to be listed. Note that the input data must contain any species whose new, but not old, efficiency is unity. Such species will be deleted from the third-body list. Similarly the list will be expanded to include every species whose efficiency is changed from unity. Such species and their new efficiencies must be included in the problem data. Finally a new species—one that was not considered for the previous problem—can be included in the third-body list by taking the following actions: The new species and its third-body efficiency are given, as discussed previously. The ADD option, described later, is then used to add the new species to the species list by means of either new reactions for a reacting species or the inert species list.

However the CHANGE option is used, the end of the reaction list is indicated by a line that either is blank or contains the word END in the first reactant field (i.e., columns 4 to 6). The reaction list must be followed by another ACTION line, which must contain either the word ADD or REPEAT, starting in column 1.

The ADD option enables the user to add new reactions to the previous mechanism and thus possibly new reacting species. If this option is used, the value of the logical variable RXNTST from the previous case is also saved. The new reactions are entered, one per line, immediately after the ACTION line. The information required on each reaction line and its format are identical to those described in section

11. Description of Code Usage

11.3.2. The end of the new reaction list is signaled by a line that either contains the word END in columns 4 to 6 or is blank.

Provision has also been made to add new inert species to the mixture. After the reaction list the next line or lines must contain the new inert species, using the same format as for the first case (see section 11.3.3). Species that were declared inert in the previous case (or cases) need not be listed. The end of the inert species list is indicated by a species field that either is blank or contains the word END in the first three columns. If no new inert species is to be considered, the reaction list must be followed by a line that either contains the word END in columns 1 to 3 or is blank.

The ADD option can be used solely to add new inert species. In this case the ACTION line must be followed by a blank line or one containing the word END in columns 4 to 6. The new inert species list is then entered, as described previously.

The REPEAT option tells the code to use exactly the same reaction mechanism as that for the previous case. The list of reacting and inert species is also unchanged. However, some reaction information may have been changed by means of the CHANGE option.

The following information must follow either the REPEAT line or, if the ADD option has been used, the inert species list.

11.4.3 Integration and Assigned Variables, Units, and Fuel Name Line

None of this information is saved from the previous case; therefore the next line of the problem data file must list the integration and assigned variables, the input and output units, and possibly the fuel name. The format is identical to that used for the first case (see section 11.3.4).

11.4.4 Problem Data, Related Options, Heat Transfer Data, and Output Controls

The variables specifying the problem type and options (i.e., namelist PROB) must be supplied next. (The variables and options that are set in this namelist are described in section 11.3.5.) As discussed in section 11.4.2, previously selected options that are not required or cannot be used must be switched off, and those needed for the new case must be switched on. In addition, new variables required for the problem and previously specified variables with new values must be set. However, previously specified variables that are not needed for this case do not have to be listed. The same rule applies to previously specified variables with the same values as before.

If no new option or variable is required and the previously assigned variables need not be changed, a dummy line simply indicating the beginning and ending of the namelist may be

inserted. It must be remembered that variables which are both required by and introduced into a NEW case and then into successive CHANGE and/or REPEAT (or ADD) cases are saved until overwritten by the new user-supplied values. Thus until the NEW option is again selected, the user has access to the most recently specified value for every quantity that is saved (see section 11.4.2 for list).

11.4.5 Assigned-Temperature Profile

If the temperature is assigned, the namelist TMPDAT, described in section 11.3.6, must follow PROB. The variables that must be set in TMPDAT depend on the options and variables, if any, that were previously assigned in this namelist. If this is the first assigned-temperature problem since the last NEW option was selected, TMPDAT must contain the variables described in section 11.3.6. However, if the temperature had been assigned previously, only new options, new variables, and variables with new values need to be listed. If no new option or variable is required and the previously assigned variables need not be changed, a dummy line simply indicating the beginning and ending of the namelist may be entered.

11.4.6 Perfectly Stirred Reactor Problem Data

If a PSR calculation is required, the namelist WSPROB, described in section 11.3.7, must follow PROB. The variables that must be set in WSPROB depend on the options and variables, if any, that were previously assigned in this namelist. If this is the first PSR problem since the last NEW option was selected, WSPROB must contain the variables described in section 11.3.7. However, if a PSR problem had been specified previously, only new options, new variables, and variables with new values need to be listed. If no new option or variable is required and the previously assigned variables need not be changed, a dummy line simply indicating the beginning and ending of the namelist may be inserted.

11.4.7 Initial Conditions

The initial values of thermodynamic and fluid dynamic variables must be given in namelist START, which must follow PROB, TMPDAT, or WSPROB. No initial condition is saved; therefore the user must supply all the required input data, which depend on the problem type, as discussed in section 11.3.8. The namelist must be followed by the initial mixture composition list, as described in section 11.3.8.6.

11.4.8 Local Error Tolerances and Other Integration Controls

Following the initial composition list, the integration controls are given in namelist SOLVER, if necessary, as

explained in section 11.3.9. If this is the first problem for which SOLVER is either required or given since the last NEW option was selected, the namelist must contain variables for which nondefault values are to be used. However, if integration controls had been assigned previously, only new variables and variables with new values need to be set. If the previously assigned variables, default or otherwise, need not be changed and no new variable is required, a dummy line simply indicating the beginning and ending of the namelist may be given.

11.4.9 Sensitivity Analysis Data

If sensitivity analysis is required for this case, all of the data described in section 11.3.10 must follow SOLVER because none of this information is saved.

11.4.10 FINIS Line

The input data must end with a line containing the word FINIS in columns 1 to 5.

Chapter 12

Description of Output

12.1 Introduction

For all problem types the input information is printed in two ways. First, an exact image of the problem data file, except any thermodynamic coefficients included here, is produced for each case. Blank lines in the input file are identified by the words BLANK LINE. To assist the user in checking input data for errors, the code prints column numbers from 1 to 80 at the top of this image. Second, more detailed input information, including problem type and options selected, is printed, usually in the output units specified by the user. This material is given in the following order:

(1) Lists of elements and reacting and inert species considered for the problem are generated, as are the thermodynamic data, molar mass, heat of formation (in calories per mole) at 25 °C, and, if required for the problem, transport data for each species.

(2) If an equilibrium or shock calculation is performed, the results are printed in the format described in section 12.4.

(3) The reaction mechanism and forward rate coefficient parameters, the third-body efficiencies, and the input and output units are printed.

(4) For a perfectly stirred reactor (PSR) problem the reactor volume, the mass flow rate either for the first solution of an assigned-mass-flow-rate problem or to start the iteration for an assigned-temperature problem, the problem type, and the input data required for the calculation are given. For all other problem types the assigned variable and temperature profiles, the number of ordinary differential equations (ODE's), the integration controls, the sensitivity analysis data, including the normalization factors for any rate coefficient parameters selected, and the output control information are printed.

(5) The heat transfer data if any, the simplified-composition input data if this option is selected, and numbers of reacting and inert species are printed.

(6) The initial values of many variables and their derivatives, described in section 12.2.1 are then listed.

(7) Finally, the central processing unit (CPU) time required so far, that is, to initialize the code, is given in seconds.

The output information related to problem solution depends on the problem type and options selected. For each problem type certain standard material is automatically generated at each print station. For static and flow kinetics and PSR problems optional output may be obtained, either in place of some standard quantities or in addition to regular output. If the sensitivity analysis option is selected, additional variables are printed. We now discuss the solution information that is generated for each problem type and related option.

12.2 Static and Flow Kinetics Problems

12.2.1 Standard Output

For a static or one-dimensional flow problem, including that following either an incident shock or a PSR computation, the following standard information is listed if no option related to output information specification is selected in namelist PROB:

- (1) Reaction time t (gas-particle time for shock reaction); flow cross-sectional area A (for a shock reaction the normalized area A/A_s , where A_s is the shock tube cross-sectional area); and axial position x
- (2) Pressure p ; velocity V ; density ρ ; temperature T ; and mass flow rate \dot{m} (mass flow rate per unit cross-sectional area, i.e., mass flux \dot{m}'' , for a shock reaction)
- (3) The thermodynamic and fluid mechanic properties
 - (a) Mass-specific entropy s

12. Description of Output

$$s = R \sum_{i=1}^{NS} \sigma_i \left(\frac{s_i^\circ}{R} - \ln \sigma_i - \ln p M_w \right) \quad (12.1)$$

where R is the universal gas constant, NS is the total number of (reacting and inert) species, σ_i is the mole number of species i (i.e., moles of species i per unit mass of mixture), s_i° is the standard-state (1 atm) molar-specific entropy of species i , and M_w is the mixture molar mass

$$M_w = \frac{1}{\sum_{i=1}^{NS} \sigma_i} \quad (12.2)$$

(b) Mach number \mathcal{M}

$$\mathcal{M} = \frac{V}{\sqrt{\gamma R T / M_w}} \quad (12.3)$$

where γ is the frozen specific heat ratio of the mixture

$$\gamma = \frac{c_p}{c_p - R / M_w} \quad (12.4)$$

and c_p is the mixture mass-specific heat at constant pressure

$$c_p = \sum_{i=1}^{NS} c_{p,i} \sigma_i \quad (12.5)$$

where $c_{p,i}$ is the constant-pressure molar-specific heat of species i .

(c) Specific heat ratio γ , equation (12.4)

(d) Mixture mass-specific enthalpy h , equation (8.36)

(e) Constant-pressure specific heat of the mixture c_p , equation (12.5)

(4) For a problem with heat transfer the heat transfer rate per unit mass of mixture \dot{Q}/\dot{m} for a static problem or the heat transfer per unit mass of mixture per unit length in the flow direction \dot{Q}'/\dot{m} for a flow problem

(5) Integration indicators

(a) Number of integration steps from last print INST

(b) Average step size since last print AVH

(c) Order of the integration method used on the last step NQ

(d) Total number of integration steps NSTEP

(e) Total number of functional (i.e., derivative) evaluations NFE

(f) Total number of Jacobian matrix evaluations NJE (This number is equal to the total number of LU-decompositions of the iteration matrix.)

(6) For each species i

(a) Molar concentration C_i

$$C_i = \rho \sigma_i \quad (12.6)$$

(b) Mole fraction x_i

$$x_i = \sigma_i M_w \quad (12.7)$$

(c) Net species production rate per unit volume due to all reactions W_i

$$W_i = \sum_{j=1}^{NR} \omega_{ij} \quad (12.8)$$

where NR is the total number of reactions and ω_{ij} , the net molar formation rate of species i per unit volume by reaction j , is given by equation (8.58). If species i is being destroyed, W_i is negative.

(7) For each reaction j the forward rate coefficient k_j , equation (8.3) or (8.4); the net reaction conversion rate X_j , equation (8.56); and the equilibration factor $E_{q,j}$, equation (8.61). For a third-body collisional reaction the k_j printed is for the reference rate coefficient formula.

(8) Derivatives of temperature, density, and for a flow problem velocity with respect to the integration variable

(9) Mixture molar mass M_w , equation (12.2)

(10) Total energy exchange rate X_H

$$X_H = \sum_{j=1}^{NR} X_j \Delta H_{298,j}^\circ \quad (12.9)$$

where $\Delta H_{298,j}^\circ$ is the standard-state enthalpy change for the j th reaction at a temperature of 298 K

$$\Delta H_{298,j}^\circ = \sum_{i=1}^{NS} v_{ij}'' h_i^\circ(298 \text{ K}) - \sum_{i=1}^{NS} v_{ij}' h_i^\circ(298 \text{ K}) \quad (12.10)$$

Here v_{ij}' and v_{ij}'' are, respectively, the stoichiometric coefficients of reactant species i and product species i in reaction j , and $h_i^\circ(298 \text{ K})$ is the standard-state, molar-specific enthalpy of species i at 298 K. The quantity X_H is proportional to the net heat release rate for the

entire complex reaction and may be useful in studying ignition processes.

(11) Mass fraction sum S_m

$$S_m = \sum_{i=1}^{NS} \sigma_i M_{w,i} \quad (12.11)$$

where $M_{w,i}$ is the molar mass of species i . If at any time S_m is not close to unity, we have found that continued integration sometimes produces significantly inaccurate results. Therefore, if S_m is different from unity by more than 5×10^{-5} , an error exit is taken. This error exit means one or more of the following: the integration method is inappropriate, the local error tolerances must be reduced, and the reaction mechanism contains one or more reactions that do not satisfy atom and/or charge balance requirements.

(12) The CPU time in seconds used both since the last output of results ICPU and so far for the problem CPU

For each of the above dimensional quantities, except the average step size, which has the same units as the integration

variable, and the CPU time, which is always given in seconds, we list in table 12.1 the units used on output for the three systems of units included in the code. In this table ξ is the integration variable, time or distance.

After the problem has been successfully completed, the following summary of the computational work required to solve the problem is printed: total number of integration steps, total number of derivative evaluations, total number of Jacobian matrix evaluations and LU-decompositions of the iteration matrix, and total execution time in seconds. In addition, the total CPU time TCPU required for code initialization, input, processing of input and output data, problem solution, and output is given in seconds.

12.2.2 Optional Additional Output

The optional output that is produced depends on the output information specification options selected in namelist PROB (see table 11.5 and section 11.3.5.5). In particular, setting nondefault values for the logical variables CONC, DEBUG, EXCHR, ORDER, and ROCKET produces the following information:

- (1) Species mass fractions $\{y_i\}$

TABLE 12.1.—UNITS FOR OUTPUT VARIABLES IN THE THREE SYSTEMS ALLOWED BY CODE

Variable	Units		
	cgs (internal)	U.S. customary	SI
Time	s	s	s
Area	cm ²	ft ²	m ²
Axial position	cm	ft	m
Pressure	atm	lb/ft ²	N/m ²
Velocity	cm/s	ft/s	m/s
Density	g/cm ³	lb/ft ³	kg/m ³
Temperature	K	°R	K
Mass flow rate	g/s	lb/s	kg/s
Mass flux	g/s-cm ²	lb/s-ft ²	kg/s-m ²
Specific entropy	cal/g-K	Btu/lb-°R	J/kg-K
Specific enthalpy	cal/g	Btu/lb	J/kg
Constant-pressure specific heat	cal/g-K	Btu/lb-°R	J/kg-K
Heat loss term:			
\dot{Q}/m	cal/s-g	Btu/s-lb	J/s-kg
\dot{Q}'/m	cal/g-cm	Btu/lb-ft	J/kg-m
Species molar concentration	mole/cm ³	lb mole/ft ³	kmole/m ³
Species production rate	mole/cm ³ -s	lb mole/ft ³ -s	kmole/m ³ -s
Reaction rate coefficient	cgs units ^a	cgs units ^a	cgs units ^a
Net reaction conversion rate	mole/cm ³ -s-ρ ²	lb mole/ft ³ -s-ρ ²	kmole/m ³ -s-ρ ²
Temperature derivative $dT/d\xi$	K/unit ξ	°R/unit ξ	K/unit ξ
Density derivative $d\rho/d\xi$	g/cm ³ -unit ξ	lb/ft ³ -unit ξ	kg/m ³ -unit ξ
Velocity derivative $dV/d\xi$	cm/s-unit ξ	ft/s-unit ξ	m/s-unit ξ
Mixture molar mass	g/mole	lb/lb mole	kg/kmole
Net energy exchange rate	cal/cm ³ -s-ρ ²	Btu/ft ³ -s-ρ ²	J/m ³ -s-ρ ²

^aUnits for first-order reaction are s⁻¹.
 Units for bimolecular reaction are cm³/mole-s.
 Units for termolecular reaction are cm⁶/mole²-s.

12. Description of Output

$$y_i = \sigma_i M_{w,i}, \quad i = 1, \dots, NS \quad (12.12)$$

are listed in place of molar concentrations $\{C_i\}$, equation (12.6), if CONC is set equal to FALSE.

(2) A complete listing of the $\{\sigma_i\}$, derivatives of $\{\sigma_i\}$ with respect to the integration variable, and $\{\omega_{ij}\}$, equation (8.58), is provided if DBUGO is set equal to TRUE. This information is useful for debugging and is always given in cgs units.

(3) Setting EXCHR equal to TRUE will produce the net energy exchange rates $\{X_{H,j}\}$

$$X_{H,j} = X_j \Delta H_{298,j}^\circ \quad (12.13)$$

instead of the $\{X_j\}$, equation (8.56).

(4) An ordered list of nonzero net formation rates $\{\omega_{ij}\}$, equation (8.58), is printed (in moles per cubic centimeter per second) for each species i and each reaction j if ORDER is set equal to TRUE.

(5) If rocket performance parameters are required for a flow problem (i.e., if ROCKET is set equal to TRUE), the following variables are tabulated:

(a) Specific impulse I_{sp}

$$I_{sp} = \frac{V}{g_c} \quad (12.14)$$

(b) Vacuum specific impulse I_{vac}

$$I_{vac} = I_{sp} + \frac{pA}{\dot{m}} \quad (12.15)$$

(c) Area ratio A_r

$$A_r = \frac{A}{A_t} \quad (12.16)$$

(d) Characteristic velocity c^*

$$c^* = \frac{p_c A_t g_c}{\dot{m}} \quad (12.17)$$

(e) Thrust coefficient c_f

$$c_f = \frac{V}{c^*} \quad (12.18)$$

In these equations g_c is a units conversion factor, A_t is the nozzle throat area, and p_c is the rocket combustion chamber

pressure. Irrespective of the output units selected by the user, I_{sp} and I_{vac} are given in both SI and U.S. customary units, and c^* is printed in feet per second.

12.3 Sensitivity Analysis

Two formats, one default and the other optional, are available for printing the normalized sensitivity coefficients with respect to the rate coefficient parameters. The formats are selected by means of the logical variables OUTPUT and ORDER, as described in section 11.3.10. The default output control, corresponding to OUTPUT = TRUE, produces a table of all normalized sensitivity coefficients $\{\langle S_{ij} \rangle\}$, including those set equal to zero because their magnitudes are smaller than TINY (see section 11.3.5.1.5). The first column of this table gives the reaction numbers, which are listed either in the same order as that specified in the array RXNUM or serially if ALLRXN has been set equal to TRUE (see section 11.3.10). The dependent variable names are printed in the first row. The species, if any, are tabulated in the order that they appear in either the list of dependent variable names or the reaction mechanism if either keyword ALLSP or ALL has been used (see section 11.3.10). They are followed by any nonconstant variables, such as temperature, density, and pressure. Along every subsequent row the reaction number and the normalized sensitivity coefficients of the appropriate dependent variables with respect to the relevant rate coefficient parameter are listed.

At the user's option a differently indexed table is produced, either in addition to, or in place of, the default format. The optional table, selected by setting ORDER equal to TRUE, is indexed by dependent variable names, reaction numbers, and normalized sensitivity coefficients along the rows. The order used for the names is the same as that described for the default format. For each dependent variable the reaction numbers are listed in order of decreasing importance, that is, decreasing magnitude of the $\{\langle S_{ij} \rangle\}$. The normalized sensitivity coefficient of the dependent variable listed in the first column with respect to the appropriate rate coefficient parameter is placed directly under the reaction number. Neither the $\{\langle S_{ij} \rangle\}$ that have been set equal to zero nor the corresponding reaction numbers are printed.

For both output formats a separate table is produced for each rate coefficient parameter selected by the user. The reaction time, the rate coefficient parameter name, and the procedure used to normalize the results with respect to that parameter are given.

Only the default format is used for initial condition sensitivities. The first column of the table gives the names of the variables with respect to whose initial values sensitivities are computed. The dependent variables are listed along the first row. The order used for both quantities is exactly the same as that described earlier for the dependent variables. The sec-

ond and subsequent rows contain the appropriate $\{\langle \dot{S}_{ij} \rangle\}$. The normalization procedures are given for these sensitivities also.

If sensitivity coefficients of the temporal derivatives $\{\langle \dot{S}_{ij} \rangle\}$ are computed, exactly the same output formats as described previously are used. In this case also, the normalization procedures are printed for both initial condition and rate coefficient parameter sensitivities.

After successful completion of the problem summary information regarding the computational work required for sensitivity analysis is given. The following quantities are printed: total number of steps, total number of derivative evaluations, and total number of Jacobian matrix evaluations and LU-decompositions of the iteration matrix. Finally the dimensions of the real and integer work arrays (i.e., total storage) used for integrating the kinetics and sensitivity ODE's are listed.

12.4 Equilibrium and Shock Calculations

The output format is the same for the chemical equilibrium, equilibrium shock, and frozen shock states. First, the initial and final state values and the ratio of final to initial state values are listed for each of the following variables:

- (1) Pressure p
- (2) Gas velocity V
- (3) Density ρ
- (4) Temperature T
- (5) Mixture mass-specific enthalpy h , equation (8.36)
- (6) Mixture mass-specific internal energy u

$$u = h - RT/M_w \quad (12.19)$$

- (7) Constant-pressure specific heat of the mixture $(\partial h/\partial T)_p$, which reduces to equation (12.5) for both initial states and the final state of the frozen shock computations
- (8) Mixture mass-specific entropy s , equation (12.1)
- (9) Mach number \mathcal{M}

$$\mathcal{M} = V/c \quad (12.20)$$

where the sonic velocity c is computed as follows:

$$c = \sqrt{\Gamma RT/M_w} \quad (12.21)$$

where Γ depends on the problem type and the state (i.e., initial or final). For both equilibrium problems the final state Γ is given by

$$\Gamma = \frac{-1}{\left(\frac{\partial \ln v}{\partial \ln p}\right)_T + \left(\frac{\partial \ln v}{\partial \ln T}\right)_p^2 / \frac{M_w}{R} \left(\frac{\partial h}{\partial T}\right)_p} \quad (12.22)$$

where v is the mixture mass-specific volume. For all initial states and the final frozen shock state Γ reduces to the frozen specific heat ratio γ , equation (12.4).

- (10) The quantity Γ defined in equation (12.22)
- (11) The sonic velocity c , equation (12.21)

The initial and final velocities and Mach numbers are zero for the chemical equilibrium calculation. For both variables the ratio of final to initial values is set equal to unity.

The preceding list of variables is followed by the final state mixture composition, expressed as mole fractions $\{x_i\}$, equation (12.7). The following final state thermodynamic properties are then printed: mixture molar mass M_w , equation (12.2); and the derivatives $(\partial \ln v/\partial \ln T)_p$ and $(\partial \ln v/\partial \ln p)_T$. (For the frozen shock problem $(\partial \ln v/\partial \ln T)_p = 1$ and $(\partial \ln v/\partial \ln p)_T = -1$.)

Finally a summary of the computational work required to solve the problem is listed. The quantities printed include the total number of iterations NITER and the execution time CPU in seconds.

12.5 Perfectly Stirred Reactor Problem

12.5.1 Standard Output

For an assigned-mass-flow-rate or assigned-temperature problem the following standard information is listed: First, the results of the equilibrium computation are printed in exactly the same format as that given for a kinetics problem in section 12.2.1. This information is followed by a summary of the computational work required for the equilibrium calculation and the initial estimates for temperature and species mole numbers $\{\sigma_i\}$.

For both problem types the following quantities are listed after the first successful convergence:

- (1) Initial and present values of pressure p ; temperature T ; mixture mass-specific entropy s , equation (12.1); density ρ ; mixture mass-specific enthalpy h , equation (8.36); constant-pressure, mass-specific heat c_p , equation (12.5); molar mass M_w , equation (12.2); and specific heat ratio γ , equation (12.4). The ratio of present to initial values is also given for each variable.
- (2) Initial and present values of both the mole $\{x_i\}$ and mass $\{y_i\}$ fractions of each species, equations (12.7) and (12.12), respectively
- (3) For a problem with heat transfer the heat loss term $\dot{Q}\dot{m}$, where \dot{Q} is the heat loss rate and \dot{m} is the mass flow rate

12. Description of Output

- (4) The reactor volume \mathcal{V} and the mass flow rate
- (5) The ratio \dot{m}/\mathcal{V} , the reactor average residence time τ_r ($= \rho\mathcal{V}/\dot{m}$), and the number of iterations needed for convergence NITER

The above information is also printed after every MPR (see section 11.3.7) successful convergence and, irrespective of MPR, after the last successful convergence for the problem, that is, the one at the desired exit conditions. After the last convergence the total number of iterations and the CPU time in seconds required for the PSR calculation are given. If output is not required after any convergence, the following quantities are printed: convergence number, NCONV; number of iterations; mass flow rate; temperature; and residence time.

12.5.2 Optional Additional Output

The optional additional output that is produced depends on the values assigned for the logical variables DBUGO in namelist PROB and RITE in namelist WSPROB (see sections 11.3.5.5 and 11.3.7). If debug output information has been asked for (i.e., if DBUGO has been set equal to TRUE), the following variables are given on each iteration: iteration number ITER; temperature; mass flow rate; net species production rate per unit volume $\{W_i\}$, equation (12.8); the Newton-Raphson functionals $\{f_i\}$, called $\{TZ(I)\}$ in the output; the sum of the squares of the $\{f_i\}$, FSUM; the FSUM value from the previous iteration PFSUM, called previous FSUM in the output (for the first iteration PFSUM is

assigned a value of zero); either the log correction $\Delta \ln T$ for an assigned-mass-flow-rate problem or $\Delta \ln \dot{m}$ for an assigned-temperature problem; and the sum of the species mole numbers σ_m

$$\sigma_m = \sum_{i=1}^{NS} \sigma_i \quad (12.23)$$

The above list is followed by the species mole numbers $\{\sigma_i\}$ and their log corrections $\{\Delta \ln \sigma_i\}$; the underrelaxation factors Λ and Λ_1 used to control the corrections (see chapter 7 of part I); and the mass fraction sum S_m , equation (12.11).

If reaction rate information is required (i.e., RITE has been set equal to TRUE), the following quantities are printed on every iteration: the net reaction rates per unit volume $\{r_j\}$, equation (8.57); and the species net molar formation rates per unit volume due to each reaction $\{\omega_{ij}\}$, equation (8.58).

Irrespective of the output units selected by the user, $\{r_j\}$ and $\{\omega_{ij}\}$ are given in cgs units. Both quantities are also generated if any PSR calculation is abandoned because successful convergence could not be achieved after 75 iterations.

12.6 Summary of Output Information

For the user's convenience the output information generated for each problem type and option included in the code is summarized in table 12.2.

TABLE 12.2.—SUMMARY OF OUTPUT

Problem type or option	Information type: standard or optional	Keyword and value required to either switch off standard output or switch on optional output	Namelist where keyword appears	Variables printed ^a
Kinetics only (static or one-dimensional flow reaction)	Standard	-----	-----	t, A or $A/A_s, x, p, V, \rho, T, \dot{m}$ or $\dot{m}^{\prime\prime}, s, \mathcal{M}, \gamma, h, c_p, \dot{Q}/\dot{m}$ or $\dot{Q}'/\dot{m},$ INST, AVH, NQ, NSTEP, NFE, NJE, $\{C_i\}, \{x_i\}, \{W_i\}, \{k_j\}, \{X_j\},$ $\{E_{ij}\}, dT/d\xi, dp/d\xi, dV/d\xi, M_w, X_{H_2}, S_m, \text{ICPU, and CPU. At problem end: NSTEP, NFE, NJE, CPU, and TCPU.}$
	Optional	CONC=FALSE. EXCHR=TRUE. ORDER=TRUE. ^b DBUGO=TRUE. ^b ROCKET=TRUE. ^c	PROB PROB PROB PROB PROB	$\{y_i\}$ in place of $\{C_i\}$ $\{X_{H_2}\}$ in place of $\{X_j\}$ Nonzero $\{\omega_{ij}\}$ listed in decreasing magnitude, along with corresponding reaction number j , for each species i $\{\sigma_i\}, \{d\sigma_i/d\xi\},$ and $\{\omega_{ij}\}$ $p_c, A_r, I_{sp}, I_{vac}, c^*, c_f,$ and A/A_s
Kinetics and sensitivity analysis (static reaction only) ^d	Standard	----- OUTPUT=FALSE.	----- SENRXN	$\{S_{ij}\}$ with respect to initial condition values, indexed by variable name $\{S_{ij}\}$ with respect to rate coefficient parameters indexed by reaction number
	Optional	TINY > 0.0 SENSTD=TRUE. ORDER=TRUE.	PROB PROB SENRXN	If $ \{S_{ij}\} < \text{TINY}$, set $\{S_{ij}\} = 0.0$. If $\{\dot{S}_{ij}\}$ required (see below), same rule applies. $\{\dot{S}_{ij}\}$, indexed by initial condition name and, if OUTPUT=TRUE., by reaction number Nonzero $\{\{S_{ij}\}\}$ and, if required, $\{\dot{S}_{ij}\}$, with respect to rate coefficient parameters, listed in decreasing magnitude, along with corresponding reaction numbers
Chemical equilibrium, equilibrium shock, and frozen shock	Standard	-----	-----	Initial and final state values and ratio of latter to former of $p, V, \rho, T, h, u, (\partial h/\partial T)_p, s, \mathcal{M}, \Gamma,$ and c . Final state values of $\{x_i\}, M_w, (\partial \ln v/\partial \ln T)_p,$ and $(\partial \ln v/\partial \ln p)_T$. NITER and CPU.
Perfectly stirred reactor	Standard	-----	-----	Results of equilibrium calculation, in format given above for kinetics-only problem; NITER and CPU required for this computation; and initial estimates for T and $\{\sigma_i\}$. After the first, then every MPRth ^e and final, successful convergence: initial and present values and ratio of latter to former of $p, T, s, \rho, h,$ and c_p ; initial and present values of $M_w, \gamma, \{x_i\},$ and $\{y_i\}; \dot{Q}/\dot{m}, \mathcal{V}, \dot{m}, \dot{m}/\mathcal{V},$ and τ_r ; and NITER. After convergences for which detailed output not required: NCONV, NITER, $\dot{m}, T,$ and τ_r . At end of problem: computational work for PSR calculation.
	Optional	RITE=TRUE. ^{b,f} DBUGO=TRUE. ^{b,f}	WSPROB PROB	After every iteration: $\{\tau_j\}$ and $\{\omega_{ij}\}$ After every iteration: ITER, $T, \dot{m}, \{W_i\}, \{f_i\}, \text{FSUM, PFSUM, } \Delta \ln T$ or $\Delta \ln \dot{m}, \sigma_m, \{\sigma_i\}, \{\Delta \ln \sigma_i\}, \Lambda, \Lambda_1,$ and S_m

^aExcept where indicated, output can be obtained in one of the following three systems of units: cgs (default), U.S. customary, and SI. All CPU times are given in seconds.

^bThis information is given in cgs units.

^cThe quantities $p_c, A_r,$ and c^* are given in U.S. customary units, and I_{sp} and I_{vac} are given in both SI and U.S. customary units.

^dThe information listed for a kinetics-only problem is also generated at every print station. At end of problem NSTEP, NFE, and NJE required for sensitivity computations and the required lengths of the work arrays RWORK and IWORK (see appendix C) are printed.

^eThe integer variable MPR (default value, 1) is set in namelist WSPROB.

^fThese variables are also printed if any PSR calculation is abandoned because successful convergence is not achieved in 75 iterations.

Chapter 13

Examples of Code Usage

In this chapter we illustrate problem data file preparation by means of two kinetics-only test cases and one kinetics-plus-sensitivity-analysis problem. In addition, we examine the effects of local error tolerance parameters on the accuracy of the solutions and computational work for both types of calculations. The information will help the user select error parameters that give the desired compromise between accuracy and computational speed. All results presented in this chapter were generated on the NASA Lewis Research Center's Amdahl 5870 computer using the UTS operating system, the Fujitsu 77 compiler (optimization level, 3), and double-precision arithmetic.

13.1 Kinetics Test Cases

13.1.1 Propane-Air, Perfectly-Stirred-Reactor-Plus-Plug-Flow Problem

The first example problem involves a perfectly stirred reactor (PSR) problem followed by a flow problem. The task is to perform an assigned-mass-flow-rate PSR computation for a fuel-rich (fuel-air equivalence ratio $\phi \approx 2$) propane-air mixture. The air is a mixture of oxygen, nitrogen, argon, and carbon dioxide. The chemical mechanism involves 42 species and 136 reactions. Thermodynamic data for all species are to be taken from the standard thermodynamic data file. The code will be required to check the legality of the reaction mechanism because of its relatively large size.

The reactor volume is 300 cm^3 . The mass flow rate, pressure, and inlet temperature are 1600 g/s , 5 atm , and 614 K , respectively. The inlet composition will be specified by mass fractions. The reactor is not adiabatic and the heat loss rate \dot{Q} (in calories per second) is given by

$$\dot{Q} = -42.88 + 0.05T \quad (13.1)$$

where T is the temperature in Kelvin.

The exit gas from the PSR will be expanded through a diverging 4-cm-long nozzle whose area profile is as follows:

$$A = 15 + 20x \quad (13.2)$$

where A is the area in square centimeters and x is the distance in centimeters. The throat is assumed to be at $x = 0.0 \text{ cm}$ and therefore it has a cross-sectional area of 15 cm^2 (2.325 in.^2). The integration variable for the flow problem is distance. This problem is also nonadiabatic and the heat transfer rate is to be computed by using the built-in correlations and a nozzle wall temperature of 700 K .

Computed results, including rocket performance parameters, are to be printed in cgs units at $x = 0.4, 2, \text{ and } 4 \text{ cm}$. To make sure that the expansion computation is not started when the Mach number is too close to unity, an initial value of $x = 0.2 \text{ cm}$ will be used. However, the initial reaction time is zero. Mass fractions are to be used for printing the mixture composition at the output stations.

13.1.1.1 Problem data file.—Table 13.1 lists the required problem data file, which is now described.

13.1.1.1.1 Thermodynamic data, title line, and reaction mechanism data: Because this test case obtains all thermodynamic data from the standard thermodynamic data file, the first line of the problem data file has the word TAPE starting in column 1. The next line contains an identifying title. It is followed by the reaction mechanism. The hydrogen-oxygen mechanism is the most recent version of the scheme developed by Brabbs and Musiak (ref. 39) and given by Bittker (ref. 40). The nitrogen-oxygen reactions were taken from reference 41, and those involving carbon monoxide and all hydrocarbon species from references 42 and 43.

The blank line after the chemical mechanism signifies the end of the reaction list. The next line lists the inert (i.e., nonreacting) species argon. The next species field is blank, indicating the end of the inert species list.

13.1.1.1.2 Problem setup data: The next line contains the integration and assigned variables for the flow problem

TABLE 13.1.—PROBLEM DATA FILE FOR PROPANE-AIR, PSR-PLUS-FLOW EXAMPLE PROBLEM

TAPE	LSSENS	PROPANE-AIR	WELL-STIRRED	REACTOR +	ROCKET EXP:	CHAPTER 13,	CASE 1	
		C3H8	=	C2H5	+	CH3	5.0E+15 0. 83500.	
		C3H8	=	CH4	+	C3H7	3.55E+12 0. 10300.	
		C3H7	=	C2H4	+	CH3	3.0E+14 0. 33200.	
	M	+	CH4	=	CH3	+	H	2.0E+17 0. 88000.
	H	+	CH4	=	CH3	+	H2	1.26E+14 0. 11900.
	CH4	+	O2	=	CH3	+	HO2	7.94E+13 0. 56000.
	O	+	CH4	=	CH3	+	OH	1.9E+14 0. 11720.
	OH	+	CH4	=	CH3	+	H2O	2.5E+13 0. 5010.
	CH3	+	O2	=	CH3O	+	O	2.4E+13 0. 28680.
	CH3	+	OH	=	CH3O	+	H	6.3E+12 0. 0.
	M	+	CH3O	=	CH2O	+	H	5.0E+13 0. 21000.
	CH3	+	CH3	=	C2H6			2.4E+14 -.4 0.
	H	+	C2H6	=	C2H5	+	H2	1.32E+14 0. 9700.
	O	+	C2H6	=	C2H5	+	OH	1.13E+14 0. 7850.
	OH	+	C2H6	=	C2H5	+	H2O	8.7E+13 0. 3520.
	M	+	C2H5	=	C2H4	+	H	1.0E+17 0. 31000.
	C2H5	+	O2	=	C2H4	+	HO2	2.0E+12 0. 5000.
	H	+	C2H5	=	C2H4	+	H2	4.8E+13 0. 0.
	CH3	+	CH2	=	C2H4	+	H	2.0E+13 0. 0.
	H	+	C2H4	=	H2	+	C2H3	1.5E+14 0. 10200.
	M	+	C2H4	=	C2H2	+	H2	2.6E+17 0. 79300.
	C2H4	+	OH	=	C2H3	+	H2O	4.8E+12 0. 1230.
	C2H4	+	OH	=	CH3	+	CH2O	2.0E+12 0. 960.
	C2H4	+	O	=	CH3	+	HCO	3.3E+12 0. 1130.
	C2H4	+	O	=	CH2O	+	CH2	2.5E+13 0. 5000.
	M	+	C2H3	=	C2H2	+	H	3.0E+15 0. 32000.
	C2H3	+	O2	=	CH2O	+	HCO	3.98E+12 0. -250.
	C2H3	+	H	=	C2H2	+	H2	6.0E+12 0. 0.
	C2H3	+	O	=	C2H2O	+	H	3.3E+13 0. 0.
	C2H3	+	OH	=	C2H2	+	H2O	5.0E+12 0. 0.
	C2H3	+	CH2	=	C2H2	+	CH3	3.0E+13 0. 0.
	C2H3	+	C2H	=	2.0C2H2			3.0E+13 0. 0.
	M	+	C2H2	=	C2H	+	H	4.2E+16 0. 107000.
	C2H2	+	O	=	CH2	+	CO	1.6E+14 .0 9890.
	C2H2	+	O	=	C2HO	+	H	4.0E+14 0.0 10660.
	C2H2	+	OH	=	C2H	+	H2O	6.3E+12 0.0 7000.
	C2H2	+	OH	=	C2H2O	+	H	3.2E+11 0.0 200.
	C2H	+	O2	=	C2HO	+	O	5.00E+13 0. 1500.
	C2H	+	OH	=	C2HO	+	H	2.0E+13 0. 0.
	C2HO	+	O2	=	2.0CO	+	OH	1.46E+12 0. 2500.
	C2HO	+	O	=	2.0CO	+	H	1.202E+12 0. 0.
	C2HO	+	OH	=	2.0HCO			1.0E+13 0. 0.
	C2HO	+	H	=	CH2	+	CO	5.0E+13 0. 0.
	C2HO	+	CH2	=	C2H3	+	CO	3.0E+13 0. 0.
	C2HO	+	CH2	=	CH2O	+	C2H	1.0E+13 0. 2000.
		2.0C2HO	=	C2H2	+	2.0CO		1.0E+13 0. 0.
	C2H2O	+	OH	=	CH2O	+	HCO	2.8E+13 0. 0.
	C2H2O	+	OH	=	C2HO	+	H2O	7.5E+12 0. 3000.
	C2H2O	+	H	=	CH3	+	CO	1.13E+13 0. 3428.
	C2H2O	+	H	=	C2HO	+	H2	7.5E+13 0. 8000.
	C2H2O	+	O	=	C2HO	+	OH	5.0E+13 0. 8000.
	C2H2O	+	O	=	CH2O	+	CO	2.0E+13 0. 0.
	M	+	C2H2O	=	CH2	+	CO	2.0E+16 0. 60000.
	C2H	+	O	=	CO	+	CH	5.0E+13 0. 0.
	CH3O	+	O2	=	CH2O	+	HO2	1.0E+13 0. 7170.
	CH3O	+	H	=	CH2O	+	H2	2.0E+13 0. 0.
	M	+	CH2O	=	HCO	+	H	5.0E+16 0. 81000.
	CH2O	+	OH	=	HCO	+	H2O	3.0E+13 0. 1200.
	CH2O	+	H	=	HCO	+	H2	2.5E+13 0. 3990.
	CH2O	+	O	=	HCO	+	OH	3.5E+13 0. 3510.
	CH3	+	CH2O	=	CH4	+	HCO	1.0E+10 0.5 6000.
	CH3	+	HCO	=	CH4	+	CO	3.0E+11 .5 0.
	CH3	+	HO2	=	CH3O	+	OH	2.0E+13 0. 0.
	M	+	CH3	=	CH2	+	H	1.95E+16 0. 91600.
	H	+	CH3	=	H2	+	CH2	2.7E+11 .67 25700.
	O	+	CH3	=	OH	+	CH2	1.9E+11 .68 25700.
	OH	+	CH3	=	H2O	+	CH2	2.7E+11 .67 25700.
	CH	+	CO2	=	HCO	+	CO	3.7E+12 0. 0.
	CH	+	O2	=	HCO	+	O	1.0E+13 0. 0.
	CH2	+	O2	=	CH2O	+	O	5.0E+11 0.5 6960.
	CH2	+	O	=	CH	+	OH	2.0E+11 .7 25800.
	CH2	+	OH	=	CH	+	H2O	5.0E+11 .5 5900.
	CH2	+	H	=	CH	+	H2	3.2E+11 0.7 4970.
	CH2	+	CH2	=	C2H3	+	H	5.0E+12 0. 0.

TABLE 13.1.—Continued.

CH2	+	CH2	=	C2H2	+	H2	4.0E+13	0.	0.
HCO	+	O2	=	CO	+	HO2	3.0E+13	0.	0.
HCO	+	O	=	CO	+	OH	3.0E+13	0.	0.
HCO	+	OH	=	CO	+	H2O	3.0E+13	0.	0.
HCO	+	H	=	CO	+	H2	2.0E+13	0.	0.
M	+	HCO	=	H	+	CO	2.9E+14	0.	15570.
CO	+	O	=	CO2	+	M	2.4E+15	0.	4100.
CO	+	O2	=	CO2	+	O	2.5E+12	0.	47690.
CO	+	OH	=	CO2	+	H	4.17E+11	0.	1000.
CO	+	HO2	=	CO2	+	OH	5.75E+13	0.	22930.
O	+	H2O	=	OH	+	OH	6.8E+13	0.	18365.
H	+	O2	=	OH	+	O	1.89E+14	0.	16400.
O	+	H2	=	OH	+	H	4.20E+14	0.	13750.
H	+	HO2	=	H2	+	O2	7.28E+13	0.	2126.
O	+	HO2	=	OH	+	O2	5.0E+13	0.	1000.
HO2	+	OH	=	H2O	+	O2	8.0E+12	0.	0.
H	+	HO2	=	2.0OH			1.34E+14	0.	1070.
H2	+	HO2	=	H2O2	+	H	7.91E+13	0.	25000.
OH	+	H2O2	=	H2O	+	HO2	6.1E+12	0.	1430.
HO2	+	HO2	=	H2O2	+	O2	1.8E+12	0.	0.
H	+	H2O2	=	OH	+	H2O	7.8E+11	0.	0.
M	+	H2O2	=	2.0OH			1.44E+17	0.	45510.
THIRDBODY									
H2		2.30	O2		.78	H2O	6.0	H2O2	6.6
END									
H2	+	OH	=	H2O	+	H	4.74E+13	0.	6098.
H	+	O2	=	HO2	+	M	1.46E+15	0.	-1000.
THIRDBODY									
O2		1.30	N2		1.3	H2O	21.3	CO2	7.0
END									
M	+	H2O	=	H	+	OH	1.30E+15	0.	105140.
THIRDBODY									
H2		4.00	O2		1.5	H2O	20.0	N2	1.5
CO2		4.0	END						
H	+	O	=	OH	+	M	7.1E+18	-1.	0.
M	+	H2	=	H	+	H	2.2E+14	0.	96000.
THIRDBODY									
H2		4.10	O2		2.0	H2O	15.0	N2	2.0
END									
M	+	O2	=	O	+	O	1.80E+18	-1.	118020.
CH	+	N2	=	HCN	+	N	1.0E+11	0.	19000.
CN	+	H2	=	HCN	+	H	6.0E+13	0.	5300.
O	+	HCN	=	OH	+	CN	1.4E+11	.68	16900.
OH	+	HCN	=	HNCO	+	H	4.0E+11	0.	2800.
CN	+	O	=	CO	+	N	1.2E+13	0.	0.
CN	+	OH	=	NCO	+	H	2.5E+14	0.	6000.
H2	+	NCO	=	HNCO	+	H	1.0E+14	0.	9000.
HNCO	+	H	=	NH2	+	CO	1.0E+14	0.	8500.
CN	+	O2	=	NCO	+	O	3.2E+13	0.	1000.
CN	+	CO2	=	NCO	+	CO	3.7E+12	0.	0.
O	+	NCO	=	NO	+	CO	2.0E+13	0.	0.
N	+	NCO	=	N2	+	CO	1.0E+13	0.	0.
H	+	NCO	=	NH	+	CO	2.0E+13	0.	0.
CH	+	NO	=	N	+	HCO	1.6E+13	0.	9940.
CH	+	NO	=	O	+	HCN	2.0E+12	0.	0.
NH	+	OH	=	N	+	H2O	5.0E+11	0.5	2000.
HO2	+	NO	=	NO2	+	OH	2.09E+12	0.	-477.
O	+	NO2	=	NO	+	O2	1.0E+13	0.	596.
NO	+	O	=	NO2	+	M	5.62E+15	0.	-1160.
NO2	+	H	=	NO	+	OH	3.47E+14	0.	1470.
NO	+	H	=	N	+	OH	2.63E+14	0.	50410.
NO	+	O	=	N	+	O2	3.8E+9	1.	41370.
O	+	N2	=	NO	+	N	1.80E+14	0.	76250.
N	+	NO2	=	2.0NO			4.0E+12	0.	0.
M	+	N2O	=	N2	+	O	6.92E+23	-2.5	65000.
O	+	N2O	=	N2	+	O2	1.0E+14	0.	28020.
O	+	N2O	=	2.0NO			6.92E+13	0.	26630.
N2O	+	H	=	N2	+	OH	7.59E+13	0.	15100.
NO2	+	H2	=	HNO2	+	H	2.4E+13	0.	29000.
OH	+	NO2	=	HNO3	+	M	3.0E+15	0.	-3800.
THIRDBODY									
O2		0.70	H2		1.4	END			
OH	+	NO	=	HNO2	+	M	5.6E+15	0.	-1700.
HNO	+	H	=	H2	+	NO	5.0E+12	0.	0.
H	+	NO	=	HNO	+	M	5.4E+15	0.	-600.

TABLE 13.1.—Concluded.

```

HNO      +   OH      =   H2O      +   NO      3.6E+13   0.      0.
AR
DISTANCE AREA      CGS      CGS
&prob welstr=.true., conc=.false., wsflow=.true., rxntst=.true.,
      cx0=15.0, cx1=20.0, print=0.4,2.0,4.0, rocket=.true., htran=.true.,
      qmread=.false., twall=700.0, pc=73.5, athrot=2.325, &end
&wsprob dotmax=1600.0, delmd=800.0, mpr=1, volume=300.0,
      wsrht=.true., wsrht1=0.05, wsrht0=-42.88, &end
&start t=614.0, p=5.0, mdot=10.0, molef=.false., x=0.2, &end
C3H8      0.0873262
N2        0.6892887
O2        0.211232
AR        0.011737
CO2      0.0004162
END
&solver emax=1.0E-4, atolsp=1.0E-13, &end
FINIS

```

following the PSR computation and the input and output units. The words DISTANCE, starting in column 1, and AREA, starting in column 11, tell the code that an assigned-area calculation will be performed with distance as the integration variable. Note that only the second field in this line has to contain information because distance is the integration variable, individual species mass fractions at the reactor inlet will be specified, and cgs units are used for both input and output. However, for illustrative purposes we have included the integration variable and both input and output units: the word CGS starting in columns 21 and 31, respectively.

Next the problem data file contains namelist PROB, in which the variables WELSTR and WSFLOW are set equal to TRUE to indicate, respectively, that a PSR problem and a flow problem following it are to be solved. (Note that if WSFLOW = TRUE, WELSTR need not be set and has been included for illustrative purposes only.) The variable CONC is set equal to FALSE so that for the flow problem mass fractions will be printed (instead of the default molar concentrations). The variable RXNTST is set equal to TRUE, so that the reaction mechanism legality test will be performed. The area profile for the flow problem is specified by means of the variables CX0 (= 15.0) and CX1 (= 20.0), which together indicate that area is a linear function of distance. The distance values at which output is required are entered into the array PRINT. Because rocket performance parameters are required, ROCKET is set equal to TRUE and values are supplied for the combustion chamber pressure PC (in pounds per square inch absolute) and the nozzle throat area ATHROT (in square inches). The chamber pressure will be considered equal to the PSR pressure, and so PC = 73.5 psia (5 atm). Finally HTRAN is set equal to TRUE to indicate that heat loss from the nozzle is to be considered. Because the built-in correlations are to be used for \dot{Q} , QMREAD is set equal to FALSE and the wall temperature is entered into TWALL.

Because a PSR computation is required, PROB is followed by namelist WSPROB. The variable DELMD (= 800 g/s), the mass flow rate increment for successive convergences,

tells the code that an assigned-mass-flow-rate calculation is to be performed. The desired mass flow rate (1600 g/s) through the PSR is entered into DOTMAX. The volume of the reactor is given by VOLUME. The variable WSRHTR is set equal to TRUE, indicating that the problem is not adiabatic and that the heat loss rate must be computed. The coefficients for the \dot{Q} calculation are the variables WSRHT0 and WSRHT1. The value MPR = 1 tells the code that PSR results are to be printed after every convergence. (Although the default value for MPR is unity, we have included this variable for demonstration purposes.)

13.1.1.1.3 Reactor inlet and initial conditions: Next, namelist START gives the reactor inlet temperature T and the mass flow rate MDOT for the first solution. Because the PSR solution is started with conditions close to the equilibrium state (see chapter 9, section 9.8), we assign a small value (10 g/s) for MDOT. The reactor pressure P is set because an assigned-area flow problem follows the PSR problem. Because area was specified in PROB, START does not list any of the three quantities AREA, V, and MACH. The variable MOLEF is set equal to FALSE to indicate that the mixture composition at the reactor inlet will be specified by means of mass fractions (instead of the default mole fractions).

For reasons given in section 13.1.1 we set X = 0.20 for the initial value of distance for the flow problem. Finally, because the initial reaction time for this problem is zero, TIME is not given.

Namelist START is followed by the species names and their mass fractions at the reactor inlet. The end of the species list is denoted by a line containing the word END in the first three columns.

13.1.1.1.4 Integration controls for flow problem: Namelist SOLVER follows the initial mixture composition and contains values for the local relative (EMAX) and absolute (ATOLSP) error tolerances for the flow problem.

13.1.1.1.5 FINIS line: The last line contains the word FINIS starting in column 1 to indicate the end of the case data.

13.1.1.2 Sample results.—Table 13.2 gives some of the computed results for the PSR problem and for the flow problem at $x = 4$ cm. The PSR solution required 22 iterations and 1.2 s of CPU time. The computational work for the flow problem was as follows: 158 steps, 203 derivative evaluations, 30 Jacobian matrix evaluations, and 4.2 s of CPU time.

13.1.2 Hydrogen-Air Flow Problem and Accuracy of Integration Method

When using any numerical integration procedure such as LSODE, the user must have an idea of the effects of the local error tolerance parameters on the results. The error control should be set tight enough to give the desired accuracy with the smallest amount of computational work. To examine how the computational work and solution accuracy are influenced by the local error tolerances, we study the ignition of a stoichiometric hydrogen-air mixture in supersonic flow.

The chemical mechanism is the H-O-N reaction subset of the propane-air reactions used for the first example problem.

Because the mechanism for that case was found to be legal, we will not ask for mechanism legality testing here. However, we want to change temporarily the thermodynamic data for the hydroperoxyl radical (HO_2) to reflect a new value for its heat of formation at 298.15 K.

A stoichiometric hydrogen-air mixture at an initial temperature of 1559 K and an initial pressure of 0.956 atm starts flowing (at $x = 0$ and $t = 0$) in a constant-area (2000 cm^2) duct at a Mach number of 5.0. The integration variable is distance and the solution is required at $x = 3.048, 6.096, 7.620,$ and 12.19 cm. The initial mixture composition is to be specified by using the simplified-composition input option. In particular, the equivalence ratio will be given, and the oxidant composition is the same as that built into the code. Also, the initial mixture will consist of only the fuel and the standard oxidant species (i.e., it will not contain any trace species).

The flow is nonadiabatic, with heat loss rate per unit distance \dot{Q}' (in calories per second per centimeter) as a function of temperature, given by

TABLE 13.2.—COMPUTED RESULTS FOR PROPANE-AIR, PSR-PLUS-FLOW EXAMPLE PROBLEM

Variable	PSR problem ($\dot{m} = 1600 \text{ g/s}$)	Flow problem ($x = 4 \text{ cm}$)
Reaction time, s	-----	2.108×10^{-5}
Pressure, atm	5.0	0.311
Velocity, m/s	-----	2057.4
Temperature, K	2148.7	1204.1
Density, g/cm^3	7.361×10^{-4}	8.186×10^{-5}
Specific heat, cal/g-K	0.3661	0.3334
Specific heat ratio	1.264	1.298
Species mass fractions		
C_3H_8	3.933×10^{-5}	3.618×10^{-7}
CH_3	1.006×10^{-3}	1.442×10^{-4}
CH_4	7.496×10^{-4}	6.337×10^{-4}
C_2H_4	3.306×10^{-4}	2.073×10^{-4}
H	1.811×10^{-4}	9.783×10^{-5}
H_2	3.674×10^{-3}	3.877×10^{-3}
O_2	2.705×10^{-3}	7.138×10^{-4}
HO_2	4.010×10^{-6}	4.010×10^{-7}
O	9.398×10^{-5}	4.246×10^{-6}
OH	1.268×10^{-3}	4.445×10^{-5}
H_2O	0.1014	0.1028
CH_2O	7.772×10^{-5}	3.408×10^{-5}
C_2H_6	3.677×10^{-5}	1.356×10^{-4}
C_2H_2	2.671×10^{-3}	2.542×10^{-3}
CO	0.1073	0.1081
C_2HO	8.319×10^{-5}	2.913×10^{-6}
CO_2	7.710×10^{-2}	7.924×10^{-2}
H_2O_2	1.942×10^{-7}	1.464×10^{-8}
N_2	0.6892	0.6892
HCN	2.937×10^{-5}	2.977×10^{-5}
HNCO	6.680×10^{-6}	2.930×10^{-6}
NH_2	2.934×10^{-5}	3.117×10^{-5}
NO	1.245×10^{-4}	1.255×10^{-4}

13. Examples of Code Usage

$$\dot{Q}' = -42.88 + 5.863T \quad (13.3)$$

where T is the temperature in Kelvin. Finally, the input data will be given in cgs units, and output information must be supplied in the same units.

13.1.2.1 Description of problem data file.—The problem data file for this case is shown in table 13.3. The first line of the file contains the word CARD in the first four columns to tell the code that thermodynamic data for some (or all) species are provided in the problem data file. The next four lines give the required data for HO₂. Because this is the only species for which thermodynamic data are to be furnished in the problem data file, the next line contains the word END in columns 1 to 3. The title for the problem follows this information.

Then we give the reaction list, which is ended by using a different method than that used in the previous case. The word END is written in columns 4 to 6 of the line following the last reaction. The combustion air contains small amounts of CO₂ and argon. Because we have chosen not to include in the mechanism any reaction involving CO₂, the next line of the file contains the names of the two inert species CO₂ (= CO2) and Ar (= AR). For this case we use a different method of ending the list of inert species than in the previous example. Instead of leaving the third species field blank, we write END in its first three columns.

The integration and assigned variables, units, and fuel name line contains a blank first field and the word AREA in the second field because we are performing an assigned-area flow problem with distance as the integration variable. Because the simplified-composition input option is being used, the fuel name (H2) must be given on this line, starting in column 41. Finally, because both input and output units are cgs, columns 21 to 40 are blank.

In namelist PROB we indicate a constant-area profile by setting a nonzero value for only CX0 (= 2000.0). The desired print stations are entered into the array PRINT. Finally HTRAN and QMREAD are set equal to TRUE to indicate that heat transfer rates are to be computed for the problem and that the necessary coefficients will be provided by the user. The two coefficients in the \dot{Q}' expression, equation (13.3), are listed as the variables HT0 and HT1. (We have included QMREAD for illustrative purposes only. Its default value is TRUE.)

Namelist START is next and the variables P (initial pressure), T (initial temperature), and MACH (initial Mach number) provide the necessary initial thermodynamic and fluid mechanic properties. Because the initial distance and reaction time are both zero, neither TIME nor X is set. The nonzero value for ERATIO (= 1.0) indicates that the simplified-composition input option is being used by means of the fuel-oxidant equivalence ratio. The latter fact means that we must also give the nonzero stoichiometric coefficients of carbon (SCC), hydrogen (SCH), and oxygen (SCOX) in the fuel.

Because the fuel is hydrogen (H₂), SCH is set equal to 2.0. (For illustrative purposes we have assigned a value of zero for the other two coefficients.) Because the equivalence-ratio option requires the (default) value of TRUE for MOLEF, this variable is not given. Finally, because the built-in oxidant composition is to be used, the variables ARAT, CRAT, and NOXRAT are not set.

No individual initial mole fraction is listed after namelist START because the initial mixture does not contain any trace species. A blank line, or one containing the word END in the first three columns, must nevertheless follow START, as shown in table 13.3.

The data file ends with namelist SOLVER and the usual FINIS line. Because default values are to be used for all variables, including EMAX (= 10⁻⁵) and ATOLSP (= 10⁻¹⁴), SOLVER contains only the beginning and ending of the namelist.

13.1.2.2 Computed results and error considerations.—In order to examine the dependences of solution accuracy and computational work on the local error tolerances, this example problem was solved with various values for EMAX, which was progressively decreased from 10⁻² to 10⁻⁹ by a factor of 10. For this study ATOLSP was set equal to 10⁻⁹EMAX.

The effects of EMAX can be ascertained from table 13.4, which gives selected results at $x = 6.096$ cm. The computational work, as measured by the first four quantities in this table, increased significantly as EMAX was decreased. However, the differences in the computed solutions were quite small. Temperature decreased by about 0.1 percent, and the largest change in a species mole fraction was for nitric oxide (NO), which decreased by about 3 percent when EMAX was reduced from 10⁻² to 10⁻⁹. All other mole fractions changed by less than 1 percent. Table 13.4 shows that the computed results become essentially tolerance independent at EMAX = 10⁻⁴. In fact, for this problem LSODE is quite accurate even for EMAX = 10⁻². However, because this may not always be the case, it is usually necessary to find the optimal EMAX for any new problem by a few trial computations.

The effects of ATOLSP on solution accuracy and computational efficiency were studied by progressively increasing ATOLSP from 10⁻¹⁴ to 10⁻⁷ by a factor of 10. All results were generated with EMAX = 10⁻⁵. Selected results at $x = 6.096$ cm are presented in table 13.5. The computational work generally decreased when ATOLSP was increased. The results were relatively insensitive to ATOLSP in the range 10⁻¹⁴ to 10⁻⁹. However, for ATOLSP ≥ 10⁻⁸ they deviated from their accurate values: for ATOLSP = 10⁻⁷ the solution was significantly inaccurate, with several species concentrations being less than zero. Also no heat release had occurred and the temperature was little changed from its initial value. The same behavior was observed at $x = 12.19$ cm: the temperature was 1556 K, instead of 2683 K (the accurate value). For this ATOLSP the maximum local

TABLE 13.3.—PROBLEM DATA FILE FOR HYDROGEN-AIR SUPERSONIC FLOW EXAMPLE PROBLEM

```

CARD
  300.000 1000.000 5000.000
HO2 J 9/78H 1.0 2. 0. 0.G 300.000 5000.000 33.00669 1
0.40173060E 01 0.22175883E-02-0.57710171E-06 0.71372882E-10-0.36458591E-14 2
-0.11412445E 04 0.37846051E 01 0.35964102E 01 0.52500748E-03 0.75118344E-05 3
-0.95674952E-08 0.36597628E-11-0.89333502E 03 0.66372671E 01 0.00000000 4
END
LSENS STOICH HYDROGEN-AIR TEST WITH HEAT TRANSFER: CHAPTER 13, CASE 2
O + H2O = OH + OH 6.8E+13 0. 18365.
H + O2 = OH + O 1.89E+14 0. 16400.
O + H2 = OH + H 4.20E+14 0. 13750.
H + HO2 = H2 + O2 7.28E+13 0. 2126.
O + HO2 = OH + O2 5.0E+13 0. 1000.
HO2 + OH = H2O + O2 8.0E+12 0. 0.
H + HO2 =2.OOH 1.34E+14 0. 1070.
H2 + HO2 = H2O2 + H 7.91E+13 0. 25000.
OH + H2O2 = H2O + HO2 6.1E+12 0. 1430.
HO2 + HO2 = H2O2 + O2 1.8E+12 0. 0.
H + H2O2 = OH + H2O 7.8E+11 0. 0.
M + H2O2 =2.OOH 1.44E+17 0. 45510.
THIRDBODY
H2 2.30 O2 .78 H2O 6.0 H2O2 6.6
END
H2 + OH = H2O + H 4.74E+13 0. 6098.
H + O2 = HO2 + M 1.46E+15 0. -1000.
THIRDBODY
O2 1.30 N2 1.3 H2O 21.3 H2 3.0
END
M + H2O = H + OH 1.30E+15 0. 105140.
THIRDBODY
H2 4.00 O2 1.5 H2O 20.0 N2 1.5
END
H + O = OH + M 7.1E+18 -1. 0.
M + H2 = H + H 2.2E+14 0. 96000.
THIRDBODY
H2 4.10 O2 2.0 H2O 15.0 N2 2.0
END
M + O2 = O + O 1.80E+18 -1. 118020.
HO2 + NO = NO2 + OH 2.09E+12 0. -477.
O + NO2 = NO + O2 1.0E+13 0. 596.
NO + O = NO2 + M 5.62E+15 0. -1160.
NO2 + H = NO + OH 3.47E+14 0. 1470.
NO + O = N + O2 3.8E+9 1. 41370.
O + N2 = NO + N 1.8E+14 0. 76250.
NO + H = N + OH 2.63E+14 0. 50410.
M + N2O = N2 + O 6.92E+23 -2.5 65000.
O + N2O = N2 + O2 1.0E+14 0. 28020.
O + N2O =2.ONO 6.92E+13 0. 26630.
N + NO2 =2.ONO 4.0E+12 0. 0.
N2O + H = N2 + OH 7.59E+13 0. 15100.
NO2 + H2 = HNO2 + H 2.4E+13 0. 29000.
OH + NO2 = HNO3 + M 3.0E+15 0. -3800.
THIRDBODY
O2 0.70 H2 1.4 END
OH + NO = HNO2 + M 5.6E+15 0. -1700.
HNO + H = H2 + NO 5.0E+12 0. 0.
H + NO = HNO + M 5.4E+15 0. -600.
HNO + OH = H2O + NO 3.6E+13 0. 0.
END
CO2 AR END
AREA H2
&prob cx0=2000.0, print=3.048,6.096,7.620,12.19,
htran=.true., qmread=.true., ht1=5.863, ht0=-42.88, &end
&start p=0.956, t=1559.0, mach=5.0,
eratio=1.0, scc=0.0, sch=2.0, scox=0.0, &end
END
&solver &end
FINIS

```


TABLE 13.5.—VARIATION OF COMPUTATIONAL WORK AND SOLUTION WITH ATOLSP FOR H₂-AIR EXAMPLE PROBLEM

[EMAX = 10⁻⁵, x = 6.096 cm.]

Variable	ATOLSP							
	10 ⁻¹⁴	10 ⁻¹³	10 ⁻¹²	10 ⁻¹¹	10 ⁻¹⁰	10 ⁻⁹	10 ⁻⁸	10 ⁻⁷
Number of steps	178	172	150	158	126	110	113	6
Number of derivative evaluations	239	232	220	219	182	159	164	9
Number of Jacobian matrix evaluations	26	25	18	21	19	18	20	3
Execution time, s	0.67	0.64	0.58	0.60	0.50	0.44	0.46	0.03
Reaction time, s	1.3567×10 ⁻⁵	1.3567×10 ⁻⁵	1.3567×10 ⁻⁵	1.3567×10 ⁻⁵	1.3567×10 ⁻⁵	1.3568×10 ⁻⁵	1.3573×10 ⁻⁵	1.3518×10 ⁻⁵
Pressure, atm	1.327	1.327	1.327	1.327	1.328	1.330	1.341	0.9550
Velocity, m/s	4456.3	4456.3	4456.3	4456.3	4456.2	4455.9	4454.4	4509.6
Temperature, K	2216.4	2216.4	2216.4	2216.5	2217.7	2221.5	2241.8	1557.4
Density, g/cm ³	1.5886×10 ⁻⁴	1.5886×10 ⁻⁴	1.5886×10 ⁻⁴	1.5886×10 ⁻⁴	1.5886×10 ⁻⁴	1.5887×10 ⁻⁴	1.5893×10 ⁻⁴	1.5698×10 ⁻⁴
Molar mass	21.767	21.767	21.767	21.767	21.769	21.775	21.803	21.007
Species mole fractions								
O	0.02752	0.02752	0.02752	0.02752	0.02751	0.02745	0.02715	-1.487×10 ⁻⁷
H ₂ O	0.1819	0.1819	0.1819	0.1819	0.1819	0.1821	0.1830	-2.759×10 ⁻⁵
OH	0.03279	0.03279	0.03279	0.03279	0.03282	0.03291	0.03337	-1.474×10 ⁻⁷
H	0.08201	0.08201	0.08201	0.08200	0.08191	0.08162	0.08005	-1.562×10 ⁻⁶
O ₂	0.03186	0.03186	0.03186	0.03186	0.03184	0.03178	0.03147	0.1476
H ₂	0.06666	0.06666	0.06666	0.06666	0.06667	0.06668	0.06677	0.2953
HO ₂	1.487×10 ⁻⁵	1.487×10 ⁻⁵	1.487×10 ⁻⁵	1.487×10 ⁻⁵	1.486×10 ⁻⁵	1.483×10 ⁻⁵	1.466×10 ⁻⁵	1.154×10 ⁻⁵
NO	2.170×10 ⁻⁶	2.170×10 ⁻⁶	2.170×10 ⁻⁶	2.172×10 ⁻⁶	2.198×10 ⁻⁶	2.283×10 ⁻⁶	2.798×10 ⁻⁶	-5.026×10 ⁻¹⁴

13. Examples of Code Usage

error in a species mole fraction is only about 2 ppm. These results demonstrate the importance of being conservative in selecting ATOLSP. Table 13.5 shows that the optimal ATOLSP is 10^{-12} ; note that it required less computational work than 10^{-11} despite being more accurate. The value of 10^{-12} may be peculiar to this case, and so in general a trial-and-error optimization procedure for ATOLSP is necessary.

13.2 Sensitivity Test Case: Constant-Volume, Adiabatic, Static Benzene-Oxygen-Argon Combustion Reaction

The last example problem concerns kinetics and sensitivity analysis calculations in a constant-volume, adiabatic, static system. The problem describes the ignition and subsequent combustion of a near-stoichiometric ($\phi = 1.007$) benzene-oxygen-argon mixture (with 85.728 percent argon in the mixture). The initial temperature and pressure are 1405 K and 2.3868 atm, respectively. The reaction mechanism contains 40 species and 120 reactions, and so reaction mechanism legality testing will be required. Thermodynamic data for all species are to be taken from the standard thermodynamic data file. For both input and output cgs units are to be used. We will specify the initial mixture composition by means of species mole fractions.

Sensitivity coefficients are required for the 14 dependent variables $\sigma_{C_6H_6}$, σ_{OH} , $\sigma_{C_6H_5}$, σ_H , σ_{H_2} , σ_O , σ_{H_2O} , σ_{CO} , $\sigma_{C_2H_2}$, $\sigma_{C_6H_5OH}$, $\sigma_{C_5H_6}$, σ_{CO_2} , T , and p with respect to all three rate coefficient parameters of the seven reactions listed in table 13.6. Here σ_i is the mole number of species i . Sensitivity coefficients with respect to the initial values of the six variables $\sigma_{C_6H_6}$, σ_{OH} , σ_{H_2O} , σ_{Ar} , T , and p must also be computed. The solution is required at the five output stations 10^{-6} , 10^{-5} , 6×10^{-5} , 2.8×10^{-4} , and 3×10^{-4} s (initial time = 0). The standard format is to be used for printing the sensi-

tivity coefficients with respect to the rate coefficient parameters. Finally any normalized sensitivity coefficient with magnitude less than 10^{-7} is to be set equal to zero.

13.2.1 Description of Problem Data File

The required problem data file for this case is listed in table 13.7 and described here.

13.2.1.1 Thermodynamic data, title line, and reaction mechanism data.—The first line of the problem data file contains the word TAPE in columns 1 to 4 because all thermodynamic data are to be taken from the standard thermodynamic data file. The second line contains the case title. It is followed by the chemical mechanism, which was taken from reference 40. The blank line after the last reaction denotes the end of the reaction list. It is followed by a line with the name (AR) of the inert species argon in columns 1 and 2. The blank species field in columns 11 to 18 designates the end of the inert species list.

13.2.1.2 Problem setup data.—The next line contains the word TIME starting in column 1. This is the integration variable for a constant-volume (or any) static reaction problem. The remainder of the line is blank because there is no assigned variable; (the default) cgs units are used for both input and output; and the simplified-composition input option is not used.

Namelist PROB, which is placed next, contains the following variables: The array PRINT lists the print stations (in seconds) at which output is required. The variable RXNTST is set equal to TRUE because the legality of the reaction mechanism is to be tested. The variables RHOCON and SENCAL are set equal to TRUE because this is a constant-volume (density) problem and sensitivity analysis computations are required. Finally TINY is set equal to 1.0×10^{-7} so that any normalized sensitivity coefficient with magnitude less than this value will be set equal to zero.

13.2.1.3 Initial conditions and integration controls.—In namelist START, which follows PROB, we give initial values for only the temperature T and pressure P because the test case involves a constant-density calculation. Because the initial reaction time is zero, TIME is not set. Following this namelist the species names and their initial mole fractions (because MOLEF is not set equal to FALSE in START) are given. The blank line indicates the end of the list of initial mole fractions. Next, namelist SOLVER gives EMAX and ATOLSP.

13.2.1.4 Sensitivity analysis data.—All three data types that may be given for sensitivity analysis computations are listed. The first keyword designating data type is SENSVAR, which indicates that the dependent variables whose sensitivity coefficients are required will be given on the following line (or lines). Fourteen variables are listed: eight on the first line and six on the second. They include the names of the 12 species, temperature, and pressure. The dependent variable list is concluded by the word END in the variable field fol-

TABLE 13.6.—REACTIONS FOR WHICH SENSITIVITY ANALYSIS IS REQUIRED FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM

Reaction number, j	Reaction
1	$C_6H_6 + O_2 \rightleftharpoons C_6H_5O + OH$
6	$C_6H_6 + OH \rightleftharpoons C_6H_5 + H_2O$
8	$C_6H_5O \rightleftharpoons C_6H_5 + CO$
12	$C_6H_5OH \rightleftharpoons C_6H_5O + H$
17	$C_5H_6 + O_2 \rightleftharpoons C_5H_5O + OH$
18	$C_6H_5OH + OH \rightleftharpoons C_6H_5O + H_2O$
104	$H + O_2 \rightleftharpoons OH + O$

TABLE 13.7.—PROBLEM DATA FILE FOR BENZENE-OXYGEN-ARGON SENSITIVITY EXAMPLE PROBLEM

TAPES									
BENZENE-O ₂ -ARGON SHOCK IGNITION SENSITIVITY TEST CASE: CHAPTER 13, CASE 3									
C6H6	+	O2	=	C6H5O	+	OH	4.0E+13	0.	34000.
C6H6	+	C6H5	=	C12H10	+	H	4.0E+11	0.	4000.
		C6H6	=	C6H5	+	H	1.0E+16	0.	108000.
C6H6	+	H	=	C6H5	+	H2	2.5E+14	0.	16000.
C6H6	+	O	=	C6H5O	+	H	2.783E+13	0.	4910.
C6H6	+	OH	=	C6H5	+	H2O	2.132E+13	0.	4580.
M	+	C4H3	=	C4H2	+	H	1.0E+16	0.0	60000.
		C6H5O	=	C5H5	+	CO	2.51E+11	0.	43900.
C6H5	+	O2	=	C6H5O	+	O	2.1E+12	0.	7470.
C6H5	+	HO2	=	C6H5O	+	OH	2.0E+13	0.	1000.
		C6H5	=	C4H3	+	C2H2	4.50E+13	0.	72530.
		C6H5OH	=	C6H5O	+	H	2.00E+16	0.	88000.
C6H5OH	+	H	=	C6H6	+	OH	2.20E+13	0.	7910.
C6H5OH	+	H	=	C6H5O	+	H2	1.15E+14	0.	12405.
C5H5	+	C6H5OH	=	C6H5O	+	C5H6	2.67E+14	0.	25227.
		C5H6	=	C5H5	+	H	8.13E+24	-2.981	78682.
C5H6	+	O2	=	C5H5O	+	OH	1.0E+13	0.	20712.
C6H5OH	+	OH	=	C6H5O	+	H2O	3.0E+13	0.	0.
C6H5OH	+	HO2	=	C6H5O	+	H2O2	3.0E+13	0.	1500.
		C5H5O	=	C4H5	+	CO	3.0E+16	0.	15000.
C5H5	+	O	=	C5H5O	+		1.0E+13	0.	0.
C5H5	+	OH	=	C5H4OH	+	H	1.0E+13	0.	0.
		C5H4OH	=	C4H4	+	HCO	1.0E+15	0.	22000.
C5H5	+	HO2	=	C5H5O	+	OH	2.0E+13	0.	0.
		2.0C6H5	=	C12H10	+		3.1E+12	0.	0.
		C4H5	=	C2H3	+	C2H2	1.4E+13	0.	32900.
C4H2	+	O	=	C2HO	+	C2H	1.0E+13	0.	0.
C4H2	+	OH	=	HCO	+	C3H2	3.0E+13	0.	0.
C4H2	+	O	=	CO	+	C3H2	1.2E+12	0.	0.
M	+	C2H4	=	C2H2	+	H2	9.33E+16	0.	77200.
C2H4	+	OH	=	C2H3	+	H2O	4.786E+12	0.	1230.
C2H4	+	O	=	CH3	+	HCO	3.311E+12	0.	1130.
C2H4	+	O	=	CH2O	+	CH2	2.512E+13	0.	5000.
C2H4	+	OH	=	CH3	+	CH2O	1.995E+12	0.	960.
M	+	C2H3	=	C2H2	+	H	3.0E+15	0.	32000.
C2H3	+	O2	=	CH2O	+	HCO	3.98E+12	0.	-250.
C2H3	+	H	=	C2H2	+	H2	6.0E+12	0.	0.
C2H3	+	OH	=	C2H2	+	H2O	5.012E+12	0.	0.
C2H3	+	CH2	=	C2H2	+	CH3	3.020E+13	0.	0.
C2H3	+	C2H	=	2.0C2H2	+		3.020E+13	0.	0.
C2H3	+	O	=	C2H2O	+	H	3.3E+13	0.	0.
CH2	+	CH2	=	C2H2	+	H2	4.0E+13	0.	0.
CH2	+	CH2	=	C2H3	+	H	5.012E+12	0.	0.
CH2	+	OH	=	CH	+	H2O	2.51E+11	.67	25700.
CH2	+	O	=	CH	+	OH	2.0E+11	.68	25000.
CH2	+	O2	=	CO2	+	2.0H	1.59E+12	0.	1000.
M	+	C2H2	=	C2H	+	H	4.169E+16	0.	107000.
C2H2	+	C2H2	=	C4H3	+	H	2.0E+12	0.	45900.
C2H2	+	O	=	CH2	+	CO	1.6E+14	0.0	9890.
C2H2	+	O	=	C2HO	+	H	4.0E+14	0.0	10660.
C2H2	+	OH	=	C2H	+	H2O	6.310E+12	0.	7000.
C2H2	+	OH	=	C2H2O	+	H	3.2E+11	0.	200.
C2H2	+	C2H	=	C4H2	+	H	3.0E+13	0.	0.
C2H2	+	CH2	=	C3H3	+	H	1.2E+13	0.	6600.
M	+	C3H4	=	C3H3	+	H	2.0E+17	0.	65000.
C2H2O	+	OH	=	CH2O	+	HCO	2.8E+13	0.	0.
C2H2O	+	OH	=	C2HO	+	H2O	7.5E+12	0.	3000.
C2H2O	+	H	=	CH3	+	CO	1.13E+13	0.	3428.
C2H2O	+	H	=	C2HO	+	H2	7.5E+13	0.	8000.
C2H2O	+	O	=	C2HO	+	OH	5.0E+13	0.	8000.
C2H2O	+	O	=	CH2O	+	CO	2.0E+13	0.	0.
M	+	C2H2O	=	CH2	+	CO	2.0E+16	0.	60000.
C2HO	+	O2	=	2.0CO	+	OH	1.46E+12	0.	2500.
C2HO	+	O	=	2.0CO	+	H	1.202E+12	0.	0.
C2HO	+	OH	=	2.0HCO	+		1.0E+13	0.	0.
C2HO	+	H	=	CH2	+	CO	5.0E+13	0.	0.
C2HO	+	CH2	=	C2H3	+	CO	3.0E+13	0.	0.
C2HO	+	CH2	=	CH2O	+	C2H	1.0E+13	0.	2000.
		2.0C2HO	=	C2H2	+	2.0CO	1.0E+13	0.	0.
C2H	+	OH	=	C2HO	+	H	2.0E+13	0.	0.
C2H	+	O2	=	C2HO	+	O	5.00E+13	0.	1500.
C2H	+	O	=	CO	+	CH	5.012E+13	0.	0.
M	+	CH4	=	CH3	+	H	2.0E+17	0.	88000.
CH4	+	O2	=	CH3	+	HO2	7.943E+13	0.	56000.
CH4	+	H	=	CH3	+	H2	1.26E+14	0.	11900.

TABLE 13.7.—Concluded.

OH	+	CH4	=	CH3	+	H2O	2.5E+13	0.	5010.
O	+	CH4	=	CH3	+	OH	1.9E+14	0.	11720.
CH3	+	O2	=	CH3O	+	O	4.786E+13	0.	29000.
CH3	+	OH	=	CH3O	+	H	6.3E+12	0.	0.
M	+	CH3O	=	CH2O	+	H	5.0E+13	0.	21000.
CH3O	+	O2	=	CH2O	+	HO2	1.0E+12	0.	6000.
CH3O	+	H	=	CH2O	+	H2	2.0E+13	0.	0.
CH3	+	CH3	=	C2H4	+	H2	1.0E+16	0.	32000.
CH3	+	O	=	CH2O	+	H	1.288E+14	0.	2000.
CH3	+	CH2O	=	CH4	+	HCO	1.0E+10	0.5	6000.
CH3	+	HCO	=	CH4	+	CO	3.020E+11	.5	0.
CH3	+	HO2	=	CH3O	+	OH	2.00E+13	0.	0.
M	+	CH2O	=	HCO	+	H	5.0E+16	0.	81000.
CH2O	+	OH	=	HCO	+	H2O	3.0E+13	0.	1200.
CH2O	+	H	=	HCO	+	H2	2.5E+13	0.	3990.
CH2O	+	O	=	HCO	+	OH	3.5E+13	0.	3510.
HCO	+	HO2	=	CH2O	+	O2	1.0E+14	0.	3000.
M	+	HCO	=	H	+	CO	2.94E+14	0.	15569.
HCO	+	O2	=	CO	+	HO2	3.311E+12	0.	7000.
HCO	+	OH	=	CO	+	H2O	1.0E+14	0.	0.
HCO	+	H	=	CO	+	H2	1.995E+14	0.	0.
HCO	+	O	=	CO	+	OH	1.0E+14	0.	0.
CH	+	O2	=	HCO	+	O	1.0E+13	0.	0.
CO	+	O	=	CO2	+	M	5.9E+15	0.	4100.
CO	+	O2	=	CO2	+	O	2.5E+12	0.	47690.
CO	+	OH	=	CO2	+	H	4.17E+11	0.	1000.
CO	+	HO2	=	CO2	+	OH	5.75E+13	0.	22930.
O	+	H2O	=	OH	+	OH	6.8E+13	0.	18365.
H	+	O2	=	OH	+	O	1.89E+14	0.	16400.
O	+	H2	=	OH	+	H	4.20E+14	0.	13750.
H	+	HO2	=	H2	+	O2	7.28E+13	0.	2126.
O	+	HO2	=	OH	+	O2	5.0E+13	0.	1000.
HO2	+	OH	=	H2O	+	O2	8.0E+12	0.	0.
H	+	HO2	=	2.OOH			1.34E+14	0.	1070.
H2	+	HO2	=	H2O2	+	H	7.91E+13	0.	25000.
OH	+	H2O2	=	H2O	+	HO2	6.1E+12	0.	1430.
HO2	+	HO2	=	H2O2	+	O2	1.8E+12	0.	0.
H	+	H2O2	=	OH	+	H2O	7.8E+11	0.	0.
M	+	H2O2	=	OH	+	OH	1.44E+17	0.	45510.
THIRDBODY									
H2		2.30	O2		.78	H2O	6.0	H2O2	6.6
END									
H2	+	OH	=	H2O	+	H	4.74E+13	0.	6098.
H	+	O2	=	HO2	+	M	1.46E+15	0.	-1000.
THIRDBODY									
O2		1.30	CO2		7.0	H2O	21.3	H2	3.0
C6H6		20.0	CH4		5.0	END			
M	+	H2O	=	H	+	OH	1.30E+15	0.	105140.
THIRDBODY									
H2		4.00	O2		1.5	H2O	20.0	C6H6	20.0
CO2		4.00	END						
H	+	O	=	OH	+	M	7.1E+18	-1.	0.
M	+	H2	=	H	+	H	2.2E+14	0.	96000.
THIRDBODY									
H2		4.10	O2		2.0	H2O	15.0	END	
M	+	O2	=	O	+	O	1.80E+18	-1.	118020.
AR									
TIME									
&prob	rhocon=.true.,	sencal=.true.,	tiny=1.0E-7,	rxntst=.true.,					
print	=1.0E-6,1.0E-5,6.0E-5,2.8E-4,3.0E-4,	&end							
&start	t=1405.0,	p=2.3868,	&end						
C6H6	0.01690								
O2	0.12582								
AR	0.85728								
&solver	emax=1.0E-6,	atolsp=1.0E-15,	&end						
SENSVAR									
C6H6	OH	C6H5	H	H2	O	H2O	CO		
C2H2	C6H5OH	C5H6	CO2	TEMP	PRESSURE	END			
INIT									
C6H6	OH	H2O	AR	TEMP	DENSITY	END			
REAC									
&senrxn	sensaj=.true.,	sensnj=.true.,	sensej=.true.,						
rxnum	=1.0,6.0,8.0,12.0,17.0,18.0,104.0,	&end							
FINIS									

lowing PRESSURE. (Note that if an exact multiple of eight variables had been listed, an additional line that either is blank or contains the word END starting in column 1 would have been required.)

The next keyword is INIT, which signifies that sensitivities with respect to the initial values of certain variables must be computed. These variables are given on the next line, using the same format as that used for the dependent variables.

The last keyword is REAC, which means that sensitivity coefficients with respect to rate coefficient parameters are required. The keyword is followed by namelist SENRXN, which gives the necessary information about the rate coefficient parameters and reactions to be considered. The variables SENSAJ, SENSNJ, and SENSEJ are set equal to TRUE because sensitivity coefficients with respect to the preexponential factor A_j , the temperature exponent n_j , and the activation energy E_j are needed. The array RXNUM contains the reaction numbers for which sensitivity coefficients must be produced. Because these coefficients do not have to be computed for all reactions, ALLRXN is not employed. Finally neither OUTPUT nor ORDER is set because (1) the standard format is to be used for printing the normalized sensitivity coefficients with respect to the rate coefficient parameters and (2) reaction numbers do not have to be listed in order of decreasing importance.

13.2.1.5 FINIS line.—The last line contains the word FINIS starting in column 1 to signal the end of the problem data file.

13.2.2 Computed Results and Effects of Local Error Tolerances

Selection of the appropriate local error tolerance parameters is especially important for sensitivity analysis computations because the accuracy of the sensitivity coefficients is controlled by the accuracy of the kinetics solution. In order to understand how the computed results vary with the local error tolerances, EMAX was progressively decreased from 10^{-2} to 10^{-10} by a factor of 10, while keeping ATOLSP = 10^{-9} EMAX. Table 13.8 shows the effects of EMAX on several species molar concentrations, temperature, pressure, and the computational work for the combined kinetics and sensitivity analysis computations. The kinetics solution was quite inaccurate at EMAX = 10^{-2} and did not stabilize until EMAX was decreased to 10^{-6} . This behavior is different from that observed previously for the hydrogen-air case (see table 13.4) and illustrates the need to be conservative in selecting EMAX. However, table 13.8 shows a strong dependence of the execution time on EMAX, which must therefore be optimized to minimize computational cost.

The effects of ATOLSP on the accuracy of the solutions were studied by progressively increasing ATOLSP from

10^{-15} to 10^{-8} with EMAX = 10^{-6} . The kinetics results are given in table 13.9. Again the execution time includes that required for sensitivity analysis. Table 13.9 shows that ATOLSP has to be increased to 10^{-8} before any appreciable differences from the accurate values are noted in the results. Note that even ATOLSP = 10^{-9} is sufficiently accurate. It was also significantly less expensive than ATOLSP = 10^{-15} . Table 13.9 reinforces the desirability of optimizing ATOLSP.

The effects of EMAX on the normalized sensitivity coefficients $\{\langle S_{ij} \rangle\}$ with respect to the initial condition values are presented in tables 13.10 to 13.15. In these tables Y_i is the i th dependent variable and $\sigma_{j,0}$ is the initial value of σ_j . The first four tables give the $\{\langle S_{ij} \rangle\}$ with respect to the initial mole numbers of benzene, hydroxyl radical, water, and argon. Tables 13.14 and 13.15 list the normalized sensitivity coefficients with respect to the initial temperature and density, respectively. The six tables show that the $\{\langle S_{ij} \rangle\}$ have a much stronger dependence than the kinetics solution on EMAX. They change substantially as EMAX is reduced from 10^{-2} to 10^{-5} and stabilize to three significant figures at EMAX = 10^{-6} . The latter fact is not surprising inasmuch as the kinetics solution required EMAX = 10^{-6} to become essentially tolerance independent.

Tables 13.16 to 13.24 show the variations of several rate coefficient parameter sensitivity coefficients with EMAX. The first seven tables give $\{\langle S_{ij} \rangle\}$ with respect to the $\{A_j\}$. Tables 13.23 and 13.24 present the effects of varying EMAX on the normalized sensitivity coefficients with respect to n_8 and E_8 , respectively. All $\{\langle S_{ij} \rangle\}$ stabilized to three significant figures only when EMAX was reduced to 10^{-6} , similar to the initial condition sensitivities. EMAX should therefore be approximately 10^{-6} for accurate sensitivity analysis computations for this typical ignition and combustion problem.

The variations of the $\{\langle S_{ij} \rangle\}$ with ATOLSP are shown in tables 13.25 to 13.30 for several parameters. The first three tables give normalized sensitivity coefficients with respect to the initial mole numbers of benzene and water and the initial temperature. Tables 13.28 to 13.30 present $\{\langle S_{ij} \rangle\}$ with respect to A_1 , n_8 , and E_8 , respectively. These tables show that varying ATOLSP in the range 10^{-15} to 10^{-10} has a small effect on the solutions. In all cases the change is a fraction of 1 percent. Although somewhat less accurate, the results obtained with ATOLSP = 10^{-9} were adequate. The run with ATOLSP = 10^{-8} , especially the sensitivities of $\sigma_{C_6H_5}$, did, however, incur errors greater than 10 percent. Again the sensitivity coefficients display similar dependence on the local error tolerance as the kinetics solution. Thus the need to generate reliable kinetics results is indicated. It is therefore recommended that first optimal EMAX and ATOLSP values be established for the kinetics solution and then sensitivities be generated with somewhat more accurate tolerances.

TABLE 13.8.—VARIATION OF COMPUTATIONAL WORK AND KINETICS SOLUTION WITH EMAX
 FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
 [ATOLSP = 10^{-9} EMAX, $t = 300 \mu\text{s}$.]

Variable	EMAX									
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}	
Number of steps	74	97	187	246	391	649	954	1379	2067	
Number of derivative evaluations	131	163	285	367	560	879	1299	1950	2811	
Number of Jacobian matrix evaluations	21	22	34	38	47	91	127	206	284	
CPU time, s	6.8	8.9	16	22	34	57	83	121	181	
Temperature, K	1956.9	1914.4	1910.7	1909.7	1909.6	1909.6	1909.6	1909.6	1909.6	
Pressure, atm	3.414	3.332	3.325	3.324	3.323	3.323	3.323	3.323	3.323	
	Species molar concentrations, moles/cm ³									
C ₆ H ₆	4.003×10^{-8}	4.929×10^{-8}	5.012×10^{-8}	5.034×10^{-8}	5.036×10^{-8}	5.036×10^{-8}	5.036×10^{-8}	5.036×10^{-8}	5.036×10^{-8}	
OH	2.722×10^{-8}	1.874×10^{-8}	1.813×10^{-8}	1.797×10^{-8}	1.795×10^{-8}	1.795×10^{-8}	1.795×10^{-8}	1.795×10^{-8}	1.795×10^{-8}	
C ₆ H ₅	4.711×10^{-9}	4.700×10^{-9}	4.685×10^{-9}	4.681×10^{-9}	4.680×10^{-9}	4.680×10^{-9}	4.680×10^{-9}	4.680×10^{-9}	4.680×10^{-9}	
H	1.108×10^{-8}	7.844×10^{-9}	7.606×10^{-9}	7.544×10^{-9}	7.539×10^{-9}	7.538×10^{-9}	7.538×10^{-9}	7.538×10^{-9}	7.538×10^{-9}	
H ₂	1.972×10^{-8}	1.819×10^{-8}	1.807×10^{-8}	1.804×10^{-8}	1.804×10^{-8}	1.803×10^{-8}	1.803×10^{-8}	1.803×10^{-8}	1.803×10^{-8}	
O	1.974×10^{-8}	1.336×10^{-8}	1.290×10^{-8}	1.278×10^{-8}	1.277×10^{-8}	1.276×10^{-8}	1.276×10^{-8}	1.276×10^{-8}	1.276×10^{-8}	
H ₂ O	4.142×10^{-7}	3.832×10^{-7}	3.805×10^{-7}	3.797×10^{-7}	3.797×10^{-7}	3.797×10^{-7}	3.797×10^{-7}	3.797×10^{-7}	3.797×10^{-7}	
CO	8.256×10^{-7}	7.650×10^{-7}	7.599×10^{-7}	7.585×10^{-7}	7.584×10^{-7}	7.584×10^{-7}	7.584×10^{-7}	7.584×10^{-7}	7.584×10^{-7}	
C ₂ H ₂	3.180×10^{-8}	3.655×10^{-8}	3.698×10^{-8}	3.710×10^{-8}	3.711×10^{-8}	3.711×10^{-8}	3.711×10^{-8}	3.711×10^{-8}	3.711×10^{-8}	
C ₆ H ₅ OH	3.201×10^{-8}	3.758×10^{-8}	3.804×10^{-8}	3.816×10^{-8}	3.817×10^{-8}	3.817×10^{-8}	3.817×10^{-8}	3.817×10^{-8}	3.817×10^{-8}	
C ₅ H ₆	1.421×10^{-8}	1.617×10^{-8}	1.633×10^{-8}	1.637×10^{-8}	1.638×10^{-8}	1.638×10^{-8}	1.638×10^{-8}	1.638×10^{-8}	1.638×10^{-8}	
CO ₂	7.311×10^{-8}	6.052×10^{-8}	5.948×10^{-8}	5.920×10^{-8}	5.918×10^{-8}	5.918×10^{-8}	5.918×10^{-8}	5.918×10^{-8}	5.918×10^{-8}	

TABLE 13.9.—VARIATION OF COMPUTATIONAL WORK AND KINETICS SOLUTION WITH ATOLSP
FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[EMAX = 10^{-6} , $t = 300 \mu\text{s}$.]

Variable	ATOLSP									
	10^{-15}	10^{-14}	10^{-13}	10^{-12}	10^{-11}	10^{-10}	10^{-9}	10^{-8}		
Number of steps	391	354	348	236	195	137	99	85		
Number of derivative evaluations	560	482	502	338	299	207	159	134		
Number of Jacobian matrix evaluations	47	52	49	31	30	20	20	17		
CPU time, s	34	31	30	21	17	12	8.9	7.5		
Temperature, K	1909.6	1909.6	1909.6	1909.7	1909.7	1910.0	1910.5	1914.4		
Pressure, atm	3.323	3.323	3.323	3.323	3.324	3.324	3.325	3.333		
Species molar concentrations, moles/cm ³										
C ₆ H ₆	5.036×10 ⁻⁸	5.036×10 ⁻⁸	5.036×10 ⁻⁸	5.034×10 ⁻⁸	5.034×10 ⁻⁸	5.027×10 ⁻⁸	5.016×10 ⁻⁸	4.925×10 ⁻⁸		
OH	1.795×10 ⁻⁸	1.796×10 ⁻⁸	1.796×10 ⁻⁸	1.796×10 ⁻⁸	1.797×10 ⁻⁸	1.802×10 ⁻⁸	1.810×10 ⁻⁸	1.875×10 ⁻⁸		
C ₆ H ₅	4.680×10 ⁻⁹	4.680×10 ⁻⁹	4.680×10 ⁻⁹	4.680×10 ⁻⁹	4.681×10 ⁻⁹	4.682×10 ⁻⁹	4.684×10 ⁻⁹	4.699×10 ⁻⁹		
H	7.539×10 ⁻⁹	7.539×10 ⁻⁹	7.539×10 ⁻⁹	7.542×10 ⁻⁹	7.544×10 ⁻⁹	7.564×10 ⁻⁹	7.594×10 ⁻⁹	7.846×10 ⁻⁹		
H ₂	1.804×10 ⁻⁸	1.804×10 ⁻⁸	1.804×10 ⁻⁸	1.804×10 ⁻⁸	1.804×10 ⁻⁸	1.805×10 ⁻⁸	1.807×10 ⁻⁸	1.820×10 ⁻⁸		
O	1.277×10 ⁻⁸	1.277×10 ⁻⁸	1.277×10 ⁻⁸	1.277×10 ⁻⁸	1.278×10 ⁻⁸	1.281×10 ⁻⁸	1.287×10 ⁻⁸	1.335×10 ⁻⁸		
H ₂ O	3.797×10 ⁻⁷	3.797×10 ⁻⁷	3.797×10 ⁻⁷	3.797×10 ⁻⁷	3.797×10 ⁻⁷	3.800×10 ⁻⁷	3.803×10 ⁻⁷	3.832×10 ⁻⁷		
CO	7.584×10 ⁻⁷	7.584×10 ⁻⁷	7.584×10 ⁻⁷	7.585×10 ⁻⁷	7.585×10 ⁻⁷	7.590×10 ⁻⁷	7.596×10 ⁻⁷	7.652×10 ⁻⁷		
C ₂ H ₂	3.711×10 ⁻⁸	3.711×10 ⁻⁸	3.711×10 ⁻⁸	3.710×10 ⁻⁸	3.710×10 ⁻⁸	3.706×10 ⁻⁸	3.701×10 ⁻⁸	3.656×10 ⁻⁸		
C ₆ H ₅ OH	3.817×10 ⁻⁸	3.817×10 ⁻⁸	3.817×10 ⁻⁸	3.816×10 ⁻⁸	3.816×10 ⁻⁸	3.812×10 ⁻⁸	3.806×10 ⁻⁸	3.756×10 ⁻⁸		
C ₅ H ₆	1.638×10 ⁻⁸	1.638×10 ⁻⁸	1.638×10 ⁻⁸	1.637×10 ⁻⁸	1.637×10 ⁻⁸	1.636×10 ⁻⁸	1.634×10 ⁻⁸	1.618×10 ⁻⁸		
CO ₂	5.918×10 ⁻⁸	5.918×10 ⁻⁸	5.918×10 ⁻⁸	5.919×10 ⁻⁸	5.920×10 ⁻⁸	5.929×10 ⁻⁸	5.942×10 ⁻⁸	6.048×10 ⁻⁸		

TABLE 13.10.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y/\partial \sigma_{C_6H_6,0})$ WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y/\partial \sigma_{C_6H_6,0})$								
T	2.31	1.76	1.70	1.69	1.69	1.69	1.69	1.69	1.69
p	2.60	1.97	1.90	1.88	1.88	1.88	1.88	1.88	1.88
$\sigma_{C_6H_6}$	-20.9	-13.4	-12.8	-12.6	-12.6	-12.6	-12.6	-12.6	-12.6
σ_{OH}	37.3	29.5	28.5	28.3	28.3	28.3	28.3	28.3	28.3
$\sigma_{C_6H_5}$	-0.916	3.79	4.01	4.07	4.07	4.07	4.07	4.07	4.07
σ_H	35.8	28.4	27.5	27.3	27.3	27.3	27.3	27.3	27.3
σ_{H_2}	10.3	8.00	7.80	7.75	7.74	7.74	7.74	7.74	7.74
σ_O	39.0	30.2	29.1	28.9	28.9	28.9	28.9	28.9	28.9
σ_{H_2O}	8.22	6.81	6.63	6.59	6.59	6.59	6.59	6.59	6.59
σ_{CO}	8.30	6.66	6.48	6.44	6.43	6.43	6.43	6.43	6.43
$\sigma_{C_2H_2}$	-13.3	-8.46	-8.00	-7.88	-7.87	-7.87	-7.87	-7.87	-7.87
$\sigma_{C_6H_5OH}$	-17.3	-10.1	-9.55	-9.40	-9.39	-9.39	-9.39	-9.39	-9.39
$\sigma_{C_5H_6}$	-14.2	-6.97	-6.40	-6.26	-6.25	-6.25	-6.25	-6.25	-6.25
σ_{CO_2}	19.3	14.6	14.1	14.0	14.0	14.0	14.0	14.0	14.0

TABLE 13.11.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y/\partial \sigma_{OH,0})$ WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y/\partial \sigma_{OH,0})$								
T	0.322	0.248	0.240	0.238	0.238	0.238	0.238	0.238	0.238
p	0.358	0.274	0.264	0.262	0.262	0.262	0.262	0.262	0.262
$\sigma_{C_6H_6}$	-3.28	-2.22	-2.13	-2.11	-2.11	-2.11	-2.11	-2.11	-2.11
σ_{OH}	5.31	4.28	4.15	4.12	4.12	4.12	4.12	4.12	4.12
$\sigma_{C_6H_5}$	-0.371	0.321	0.355	0.364	0.364	0.364	0.364	0.364	0.364
σ_H	4.96	3.98	3.86	3.83	3.83	3.83	3.83	3.83	3.83
σ_{H_2}	1.22	0.918	0.889	0.883	0.882	0.882	0.882	0.882	0.882
σ_O	5.61	4.43	4.28	4.24	4.24	4.24	4.24	4.24	4.24
σ_{H_2O}	1.10	0.918	0.893	0.887	0.887	0.887	0.887	0.887	0.887
σ_{CO}	1.10	0.886	0.860	0.854	0.854	0.854	0.854	0.854	0.854
$\sigma_{C_2H_2}$	-2.13	-1.44	-1.37	-1.36	-1.36	-1.36	-1.36	-1.36	-1.36
$\sigma_{C_6H_5OH}$	-2.62	-1.61	-1.53	-1.51	-1.51	-1.51	-1.51	-1.51	-1.51
$\sigma_{C_5H_6}$	-2.21	-1.19	-1.10	-1.08	-1.08	-1.08	-1.08	-1.08	-1.08
σ_{CO_2}	2.72	2.10	2.02	2.01	2.00	2.00	2.00	2.00	2.00

TABLE 13.12.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $\langle \partial \gamma / \partial \sigma_{\text{H}_2\text{O}} \rangle$ WITH EMAX
 FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
 [ATOLSP = 10^{-9} EMAX, $t = 300 \mu\text{s}$]

Variable	EMAX									
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}	
	$\langle \partial \gamma / \partial \sigma_{\text{H}_2\text{O}} \rangle$									
T	-1.04×10^{-3}	-7.86×10^{-4}	-7.59×10^{-4}	-7.52×10^{-4}	-7.52×10^{-4}	-7.52×10^{-4}	-7.52×10^{-4}	-7.52×10^{-4}	-7.52×10^{-4}	-7.52×10^{-4}
p	-1.05×10^{-3}	-7.61×10^{-4}	-7.31×10^{-4}	-7.24×10^{-4}	-7.23×10^{-4}	-7.23×10^{-4}	-7.23×10^{-4}	-7.23×10^{-4}	-7.23×10^{-4}	-7.23×10^{-4}
$\sigma_{\text{C}_6\text{H}_6}$	9.91×10^{-3}	6.50×10^{-3}	6.20×10^{-3}	6.13×10^{-3}	6.12×10^{-3}	6.12×10^{-3}	6.12×10^{-3}	6.12×10^{-3}	6.12×10^{-3}	6.12×10^{-3}
σ_{OH}	-1.62×10^{-2}	-1.27×10^{-2}	-1.23×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}
$\sigma_{\text{C}_6\text{H}_5}$	1.24×10^{-3}	-9.35×10^{-4}	-1.04×10^{-3}	-1.07×10^{-3}	-1.07×10^{-3}	-1.07×10^{-3}	-1.07×10^{-3}	-1.07×10^{-3}	-1.07×10^{-3}	-1.07×10^{-3}
σ_{H}	-1.60×10^{-2}	-1.27×10^{-2}	-1.23×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}	-1.22×10^{-2}
σ_{H_2}	-2.47×10^{-3}	-1.51×10^{-3}	-1.42×10^{-3}	-1.40×10^{-3}	-1.40×10^{-3}	-1.40×10^{-3}	-1.40×10^{-3}	-1.40×10^{-3}	-1.40×10^{-3}	-1.40×10^{-3}
σ_{O}	-1.85×10^{-2}	-1.44×10^{-2}	-1.40×10^{-2}	-1.38×10^{-2}	-1.38×10^{-2}	-1.38×10^{-2}	-1.38×10^{-2}	-1.38×10^{-2}	-1.38×10^{-2}	-1.38×10^{-2}
$\sigma_{\text{H}_2\text{O}}$	1.62×10^{-3}	2.70×10^{-3}	2.81×10^{-3}	2.85×10^{-3}	2.85×10^{-3}	2.85×10^{-3}	2.85×10^{-3}	2.85×10^{-3}	2.85×10^{-3}	2.85×10^{-3}
σ_{CO}	-3.34×10^{-3}	-2.58×10^{-3}	-2.50×10^{-3}	-2.48×10^{-3}	-2.47×10^{-3}	-2.47×10^{-3}	-2.47×10^{-3}	-2.47×10^{-3}	-2.47×10^{-3}	-2.47×10^{-3}
$\sigma_{\text{C}_2\text{H}_2}$	6.86×10^{-3}	4.68×10^{-3}	4.47×10^{-3}	4.42×10^{-3}	4.42×10^{-3}	4.42×10^{-3}	4.42×10^{-3}	4.42×10^{-3}	4.42×10^{-3}	4.42×10^{-3}
$\sigma_{\text{C}_6\text{H}_5\text{OH}}$	7.68×10^{-3}	4.46×10^{-3}	4.21×10^{-3}	4.14×10^{-3}	4.14×10^{-3}	4.14×10^{-3}	4.14×10^{-3}	4.14×10^{-3}	4.14×10^{-3}	4.14×10^{-3}
$\sigma_{\text{C}_3\text{H}_6}$	6.73×10^{-3}	3.49×10^{-3}	3.23×10^{-3}	3.17×10^{-3}	3.17×10^{-3}	3.16×10^{-3}	3.16×10^{-3}	3.16×10^{-3}	3.16×10^{-3}	3.16×10^{-3}
σ_{CO_2}	-8.21×10^{-3}	-6.13×10^{-3}	-5.90×10^{-3}	-5.84×10^{-3}	-5.84×10^{-3}	-5.84×10^{-3}	-5.84×10^{-3}	-5.84×10^{-3}	-5.84×10^{-3}	-5.84×10^{-3}

TABLE 13.13.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y/\partial \sigma_{Ar,0})$ WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM [ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y/\partial \sigma_{Ar,0})$								
T	-3.26	-2.49	-2.40	-2.38	-2.38	-2.38	-2.38	-2.38	-2.38
p	-2.77	-1.89	-1.80	-1.78	-1.77	-1.77	-1.77	-1.77	-1.77
$\sigma_{C_6H_6}$	32.1	21.4	20.4	20.2	20.2	20.2	20.2	20.2	20.2
σ_{OH}	-53.3	-42.4	-41.1	-40.8	-40.8	-40.8	-40.8	-40.8	-40.8
$\sigma_{C_6H_5}$	3.12	-3.72	-4.04	-4.13	-4.13	-4.14	-4.14	-4.14	-4.14
σ_H	-49.9	-39.6	-38.4	-38.1	-38.1	-38.1	-38.1	-38.1	-38.1
σ_{H_2}	-12.3	-9.08	-8.79	-8.72	-8.71	-8.71	-8.71	-8.71	-8.71
σ_O	-56.4	-44.0	-42.5	-42.2	-42.2	-42.2	-42.2	-42.2	-42.2
σ_{H_2O}	-10.7	-8.67	-8.42	-8.36	-8.36	-8.36	-8.36	-8.36	-8.36
σ_{CO}	-10.7	-8.43	-8.18	-8.12	-8.11	-8.11	-8.11	-8.11	-8.11
$\sigma_{C_2H_2}$	21.2	14.2	13.6	13.4	13.4	13.4	13.4	13.4	13.4
$\sigma_{C_6H_5OH}$	26.0	15.8	15.0	14.8	14.7	14.7	14.7	14.7	14.7
$\sigma_{C_5H_6}$	22.1	11.7	10.9	10.7	10.7	10.7	10.7	10.7	10.7
σ_{CO_2}	-26.6	-20.0	-19.3	-19.1	-19.1	-19.1	-19.1	-19.1	-19.1

TABLE 13.14.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y/\partial T_0)$ WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM [ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y/\partial T_0)$								
T	96.6	74.0	71.6	71.0	70.9	70.9	70.9	70.9	70.9
p	107	81.5	78.7	78.0	78.0	78.0	78.0	78.0	78.0
$\sigma_{C_6H_6}$	-981	-660	-633	-626	-625	-625	-625	-625	-625
σ_{OH}	1.59×10^3	1.27×10^3	1.24×10^3	1.23×10^3	1.23×10^3	1.23×10^3	1.23×10^3	1.23×10^3	1.23×10^3
$\sigma_{C_6H_5}$	-110	97.4	107	110	110	110	110	110	110
σ_H	1.49×10^3	1.19×10^3	1.15×10^3	1.14×10^3	1.14×10^3	1.14×10^3	1.14×10^3	1.14×10^3	1.14×10^3
σ_{H_2}	366	273	264	262	262	262	262	262	262
σ_O	1.68×10^3	1.32×10^3	1.28×10^3	1.27×10^3	1.26×10^3	1.26×10^3	1.26×10^3	1.26×10^3	1.26×10^3
σ_{H_2O}	329	271	264	262	262	262	262	262	262
σ_{CO}	330	262	255	253	253	253	253	253	253
$\sigma_{C_2H_2}$	-639	-429	-409	-404	-404	-404	-404	-404	-404
$\sigma_{C_6H_5OH}$	-784	-478	-454	-448	-448	-448	-448	-448	-448
$\sigma_{C_5H_6}$	-662	-352	-328	-322	-321	-321	-321	-321	-321
σ_{CO_2}	814	622	600	595	595	595	595	595	595

TABLE 13.15.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $\langle \partial Y_i / \partial p_0 \rangle$
 WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
 [ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$\langle \partial Y_i / \partial p_0 \rangle$								
T	5.50	4.21	4.07	4.03	4.03	4.03	4.03	4.03	4.03
p	7.12	5.63	5.47	5.44	5.43	5.43	5.43	5.43	5.43
$\sigma_{C_6H_6}$	-56.0	-37.6	-36.1	-35.7	-35.6	-35.6	-35.6	-35.6	-35.6
σ_{OH}	90.0	71.7	69.5	69.0	68.9	68.9	68.9	68.9	68.9
$\sigma_{C_6H_5}$	-6.47	5.19	5.75	5.90	5.91	5.91	5.91	5.91	5.91
σ_H	83.8	66.4	64.4	63.9	63.9	63.9	63.9	63.9	63.9
σ_{H_2}	20.5	15.2	14.7	14.6	14.6	14.6	14.6	14.6	14.6
σ_O	95.1	74.0	71.6	71.0	70.9	70.9	70.9	70.9	70.9
σ_{H_2O}	18.8	15.5	15.1	15.0	15.0	15.0	15.0	15.0	15.0
σ_{CO}	18.8	15.0	14.6	14.4	14.4	14.4	14.4	14.4	14.4
$\sigma_{C_2H_2}$	-36.2	-24.2	-23.0	-22.7	-22.7	-22.7	-22.7	-22.7	-22.7
$\sigma_{C_6H_5OH}$	-44.4	-26.9	-25.6	-25.2	-25.2	-25.2	-25.2	-25.2	-25.2
$\sigma_{C_5H_6}$	-37.9	-20.2	-18.8	-18.5	-18.4	-18.4	-18.4	-18.4	-18.4
σ_{CO_2}	46.4	35.4	34.1	33.8	33.8	33.8	33.8	33.8	33.8

TABLE 13.16.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $\langle \partial Y_i / \partial A_1 \rangle$
 WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
 [ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$\langle \partial Y_i / \partial A_1 \rangle$								
T	2.67	2.04	1.98	1.96	1.96	1.96	1.96	1.96	1.96
p	2.97	2.25	2.18	2.16	2.16	2.16	2.16	2.16	2.16
$\sigma_{C_6H_6}$	-27.4	-18.5	-17.7	-17.5	-17.5	-17.5	-17.5	-17.5	-17.5
σ_{OH}	44.1	35.3	34.2	34.0	33.9	33.9	33.9	33.9	33.9
$\sigma_{C_6H_5}$	-3.22	2.50	2.78	2.85	2.86	2.86	2.86	2.86	2.86
σ_H	41.2	32.8	31.8	31.6	31.6	31.6	31.6	31.6	31.6
σ_{H_2}	10.1	7.51	7.28	7.23	7.22	7.22	7.22	7.22	7.22
σ_O	46.6	36.4	35.3	35.0	34.9	34.9	34.9	34.9	34.9
σ_{H_2O}	9.14	7.55	7.36	7.31	7.30	7.30	7.30	7.30	7.30
σ_{CO}	9.17	7.32	7.11	7.06	7.06	7.06	7.06	7.06	7.06
$\sigma_{C_2H_2}$	-17.6	-11.8	-11.2	-11.1	-11.1	-11.1	-11.1	-11.1	-11.1
$\sigma_{C_6H_5OH}$	-21.7	-13.2	-12.6	-12.4	-12.4	-12.4	-12.4	-12.4	-12.4
$\sigma_{C_5H_6}$	-18.3	-9.70	-9.02	-8.85	-8.84	-8.84	-8.84	-8.84	-8.84
σ_{CO_2}	22.5	17.2	16.7	16.5	16.5	16.5	16.5	16.5	16.5

TABLE 13.17.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y_i/\partial A_g)$
WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y_i/\partial A_g)$								
T	1.00	0.765	0.740	0.733	0.733	0.733	0.733	0.733	0.733
p	1.11	0.843	0.814	0.807	0.807	0.807	0.807	0.807	0.807
$\sigma_{C_6H_6}$	-10.6	-7.18	-6.89	-6.82	-6.81	-6.81	-6.81	-6.81	-6.81
σ_{OH}	16.4	13.1	12.7	12.6	12.6	12.6	12.6	12.6	12.6
$\sigma_{C_6H_5}$	-0.820	1.33	1.44	1.47	1.47	1.47	1.47	1.47	1.47
σ_H	15.5	12.3	11.9	11.8	11.8	11.8	11.8	11.8	11.8
σ_{H_2}	3.82	2.85	2.76	2.74	2.74	2.74	2.74	2.74	2.74
σ_O	17.4	13.6	13.2	13.1	13.1	13.1	13.1	13.1	13.1
σ_{H_2O}	3.44	2.83	2.76	2.74	2.74	2.74	2.74	2.74	2.74
σ_{CO}	3.45	2.74	2.67	2.65	2.64	2.64	2.64	2.64	2.64
$\sigma_{C_2H_2}$	-6.45	-4.28	-4.08	-4.03	-4.03	-4.03	-4.03	-4.03	-4.03
$\sigma_{C_6H_5OH}$	-8.03	-4.81	-4.56	-4.49	-4.49	-4.49	-4.49	-4.49	-4.49
$\sigma_{C_5H_6}$	-6.68	-3.45	-3.20	-3.14	-3.13	-3.13	-3.13	-3.13	-3.13
σ_{CO_2}	8.38	6.37	6.15	6.10	6.09	6.09	6.09	6.09	6.09

TABLE 13.18.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y_i/\partial A_g)$
WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y_i/\partial A_g)$								
T	2.84	2.17	2.09	2.08	2.07	2.07	2.07	2.07	2.07
p	3.15	2.39	2.31	2.29	2.28	2.28	2.28	2.28	2.28
$\sigma_{C_6H_6}$	-28.9	-19.4	-18.5	-18.3	-18.3	-18.3	-18.3	-18.3	-18.3
σ_{OH}	47.2	37.8	36.6	36.3	36.3	36.3	36.3	36.3	36.3
$\sigma_{C_6H_5}$	-3.06	3.06	3.35	3.43	3.44	3.44	3.44	3.44	3.44
σ_H	44.2	35.2	34.2	33.9	33.9	33.9	33.9	33.9	33.9
σ_{H_2}	10.9	8.14	7.89	7.83	7.83	7.83	7.83	7.83	7.83
σ_O	49.9	39.1	37.8	37.5	37.5	37.5	37.5	37.5	37.5
σ_{H_2O}	9.70	7.99	7.77	7.72	7.72	7.72	7.72	7.72	7.72
σ_{CO}	9.75	7.76	7.54	7.48	7.48	7.48	7.48	7.48	7.48
$\sigma_{C_2H_2}$	-18.9	-12.7	-12.1	-11.9	-11.9	-11.9	-11.9	-11.9	-11.9
$\sigma_{C_6H_5OH}$	-23.3	-14.3	-13.6	-13.4	-13.4	-13.4	-13.4	-13.4	-13.4
$\sigma_{C_5H_6}$	-19.0	-9.86	-9.14	-8.96	-8.94	-8.94	-8.94	-8.94	-8.94
σ_{CO_2}	24.1	18.4	17.8	17.6	17.6	17.6	17.6	17.6	17.6

TABLE 13.19.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $\langle \partial Y_i / \partial A_{12} \rangle$ WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$\langle \partial Y_i / \partial A_{12} \rangle$								
T	-1.95	-1.49	-1.44	-1.43	-1.43	-1.43	-1.43	-1.43	-1.43
p	-2.16	-1.64	-1.59	-1.57	-1.57	-1.57	-1.57	-1.57	-1.57
$\sigma_{C_6H_6}$	19.8	13.3	12.8	12.7	12.6	12.6	12.6	12.6	12.6
σ_{OH}	-32.2	-25.8	-25.0	-24.8	-24.8	-24.8	-24.8	-24.8	-24.8
$\sigma_{C_6H_5}$	2.19	-1.99	-2.20	-2.25	-2.25	-2.25	-2.25	-2.25	-2.25
σ_H	-30.1	-24.0	-23.3	-23.1	-23.1	-23.1	-23.1	-23.1	-23.1
σ_{H_2}	-7.41	-5.56	-5.39	-5.35	-5.35	-5.34	-5.34	-5.34	-5.34
σ_O	-34.0	-26.7	-25.8	-25.6	-25.6	-25.6	-25.6	-25.6	-25.6
σ_{H_2O}	-6.66	-5.50	-5.35	-5.32	-5.31	-5.31	-5.31	-5.31	-5.31
σ_{CO}	-6.67	-5.32	-5.18	-5.14	-5.13	-5.13	-5.13	-5.13	-5.13
$\sigma_{C_2H_2}$	12.9	8.70	8.30	8.20	8.19	8.19	8.19	8.19	8.19
$\sigma_{C_6H_5OH}$	15.9	9.71	9.23	9.11	9.10	9.09	9.09	9.09	9.09
$\sigma_{C_5H_6}$	13.3	7.01	6.52	6.40	6.39	6.39	6.39	6.39	6.39
σ_{CO_2}	-16.5	-12.6	-12.2	-12.1	-12.1	-12.1	-12.1	-12.1	-12.1

TABLE 13.20.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $\langle \partial Y_i / \partial A_{17} \rangle$ WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$\langle \partial Y_i / \partial A_{17} \rangle$								
T	1.58	1.21	1.17	1.16	1.16	1.16	1.16	1.16	1.16
p	1.76	1.33	1.29	1.28	1.28	1.28	1.28	1.28	1.28
$\sigma_{C_6H_6}$	-15.9	-10.7	-10.2	-10.1	-10.1	-10.1	-10.1	-10.1	-10.1
σ_{OH}	26.1	20.8	20.2	20.0	20.0	20.0	20.0	20.0	20.0
$\sigma_{C_6H_5}$	-1.65	1.72	1.88	1.93	1.93	1.93	1.93	1.93	1.93
σ_H	24.3	19.3	18.8	18.6	18.6	18.6	18.6	18.6	18.6
σ_{H_2}	6.02	4.49	4.35	4.32	4.31	4.31	4.31	4.31	4.31
σ_O	27.5	21.4	20.7	20.6	20.5	20.5	20.5	20.5	20.5
σ_{H_2O}	5.42	4.47	4.35	4.32	4.32	4.32	4.32	4.32	4.32
σ_{CO}	5.43	4.32	4.20	4.17	4.16	4.16	4.16	4.16	4.16
$\sigma_{C_2H_2}$	-10.3	-6.81	-6.49	-6.40	-6.40	-6.40	-6.40	-6.40	-6.40
$\sigma_{C_6H_5OH}$	-12.9	-7.84	-7.45	-7.35	-7.35	-7.34	-7.34	-7.34	-7.34
$\sigma_{C_5H_6}$	-11.1	-6.03	-5.63	-5.53	-5.52	-5.52	-5.52	-5.52	-5.52
σ_{CO_2}	13.3	10.2	9.83	9.75	9.74	9.74	9.74	9.74	9.74

TABLE 13.21.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y_f / \partial A_{1g})$ WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM [ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y_f / \partial A_{1g})$								
T	-0.874	-0.668	-0.645	-0.640	-0.639	-0.639	-0.639	-0.639	-0.639
p	-0.971	-0.734	-0.709	-0.703	-0.702	-0.702	-0.702	-0.702	-0.702
$\sigma_{C_6H_6}$	9.29	6.32	6.07	6.01	6.00	6.00	6.00	6.00	6.00
σ_{OH}	-14.8	-11.9	-11.6	-11.5	-11.5	-11.5	-11.5	-11.5	-11.5
$\sigma_{C_6H_5}$	1.12	-0.820	-0.916	-0.941	-0.943	-0.943	-0.943	-0.943	-0.943
σ_H	-13.8	-11.0	-10.7	-10.6	-10.6	-10.6	-10.6	-10.6	-10.6
σ_{H_2}	-3.32	-2.46	-2.39	-2.37	-2.37	-2.37	-2.37	-2.37	-2.37
σ_O	-15.7	-12.4	-12.0	-11.9	-11.9	-11.9	-11.9	-11.9	-11.9
σ_{H_2O}	-2.96	-2.43	-2.36	-2.35	-2.34	-2.34	-2.34	-2.34	-2.34
σ_{CO}	-2.97	-2.35	-2.28	-2.26	-2.26	-2.26	-2.26	-2.26	-2.26
$\sigma_{C_2H_2}$	6.19	4.27	4.09	4.05	4.04	4.04	4.04	4.04	4.04
$\sigma_{C_6H_5OH}$	6.48	3.70	3.48	3.42	3.42	3.42	3.42	3.42	3.42
$\sigma_{C_3H_6}$	5.91	3.08	2.86	2.81	2.80	2.80	2.80	2.80	2.80
σ_{CO_2}	-7.54	-5.78	-5.58	-5.53	-5.53	-5.53	-5.53	-5.53	-5.53

TABLE 13.22.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y_f / \partial A_{104})$ WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM [ATOLSP = 10^{-9} EMAX, $t = 300 \mu s.$]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y_f / \partial A_{104})$								
T	1.26	0.953	0.921	0.913	0.912	0.912	0.912	0.912	0.912
p	1.40	1.05	1.01	1.00	1.00	1.00	1.00	1.00	1.00
$\sigma_{C_6H_6}$	-13.0	-8.72	-8.35	-8.26	-8.25	-8.25	-8.25	-8.25	-8.25
σ_{OH}	21.1	16.8	16.3	16.2	16.2	16.2	16.2	16.2	16.2
$\sigma_{C_6H_5}$	-1.26	1.46	1.59	1.62	1.62	1.62	1.62	1.62	1.62
σ_H	19.1	15.0	14.5	14.4	14.4	14.4	14.4	14.4	14.4
σ_{H_2}	4.10	2.86	2.75	2.73	2.73	2.72	2.72	2.72	2.72
σ_O	22.6	17.7	17.2	17.0	17.0	17.0	17.0	17.0	17.0
σ_{H_2O}	4.33	3.54	3.45	3.42	3.42	3.42	3.42	3.42	3.42
σ_{CO}	4.30	3.39	3.29	3.27	3.27	3.27	3.27	3.27	3.27
$\sigma_{C_2H_2}$	-8.63	-5.88	-5.62	-5.56	-5.55	-5.55	-5.55	-5.55	-5.55
$\sigma_{C_6H_5OH}$	-10.1	-6.05	-5.74	-5.66	-5.65	-5.65	-5.65	-5.65	-5.65
$\sigma_{C_5H_6}$	-8.82	-4.72	-4.39	-4.31	-4.31	-4.31	-4.31	-4.31	-4.31
σ_{CO_2}	10.6	8.07	7.79	7.72	7.71	7.71	7.71	7.71	7.71

TABLE 13.23.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y/\partial n_g)$
WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[ATOLSP = 10^{-9} EMAX, $t = 300 \mu s$.]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y/\partial n_g)$								
T	2.86	2.18	2.11	2.09	2.09	2.09	2.09	2.09	2.09
p	3.18	2.40	2.32	2.30	2.30	2.30	2.30	2.30	2.30
$\sigma_{C_6H_6}$	-29.1	-19.5	-18.7	-18.5	-18.4	-18.4	-18.4	-18.4	-18.4
σ_{OH}	47.5	38.0	36.9	36.6	36.6	36.6	36.6	36.6	36.6
$\sigma_{C_6H_5}$	-3.07	3.08	3.38	3.46	3.47	3.47	3.47	3.47	3.47
σ_H	44.5	35.5	34.4	34.2	34.1	34.1	34.1	34.1	34.1
σ_{H_2}	11.0	8.20	7.95	7.89	7.89	7.88	7.88	7.88	7.88
σ_O	50.3	39.3	38.1	37.7	37.7	37.7	37.7	37.7	37.7
σ_{H_2O}	9.77	8.04	7.83	7.77	7.77	7.77	7.77	7.77	7.77
σ_{CO}	9.82	7.81	7.59	7.53	7.53	7.53	7.53	7.53	7.53
$\sigma_{C_2H_2}$	-19.0	-12.8	-12.2	-12.0	-12.0	-12.0	-12.0	-12.0	-12.0
$\sigma_{C_6H_5OH}$	-23.5	-14.4	-13.7	-13.5	-13.5	-13.5	-13.5	-13.5	-13.5
$\sigma_{C_5H_6}$	-19.1	-9.91	-9.19	-9.00	-8.99	-8.99	-8.99	-8.99	-8.99
σ_{CO_2}	24.2	18.5	17.9	17.7	17.7	17.7	17.7	17.7	17.7

TABLE 13.24.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y/\partial E_g)$
WITH EMAX FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[ATOLSP = 10^{-9} EMAX, $t = 300 \mu s$.]

Variable	EMAX								
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
	$(\partial Y/\partial E_g)$								
T	2.70	2.06	2.00	1.98	1.98	1.98	1.98	1.98	1.98
p	3.00	2.28	2.20	2.18	2.18	2.18	2.18	2.18	2.18
$\sigma_{C_6H_6}$	-27.5	-18.5	-17.7	-17.5	-17.5	-17.5	-17.5	-17.5	-17.5
σ_{OH}	44.8	35.9	34.8	34.6	34.5	34.5	34.5	34.5	34.5
$\sigma_{C_6H_5}$	-2.95	2.87	3.16	3.23	3.23	3.24	3.24	3.24	3.24
σ_H	42.0	33.5	32.5	32.2	32.2	32.2	32.2	32.2	32.2
σ_{H_2}	10.4	7.74	7.50	7.45	7.44	7.44	7.44	7.44	7.44
σ_O	47.4	37.1	35.9	35.6	35.6	35.6	35.6	35.6	35.6
σ_{H_2O}	9.24	7.61	7.41	7.36	7.36	7.35	7.35	7.35	7.35
σ_{CO}	9.28	7.39	7.18	7.13	7.12	7.12	7.12	7.12	7.12
$\sigma_{C_2H_2}$	-18.0	-12.1	-11.5	-11.4	-11.4	-11.4	-11.4	-11.4	-11.4
$\sigma_{C_6H_5OH}$	-22.2	-13.6	-12.9	-12.7	-12.7	-12.7	-12.7	-12.7	-12.7
$\sigma_{C_5H_6}$	-18.2	-9.50	-8.81	-8.64	-8.63	-8.62	-8.62	-8.62	-8.62
σ_{CO_2}	22.9	17.5	16.9	16.8	16.8	16.8	16.8	16.8	16.8

TABLE 13.25.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y_f / \partial \sigma_{C_6H_6,0})$ WITH ATOLSP FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[EMAX = 10^{-6} , $t = 300 \mu s.$]

Variable	ATOLSP							
	10 ⁻¹⁵	10 ⁻¹⁴	10 ⁻¹³	10 ⁻¹²	10 ⁻¹¹	10 ⁻¹⁰	10 ⁻⁹	10 ⁻⁸
	$(\partial Y_f / \partial \sigma_{C_6H_6,0})$							
T	1.69	1.69	1.69	1.69	1.69	1.69	1.70	1.75
p	1.88	1.88	1.88	1.88	1.88	1.89	1.90	1.95
$\sigma_{C_6H_6}$	-12.6	-12.6	-12.6	-12.6	-12.6	-12.7	-12.8	-13.3
σ_{OH}	28.3	28.3	28.3	28.3	28.3	28.4	28.5	29.2
$\sigma_{C_6H_5}$	4.07	4.07	4.07	4.07	4.07	4.05	4.02	3.74
σ_H	27.3	27.3	27.3	27.3	27.3	27.4	27.5	28.1
σ_{H_2}	7.74	7.75	7.74	7.75	7.75	7.76	7.79	7.94
σ_O	28.9	28.9	28.9	28.9	28.9	28.9	29.1	29.9
σ_{H_2O}	6.59	6.59	6.59	6.59	6.59	6.60	6.63	6.75
σ_{CO}	6.43	6.43	6.43	6.43	6.44	6.45	6.48	6.61
$\sigma_{C_2H_2}$	-7.87	-7.88	-7.87	-7.88	-7.88	-7.92	-7.98	-8.35
$\sigma_{C_6H_5OH}$	-9.39	-9.39	-9.39	-9.40	-9.40	-9.45	-9.53	-10.0
$\sigma_{C_5H_6}$	-6.25	-6.25	-6.25	-6.25	-6.26	-6.30	-6.38	-6.88
σ_{CO_2}	14.0	14.0	14.0	14.0	14.0	14.0	14.1	14.5

TABLE 13.26.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $(\partial Y_f / \partial \sigma_{H_2O,0})$ WITH ATOLSP FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM
[EMAX = 10^{-6} , $t = 300 \mu s.$]

Variable	ATOLSP							
	10 ⁻¹⁵	10 ⁻¹⁴	10 ⁻¹³	10 ⁻¹²	10 ⁻¹¹	10 ⁻¹⁰	10 ⁻⁹	10 ⁻⁸
	$(\partial Y_f / \partial \sigma_{H_2O,0})$							
T	-7.52×10 ⁻⁴	-7.52×10 ⁻⁴	-7.52×10 ⁻⁴	-7.52×10 ⁻⁴	-7.52×10 ⁻⁴	-7.54×10 ⁻⁴	-7.57×10 ⁻⁴	-7.78×10 ⁻⁴
p	-7.23×10 ⁻⁴	-7.23×10 ⁻⁴	-7.23×10 ⁻⁴	-7.23×10 ⁻⁴	-7.24×10 ⁻⁴	-7.26×10 ⁻⁴	-7.29×10 ⁻⁴	-7.53×10 ⁻⁴
$\sigma_{C_6H_6}$	6.12×10 ⁻³	6.13×10 ⁻³	6.12×10 ⁻³	6.13×10 ⁻³	6.13×10 ⁻³	6.15×10 ⁻³	6.19×10 ⁻³	6.44×10 ⁻³
σ_{OH}	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.23×10 ⁻²	-1.26×10 ⁻²
$\sigma_{C_6H_5}$	-1.07×10 ⁻³	-1.07×10 ⁻³	-1.07×10 ⁻³	-1.07×10 ⁻³	-1.07×10 ⁻³	-1.06×10 ⁻³	-1.04×10 ⁻³	-9.13×10 ⁻⁴
σ_H	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.22×10 ⁻²	-1.25×10 ⁻²
σ_{H_2}	-1.40×10 ⁻³	-1.40×10 ⁻³	-1.40×10 ⁻³	-1.40×10 ⁻³	-1.40×10 ⁻³	-1.40×10 ⁻³	-1.41×10 ⁻³	-1.47×10 ⁻³
σ_O	-1.38×10 ⁻²	-1.38×10 ⁻²	-1.38×10 ⁻²	-1.38×10 ⁻²	-1.38×10 ⁻²	-1.39×10 ⁻²	-1.39×10 ⁻²	-1.43×10 ⁻²
σ_{H_2O}	2.85×10 ⁻³	2.85×10 ⁻³	2.85×10 ⁻³	2.85×10 ⁻³	2.85×10 ⁻³	2.84×10 ⁻³	2.82×10 ⁻³	2.72×10 ⁻³
σ_{CO}	-2.47×10 ⁻³	-2.47×10 ⁻³	-2.47×10 ⁻³	-2.48×10 ⁻³	-2.48×10 ⁻³	-2.48×10 ⁻³	-2.49×10 ⁻³	-2.56×10 ⁻³
$\sigma_{C_2H_2}$	4.42×10 ⁻³	4.42×10 ⁻³	4.42×10 ⁻³	4.42×10 ⁻³	4.42×10 ⁻³	4.44×10 ⁻³	4.46×10 ⁻³	4.63×10 ⁻³
$\sigma_{C_6H_5OH}$	4.14×10 ⁻³	4.14×10 ⁻³	4.14×10 ⁻³	4.14×10 ⁻³	4.14×10 ⁻³	4.16×10 ⁻³	4.19×10 ⁻³	4.42×10 ⁻³
$\sigma_{C_5H_6}$	3.17×10 ⁻³	3.17×10 ⁻³	3.17×10 ⁻³	3.17×10 ⁻³	3.17×10 ⁻³	3.19×10 ⁻³	3.22×10 ⁻³	3.44×10 ⁻³
σ_{CO_2}	-5.84×10 ⁻³	-5.84×10 ⁻³	-5.84×10 ⁻³	-5.84×10 ⁻³	-5.84×10 ⁻³	-5.86×10 ⁻³	-5.88×10 ⁻³	-6.05×10 ⁻³

TABLE 13.27.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $\langle \partial Y_i / \partial T_0 \rangle$ WITH ATOLSP FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM [EMAX = 10^{-6} , $t = 300 \mu s$.]

Variable	ATOLSP							
	10^{-15}	10^{-14}	10^{-13}	10^{-12}	10^{-11}	10^{-10}	10^{-9}	10^{-8}
	$\langle \partial Y_i / \partial T_0 \rangle$							
T	70.9	70.9	70.9	70.9	71.0	71.1	71.5	73.3
p	78.0	78.0	78.0	78.0	78.0	78.2	78.6	80.7
$\sigma_{C_6H_6}$	-625	-625	-625	-625	-626	-628	-631	-655
σ_{OH}	1.23×10^3	1.23×10^3	1.23×10^3	1.23×10^3	1.23×10^3	1.23×10^3	1.23×10^3	1.26×10^3
$\sigma_{C_6H_5}$	110	110	110	110	110	109	108	95.4
σ_H	1.14×10^3	1.14×10^3	1.14×10^3	1.14×10^3	1.14×10^3	1.14×10^3	1.15×10^3	1.18×10^3
σ_{H_2}	262	262	262	262	262	263	264	270
σ_O	1.26×10^3	1.26×10^3	1.26×10^3	1.27×10^3	1.27×10^3	1.27×10^3	1.27×10^3	1.31×10^3
σ_{H_2O}	262	262	262	262	262	262	263	268
σ_{CO}	253	253	253	253	253	253	254	260
$\sigma_{C_2H_2}$	-404	-404	-404	-404	-404	-405	-408	-424
$\sigma_{C_6H_5OH}$	-448	-448	-448	-448	-448	-450	-453	-475
$\sigma_{C_3H_6}$	-321	-321	-321	-321	-322	-323	-326	-348
σ_{CO_2}	595	595	595	595	595	597	600	615

TABLE 13.28.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT $\langle \partial Y_i / \partial A_1 \rangle$ WITH ATOLSP FOR BENZENE-OXYGEN-ARGON EXAMPLE PROBLEM [EMAX = 10^{-6} , $t = 300 \mu s$.]

Variable	ATOLSP							
	10^{-15}	10^{-14}	10^{-13}	10^{-12}	10^{-11}	10^{-10}	10^{-9}	10^{-8}
	$\langle \partial Y_i / \partial A_1 \rangle$							
T	1.96	1.96	1.96	1.96	1.96	1.97	1.97	2.03
p	2.16	2.16	2.16	2.16	2.16	2.16	2.17	2.23
$\sigma_{C_6H_6}$	-17.5	-17.5	-17.5	-17.5	-17.5	-17.6	-17.7	-18.3
σ_{OH}	33.9	33.9	33.9	33.9	34.0	34.0	34.2	35.0
$\sigma_{C_6H_5}$	2.86	2.86	2.86	2.85	2.85	2.83	2.80	2.45
σ_H	31.6	31.6	31.6	31.6	31.6	31.7	31.8	32.5
σ_{H_2}	7.22	7.22	7.22	7.23	7.23	7.24	7.28	7.44
σ_O	34.9	35.0	34.9	35.0	35.0	35.1	35.2	36.1
σ_{H_2O}	7.30	7.30	7.30	7.30	7.31	7.32	7.35	7.49
σ_{CO}	7.06	7.06	7.06	7.06	7.06	7.08	7.11	7.26
$\sigma_{C_2H_2}$	-11.1	-11.1	-11.1	-11.1	-11.1	-11.1	-11.2	-11.7
$\sigma_{C_6H_5OH}$	-12.4	-12.4	-12.4	-12.4	-12.4	-12.5	-12.5	-13.1
$\sigma_{C_3H_6}$	-8.84	-8.84	-8.84	-8.85	-8.85	-8.90	-8.99	-9.59
σ_{CO_2}	16.5	16.5	16.5	16.5	16.5	16.6	16.6	17.1

TABLE 13.29.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT
 $\langle \partial Y_f / \partial n_g \rangle$ WITH ATOLSP FOR BENZENE-OXYGEN-ARGON

EXAMPLE PROBLEM
 [EMAX = 10^{-6} , $t = 300 \mu s.$]

Variable	ATOLSP							
	10 ⁻¹⁵	10 ⁻¹⁴	10 ⁻¹³	10 ⁻¹²	10 ⁻¹¹	10 ⁻¹⁰	10 ⁻⁹	10 ⁻⁸
	$\langle \partial Y_f / \partial n_g \rangle$							
<i>T</i>	2.09	2.09	2.09	2.09	2.09	2.10	2.10	2.16
<i>p</i>	2.30	2.30	2.30	2.30	2.30	2.31	2.32	2.38
$\sigma_{C_6H_6}$	-18.4	-18.5	-18.4	-18.5	-18.5	-18.5	-18.6	-19.3
σ_{OH}	36.6	36.6	36.6	36.6	36.6	36.7	36.8	37.7
$\sigma_{C_6H_5}$	3.47	3.47	3.47	3.46	3.46	3.43	3.40	3.02
σ_H	34.1	34.1	34.1	34.1	34.2	34.2	34.4	35.2
σ_{H_2}	7.89	7.89	7.89	7.89	7.89	7.91	7.94	8.12
σ_O	37.7	37.7	37.7	37.7	37.7	37.8	38.0	39.0
σ_{H_2O}	7.77	7.77	7.77	7.77	7.77	7.79	7.82	7.97
σ_{CO}	7.53	7.53	7.53	7.53	7.53	7.55	7.58	7.75
$\sigma_{C_2H_2}$	-12.0	-12.0	-12.0	-12.0	-12.0	-12.1	-12.1	-12.6
$\sigma_{C_6H_5OH}$	-13.5	-13.5	-13.5	-13.5	-13.5	-13.5	-13.6	-14.3
$\sigma_{C_5H_6}$	-8.99	-8.99	-8.99	-9.00	-9.00	-9.06	-9.15	-9.79
σ_{CO_2}	17.7	17.7	17.7	17.7	17.7	17.8	17.8	18.3

TABLE 13.30.—VARIATION OF NORMALIZED SENSITIVITY COEFFICIENT
 $\langle \partial Y_f / \partial E_g \rangle$ WITH ATOLSP FOR BENZENE-OXYGEN-ARGON

EXAMPLE PROBLEM
 [EMAX = 10^{-6} , $t = 300 \mu s.$]

Variable	ATOLSP							
	10 ⁻¹⁵	10 ⁻¹⁴	10 ⁻¹³	10 ⁻¹²	10 ⁻¹¹	10 ⁻¹⁰	10 ⁻⁹	10 ⁻⁸
	$\langle \partial Y_f / \partial E_g \rangle$							
<i>T</i>	1.98	1.98	1.98	1.98	1.98	1.98	1.99	2.05
<i>p</i>	2.18	2.18	2.18	2.18	2.18	2.18	2.19	2.25
$\sigma_{C_6H_6}$	-17.5	-17.5	-17.5	-17.5	-17.5	-17.5	-17.6	-18.3
σ_{OH}	34.5	34.5	34.5	34.5	34.6	34.6	34.8	35.6
$\sigma_{C_6H_5}$	3.23	3.23	3.23	3.23	3.23	3.20	3.17	2.82
σ_H	32.2	32.2	32.2	32.2	32.2	32.3	32.4	33.2
σ_{H_2}	7.44	7.44	7.44	7.44	7.45	7.46	7.49	7.66
σ_O	35.6	35.6	35.6	35.6	35.6	35.7	35.9	36.8
σ_{H_2O}	7.36	7.36	7.36	7.36	7.36	7.37	7.40	7.55
σ_{CO}	7.12	7.12	7.12	7.12	7.13	7.14	7.17	7.33
$\sigma_{C_2H_2}$	-11.4	-11.4	-11.4	-11.4	-11.4	-11.4	-11.5	-11.9
$\sigma_{C_6H_5OH}$	-12.7	-12.7	-12.7	-12.7	-12.7	-12.8	-12.8	-13.5
$\sigma_{C_5H_6}$	-8.63	-8.63	-8.63	-8.63	-8.64	-8.69	-8.78	-9.38
σ_{CO_2}	16.8	16.8	16.8	16.8	16.8	16.8	16.9	17.3

Appendix A

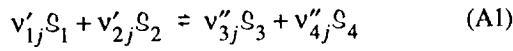
Ordinary Differential Equations and Jacobian Matrix Elements

In this appendix we list the ordinary differential equations used in the code for static and one-dimensional flow kinetics problems. For both problem types and for sensitivity analysis computations and the perfectly stirred reactor (PSR) problem, we give the partial derivatives required by the numerical solution procedures.

Net Reaction Rates and Partial Derivatives

Non-Third-Body Reaction

We write the general reaction equation in the form



where 1 and 2 are the specific reactant subscripts and 3 and 4 the specific product subscripts for reaction j , v'_{ij} and v''_{ij} are the stoichiometric coefficients of reactant species i and product species i , respectively, in reaction j , and S_i is the chemical symbol for species i . The net molar rate of reaction j per unit volume r_j is given by (see eqs. (2.16) to (2.18) in chapter 2 of part I)

$$r_j = R_j - R_{-j} \quad (\text{A2})$$

where the molar forward R_j and reverse R_{-j} rates per unit volume of reaction j are

$$R_j = k_j \rho^{v'_{1j} + v'_{2j}} \sigma_1^{v'_{1j}} \sigma_2^{v'_{2j}} \quad (\text{A3})$$

$$R_{-j} = k_{-j} \rho^{v''_{3j} + v''_{4j}} \sigma_3^{v''_{3j}} \sigma_4^{v''_{4j}} \quad (\text{A4})$$

In these equations ρ is the mixture mass density, σ_i is the mole number of species i (i.e., number of moles of species i per unit mass of mixture), and k_j and k_{-j} are the forward and reverse rate coefficients of reaction j , with $k_{-j} = k_j K_{c,j}$, where $K_{c,j}$ is the concentration equilibrium constant for reaction j (see eqs. (8.5) and (8.6)). In equations (A3) and (A4) we have used the fact that the molar concentration of species ℓ , C_ℓ , can be written as

$$C_\ell = \rho \sigma_\ell \quad (\text{A5})$$

The partial derivatives of r_j with respect to the $\{\sigma_\ell\}$, temperature T , and density ρ are as follows:

$$\frac{\partial r_j}{\partial \sigma_\ell} = \frac{v'_{\ell j} R_j}{\sigma_\ell}, \quad \ell = 1, 2 \quad (\text{A6})$$

$$\frac{\partial r_j}{\partial \sigma_\ell} = -\frac{v''_{\ell j} R_j}{\sigma_\ell}, \quad \ell = 3, 4 \quad (\text{A7})$$

$$\frac{\partial r_j}{\partial T} = r_j \frac{d \ln k_j}{dT} + R_{-j} \frac{d \ln K_{c,j}}{dT} \quad (\text{A8})$$

$$\frac{\partial r_j}{\partial \rho} = \frac{v'_{1j} + v'_{2j}}{\rho} R_j - \frac{v''_{3j} + v''_{4j}}{\rho} R_{-j} \quad (\text{A9})$$

In equation (A8) the derivative $d \ln K_{c,j} / dT$ is given by

$$\frac{d \ln K_{c,j}}{dT} = \sum_{\ell=3}^4 v''_{\ell j} \left(\frac{h_\ell}{RT^2} - \frac{1}{T} \right) - \sum_{\ell=1}^2 v'_{\ell j} \left(\frac{h_\ell}{RT^2} - \frac{1}{T} \right) \quad (\text{A10})$$

A. Ordinary Differential Equations and Jacobian Matrix Elements

where h_ℓ is the molar-specific enthalpy of species ℓ , R is the universal gas constant, and $d \ln k_j / dT$ depends on the form of the expression (eq. (8.3) or (8.4)) used for k_j . For a rate coefficient given by equation (8.3)

$$\frac{d \ln k_j}{dT} = \frac{n_j}{T} + \frac{E_j}{RT^2} \quad (\text{A11})$$

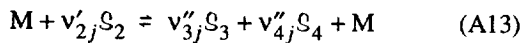
and for k_j of the form given by equation (8.4)

$$\frac{d \ln k_j}{dT} = \frac{n_j}{T} + c_j \quad (\text{A12})$$

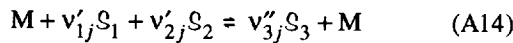
If either species S_1 or S_4 , or both, are absent, the required derivatives can be obtained from equations (A3) to (A10) by simply setting the appropriate stoichiometric coefficient or coefficients equal to zero. Similarly setting k_{-j} equal to zero gives the correct derivatives for an irreversible reaction.

Third-Body Catalyzed Reaction

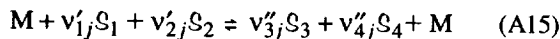
Two types of third-body reactions are considered: dissociation, written as



and recombination, written as



The catalyst collision partner M can be any species present in the mixture. For convenience, we generalize the two reaction types as follows:



where v'_{1j} and v''_{4j} are equal to zero for dissociation and recombination reactions, respectively.

The total molar concentration C_m of catalyst species is given by

$$C_m = \rho M_j \quad (\text{A16})$$

where M_j is the third-body efficiency factor for the j th reaction

$$M_j = \sum_{\ell=1}^{NS} m_{\ell j} \sigma_\ell \quad (\text{A17})$$

NS is the total number of (reacting and inert) species, and $m_{\ell j}$ the relative collisional efficiency of species ℓ in reaction j ,

is a correction for the preexponential factor A_j of the rate coefficient expression (eq. (8.3) or (8.4)) and is defined by

$$m_{\ell j} = \frac{(A_j)_\ell}{(A_j)_0} \quad (\text{A18})$$

Here $(A_j)_\ell$ and $(A_j)_0$ are the A_j values for species ℓ and a reference species (usually nitrogen or an inert species such as argon) as the collision partner, respectively. It is $(A_j)_0$ that is specified in the rate coefficient data for the reaction.

The molar forward and reverse rates per unit volume of reaction (A15) are

$$R_j = k_j M_j \rho^{v'_{1j} + v'_{2j} + 1} \sigma_1^{v'_{1j}} \sigma_2^{v'_{2j}} \quad (\text{A19})$$

$$R_{-j} = k_{-j} M_j \rho^{v''_{3j} + v''_{4j} + 1} \sigma_3^{v''_{3j}} \sigma_4^{v''_{4j}} \quad (\text{A20})$$

The net molar reaction rate per unit volume is given by equation (A2). The partial derivatives are as follows:

$$\frac{\partial r_j}{\partial \sigma_\ell} = \frac{m_{\ell j}}{M_j} r_j + \frac{v'_{\ell j}}{\sigma_\ell} R_j, \quad \ell = 1, 2 \quad (\text{A21})$$

$$\frac{\partial r_j}{\partial \sigma_\ell} = \frac{m_{\ell j}}{M_j} r_j - \frac{v''_{\ell j}}{\sigma_\ell} R_{-j}, \quad \ell = 3, 4 \quad (\text{A22})$$

$$\frac{\partial r_j}{\partial \rho} = \frac{v'_{1j} + v'_{2j} + 1}{\rho} R_j - \frac{v''_{3j} + v''_{4j} + 1}{\rho} R_{-j} \quad (\text{A23})$$

and $\partial r_j / \partial T$ is given by equation (A8). Now M_j is a function of the mole numbers of all NS species (see eq. (A17)), even those participating in the reaction as only third-body catalysts. Hence partial derivatives of r_j with respect to the mole numbers of the latter species must also be included. These derivatives are given by

$$\frac{\partial r_j}{\partial \sigma_\ell} = \frac{m_{\ell j}}{M_j} r_j \quad (\text{A24})$$

where species ℓ is a third-body catalyst.

For a third-body dissociation reaction species S_1 is absent and the required equations are obtained by setting v'_{1j} equal to zero in equations (A19) to (A23). Similarly setting v''_{4j} equal to zero gives the correct equations for a third-body recombination reaction, which does not contain species S_4 . Finally, for an irreversible reaction of either type, $k_{-j} = 0$ produces the desired results.

Ordinary Differential Equations

For ease of presentation in the following sections, we write the system of ordinary differential equations (ODE's), equation (3.1) in chapter 3 of part I, in component form:

$$\frac{dy_i}{d\xi} = f_i(y_1, \dots, y_N), \quad i = 1, \dots, N \quad (\text{A25})$$

where the independent variable ξ is either time t or distance x , the dependent variables $\{y_i\}$ are as follows:

$$y_i = \sigma_i, \quad i = 1, \dots, \text{NRS or NS} \quad (\text{A26})$$

and NRS is the number of reacting species. The remaining dependent variables, if any, are one or more of T , ρ , and velocity V depending on the type of problem being solved. The number N of ODE's also depends on the problem type. The following simplified notation will be used for the derivatives:

$$f_i = \frac{d\sigma_i}{d\xi}, \quad i = 1, \dots, \text{NRS or NS} \quad (\text{A27})$$

$$f_{NT} = \frac{dT}{d\xi} \quad (\text{A28})$$

$$f_{NR} = \frac{d\rho}{d\xi} \quad (\text{A29})$$

$$f_{NV} = \frac{dV}{d\xi} \quad (\text{A30})$$

Species Differential Equations

The ODE for the mole number of the i th reacting species can be written as

$$f_i = \frac{1}{\rho\chi} \sum_{j=1}^{\text{NR}} \omega_{ij}, \quad i = 1, \dots, \text{NRS} \quad (\text{A31})$$

where χ depends on the independent variable

$$\chi = \begin{cases} 1, & \xi = t \\ V, & \xi = x \end{cases} \quad (\text{A32})$$

NR is the number of reactions, and ω_{ij} is the net molar rate of formation of species i per unit volume by reaction j

$$\omega_{ij} = (v''_{ij} - v'_{ij})r_j \quad (\text{A33})$$

If sensitivity analysis with respect to the initial mole number of any inert species is required, ODE's are included for the mole numbers of all inert species. Each such ODE is given by $f_i = 0$ ($i = 1, \dots, \text{NINERT}$, where NINERT is the number of inert species).

Static Reaction

Assigned pressure.—We define

$$\mathcal{S}_1 = M_w \sum_{i=1}^{\text{NRS}} f_i \quad (\text{A34})$$

$$\mathcal{S}_2 = \sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} f_i \quad (\text{A35})$$

and

$$\mathcal{D} = \frac{\dot{Q}}{mRT} \quad (\text{A36})$$

In these equations M_w is the mixture molar mass, \dot{Q} is the heat transfer rate from the reacting mixture to its surroundings, and m is the mass of the mixture. The required temperature and density derivatives can then be written as

$$f_{NT} = \frac{T}{c_p/R} \left(\frac{\sigma_m}{p} \frac{dp}{dt} - \mathcal{S}_2 - \mathcal{D} \right) \quad (\text{A37})$$

$$f_{NR} = \rho \left(\frac{1}{p} \frac{dp}{dt} - \mathcal{S}_1 - \frac{f_{NT}}{T} \right) \quad (\text{A38})$$

where c_p is the mixture mass-specific heat, equation (12.5), $\sigma_m (= 1/M_w)$ is the sum of the species mole numbers, and p is the pressure. In equation (A38) f_{NT} is either given by equation (A37) or computed from the temperature profile for an assigned-temperature problem. In the latter case equation (A37) is not used. However, if the temperature is constant and sensitivity coefficients with respect to its initial value are required, the temperature ODE $f_{NT} = 0$ is included in the equation set.

Constant density.—For this case, we redefine \mathcal{S}_1 as follows:

$$\mathcal{S}_1 = \sum_{i=1}^{\text{NRS}} f_i \quad (\text{A39})$$

A. Ordinary Differential Equations and Jacobian Matrix Elements

The temperature derivative is then given by

$$f_{NT} = \frac{T}{c_p/R - \sigma_m} (S_1 - S_2 - \mathcal{D}) \quad (\text{A40})$$

where S_2 and \mathcal{D} are given by equations (A35) and (A36), respectively.

If temperature is assigned, the species ODE's completely specify the problem and equation (A40) is not needed. However, if sensitivity analysis with respect to the initial temperature is required for a constant-temperature problem, the comments given for the assigned-pressure static problem apply. Similarly, if sensitivity coefficients with respect to the initial density must be computed, the density ODE $f_{NR} = 0$ is used.

Flow Problem

Assigned pressure.— We define S_1 and S_2 by equations (A34) and (A35), respectively, and \mathcal{D} as

$$\mathcal{D} = \frac{V}{\chi} \frac{\dot{Q}'}{\dot{m}RT} \quad (\text{A41})$$

where \dot{Q}' is the heat transfer rate per unit length from the reacting mixture to its surroundings, \dot{m} is the mass flow rate, and χ depends on the independent variable (see eq. (A32)). The temperature, density, and velocity derivatives can then be expressed as

$$f_{NT} = \frac{T}{c_p/R} \left(\frac{\sigma_m}{p} \frac{dp}{d\xi} - S_2 - \mathcal{D} \right) \quad (\text{A42})$$

$$f_{NR} = \rho \left(\frac{1}{p} \frac{dp}{d\xi} - S_1 - \frac{f_{NT}}{T} \right) \quad (\text{A43})$$

$$f_{NV} = -\frac{1}{\rho V} \frac{dp}{d\xi} C_f \quad (\text{A44})$$

where $C_f (= 1.01325 \times 10^6 \text{ dyne cm}^{-2}/\text{atm})$ is a units conversion factor.

If temperature is assigned, equation (A42) is not used and f_{NT} in equation (A43) is computed from the temperature profile. For a constant-pressure problem the velocity is also constant (see eq. (A44)), and so a velocity ODE is not solved.

Assigned area.—For this problem type also S_1 and S_2 are given by equations (A34) and (A35), respectively. In addition, we define the quantities

$$\mathcal{A} = S_1 - \mathcal{B} \quad (\text{A45})$$

$$\mathcal{B} = \frac{\gamma - 1}{\gamma} M_w S_2 \quad (\text{A46})$$

$$\mathcal{D} = M_w \frac{\gamma - 1}{\gamma} \frac{V}{\chi} \frac{\dot{Q}'}{\dot{m}RT} \quad (\text{A47})$$

where γ is the frozen mixture specific heat ratio, equation (12.4). The temperature, density, and velocity derivatives are then as follows:

$$f_{NT} = -T \left[\frac{(\gamma - 1) \mathcal{M}^2}{\mathcal{M}^2 - 1} \left(\frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) + \mathcal{B} + \mathcal{D} \right] \quad (\text{A48})$$

$$f_{NR} = -\rho \left[\frac{\mathcal{M}^2}{\mathcal{M}^2 - 1} \left(\frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) + \mathcal{A} - \mathcal{D} \right] \quad (\text{A49})$$

$$f_{NV} = \frac{V}{\mathcal{M}^2 - 1} \left(\frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) \quad (\text{A50})$$

where \mathcal{M} is the Mach number, equation (12.3), and A is the area.

Assigned area and temperature.—Here S_1 is given by equation (A34) and the density and velocity derivatives are

$$f_{NR} = \frac{-\rho}{\gamma \mathcal{M}^2 - 1} \left(\frac{\gamma \mathcal{M}^2}{A} \frac{dA}{d\xi} - S_1 - \frac{f_{NT}}{T} \right) \quad (\text{A51})$$

$$f_{NV} = -V \left(\frac{1}{A} \frac{dA}{d\xi} + \frac{f_{NR}}{\rho} \right) \quad (\text{A52})$$

where f_{NT} is either equal to zero for a constant-temperature problem or computed from the assigned-temperature profile.

Jacobian Matrix Elements

We list here the equations used to calculate the Jacobian matrix elements for the various static and flow problem types considered in this work. The equations are obtained by differentiating the appropriate equations for $\{f_i\}$, f_{NT} , f_{NR} , and f_{NV} . For the sake of generality we derive expressions for all possible Jacobian matrix elements for each problem type. However, the elements that are actually needed, and therefore computed, depend on the variables, including sensitivity analysis data, specified for the problem. The ODE solver

requires the partial derivatives $\partial f_i / \partial y_j$, $i, j = 1, \dots, N$. The ODE set contains quantities that are required for the problem and vary during the course of the reaction. A constant quantity is also included in the ODE list if sensitivity analysis with respect to its initial value is required. For each such variable the partial derivatives of its temporal derivative with respect to all variables are zero, and therefore the equations given in this section are not used. They are, however, used to compute the partial derivatives with respect to this variable.

Partial Derivatives of Species Formation Rates

The partial derivatives of the species formation rates $\{f_i\}$ with respect to species mole numbers are computed from equations (A31) and (A33). For any reacting species ℓ

$$\frac{\partial f_i}{\partial \sigma_\ell} = \frac{1}{\rho\chi} \sum_{j=1}^{NR} (v''_{ij} - v'_{ij}) \frac{\partial r_j}{\partial \sigma_\ell}, \quad i, \ell = 1, \dots, \text{NRS} \quad (\text{A53})$$

where χ depends on the independent variable (see eq. (A32)) and $\partial r_j / \partial \sigma_\ell$ depends on the reaction type and whether species ℓ is a reactant or a product (see eqs. (A6), (A7), (A21), and (A22)). If partial derivatives with respect to an inert species mole number are required, they are also computed by using equation (A53) but with $\partial r_j / \partial \sigma_\ell$ given by equation (A24).

Equations (A31) and (A33) can also be used to derive the following expressions for $\{\partial f_i / \partial T\}$, $\{\partial f_i / \partial \rho\}$, and for a flow problem $\{\partial f_i / \partial V\}$:

$$\frac{\partial f_i}{\partial T} = \frac{1}{\rho\chi} \sum_{j=1}^{NR} (v''_{ij} - v'_{ij}) \frac{\partial r_j}{\partial T}, \quad i = 1, \dots, \text{NRS} \quad (\text{A54})$$

$$\frac{\partial f_i}{\partial \rho} = \frac{1}{\rho\chi} \sum_{j=1}^{NR} (v''_{ij} - v'_{ij}) \frac{\partial r_j}{\partial \rho} - \frac{f_i}{\rho}, \quad i = 1, \dots, \text{NRS} \quad (\text{A55})$$

$$\frac{\partial f_i}{\partial V} = \begin{cases} 0, & \xi = t \\ -\frac{f_i}{V}, & \xi = x \end{cases}, \quad i = 1, \dots, \text{NRS} \quad (\text{A56})$$

where $\partial r_j / \partial T$ is given by equation (A8) and $\partial r_j / \partial \rho$ depends on the reaction type (see eqs. (A9) and (A23)).

Static Reaction

Assigned pressure.—The partial derivatives of f_{NT} are obtained from equation (A37), after first replacing the term σ_m / p by $1 / (\rho RT)$ using the ideal-gas law:

$$\frac{\partial f_{NT}}{\partial \sigma_\ell} = -\frac{1}{c_p / R} \left[T \left(\frac{\partial \mathcal{S}_2}{\partial \sigma_\ell} + \frac{\partial \mathcal{D}}{\partial \sigma_\ell} \right) + f_{NT} \frac{c_{p,\ell}}{R} \right], \quad \ell = 1, \dots, \text{NRS or NS} \quad (\text{A57})$$

$$\frac{\partial f_{NT}}{\partial T} = -\frac{1}{c_p / R} \left[T \left(\sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} \frac{\partial f_i}{\partial T} + \frac{1}{RT} \frac{\partial \dot{Q}}{\partial T} \right) + \sum_{i=1}^{\text{NRS}} \frac{c_{p,i}}{R} f_i + f_{NT} \frac{\partial c_p}{\partial T} \frac{1}{R} \right] \quad (\text{A58})$$

$$\frac{\partial f_{NT}}{\partial \rho} = -\frac{T}{c_p / R} \left(\frac{\sigma_m}{\rho} \frac{1}{p} \frac{dp}{dt} + \frac{\partial \mathcal{S}_2}{\partial \rho} + \frac{\partial \mathcal{D}}{\partial \rho} \right) \quad (\text{A59})$$

In these equations $c_{p,i}$ is the molar-specific heat of species i . We use equation (A38) to compute the partial derivatives of f_{NR} :

$$\frac{\partial f_{NR}}{\partial \sigma_\ell} = -\rho \left(\frac{\partial \mathcal{S}_1}{\partial \sigma_\ell} + \frac{1}{T} \frac{\partial f_{NT}}{\partial \sigma_\ell} \right), \quad \ell = 1, \dots, \text{NRS or NS} \quad (\text{A60})$$

$$\frac{\partial f_{NR}}{\partial T} = -\rho \left[\frac{\partial \mathcal{S}_1}{\partial T} + \frac{1}{T} \left(\frac{\partial f_{NT}}{\partial T} - \frac{f_{NT}}{T} \right) \right] \quad (\text{A61})$$

$$\frac{\partial f_{NR}}{\partial \rho} = \frac{f_{NR}}{\rho} - \rho \left(\frac{\partial \mathcal{S}_1}{\partial \rho} + \frac{1}{T} \frac{\partial f_{NT}}{\partial \rho} \right) \quad (\text{A62})$$

In equations (A57) to (A62) the required partial derivatives of \mathcal{S}_1 , \mathcal{S}_2 , and c_p are as follows:

$$\frac{\partial \mathcal{S}_1}{\partial \sigma_\ell} = M_w \left(\sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial \sigma_\ell} - \mathcal{S}_1 \right), \quad \ell = 1, \dots, \text{NRS or NS} \quad (\text{A63})$$

$$\frac{\partial \mathcal{S}_1}{\partial T} = M_w \sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial T} \quad (\text{A64})$$

A. Ordinary Differential Equations and Jacobian Matrix Elements

$$\frac{\partial \mathcal{S}_1}{\partial \rho} = M_w \sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial \rho} \quad (\text{A65})$$

$$\frac{\partial \mathcal{S}_2}{\partial \sigma_\ell} = \sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} \frac{\partial f_i}{\partial \sigma_\ell}, \quad \ell = 1, \dots, \text{NRS or NS} \quad (\text{A66})$$

$$\frac{\partial \mathcal{S}_2}{\partial \rho} = \sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} \frac{\partial f_i}{\partial \rho} \quad (\text{A67})$$

$$\frac{\partial}{\partial T} \frac{c_p}{R} = \sum_{i=1}^{\text{NS}} \sigma_i \frac{d}{dT} \frac{c_{p,i}}{R} \quad (\text{A68})$$

The partial derivatives of the heat transfer terms depend on the heat transfer model. If the heat transfer rate is specified by the polynomial expression, equation (8.21), the required derivatives are as follows:

$$\frac{\partial \mathcal{D}}{\partial \sigma_\ell} = 0, \quad \ell = 1, \dots, \text{NRS or NS} \quad (\text{A69})$$

$$\frac{\partial \dot{Q}}{\partial T} = H_{T,1} + 2H_{T,2}T + 3H_{T,3}T^2 + 4H_{T,4}T^3 \quad (\text{A70})$$

$$\frac{\partial \mathcal{D}}{\partial \rho} = 0 \quad (\text{A71})$$

For presentational convenience we will refer to this heat transfer model as HT1. For the static Otto-cycle problem the heat transfer rate per unit mass of mixture is given by the empirical correlation, equation (8.29). For this heat transfer model, which is hereinafter referred to as HT2, the heat transfer coefficient and hence the required partial derivatives depend on the nature of the flow inside the cylinder. For laminar flow the following partial derivatives are used in the code:

$$\frac{\partial \mathcal{D}}{\partial \sigma_\ell} = 0, \quad \ell = 1, \dots, \text{NRS or NS} \quad (\text{A72})$$

$$\frac{\partial}{\partial T} \frac{\dot{Q}}{m} = \frac{4\mathfrak{H}}{B\rho} \quad (\text{A73})$$

$$\frac{\partial \mathcal{D}}{\partial \rho} = -\frac{\mathcal{D}}{\rho} \quad (\text{A74})$$

Equations (A72) and (A73) are also used in the situation of turbulent flow, but

$$\frac{\partial \mathcal{D}}{\partial \rho} = -0.2 \frac{\mathcal{D}}{\rho} \quad (\text{A75})$$

In equations (A72) and (A73), we have neglected the implicit dependence of \dot{Q} on $\{\sigma_i\}$ and T through \mathfrak{H} (see eqs. (8.24) to (8.30)). This approximation was made for the sake of simplicity to avoid the computational cost of evaluating the exact derivatives, which are excessively complicated (see eqs. (8.37) to (8.44)).

Constant density.—For this problem type the only additional partial derivatives required are those of f_{NT} , which are evaluated from equation (A40):

$$\frac{\partial f_{\text{NT}}}{\partial \sigma_\ell} = \frac{1}{c_p/R - \sigma_m} \left[T \left(\frac{\partial \mathcal{S}_1}{\partial \sigma_\ell} - \frac{\partial \mathcal{S}_2}{\partial \sigma_\ell} - \frac{\partial \mathcal{D}}{\partial \sigma_\ell} \right) - f_{\text{NT}} \left(\frac{c_{p,\ell}}{R} - 1 \right) \right], \quad \ell = 1, \dots, \text{NRS or NS} \quad (\text{A76})$$

$$\begin{aligned} \frac{\partial f_{\text{NT}}}{\partial T} = \frac{1}{c_p/R - \sigma_m} \left[T \left(\frac{\partial \mathcal{S}_1}{\partial T} - \sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} \frac{\partial f_i}{\partial T} - \frac{1}{RT} \frac{\partial \dot{Q}}{\partial T} \frac{1}{m} \right) \right. \\ \left. + \mathcal{S}_1 - \sum_{i=1}^{\text{NRS}} \frac{c_{p,i}}{R} f_i - f_{\text{NT}} \frac{\partial}{\partial T} \frac{c_p}{R} \right] \quad (\text{A77}) \end{aligned}$$

$$\frac{\partial f_{\text{NT}}}{\partial \rho} = \frac{T}{c_p/R - \sigma_m} \left(\frac{\partial \mathcal{S}_1}{\partial \rho} - \frac{\partial \mathcal{S}_2}{\partial \rho} - \frac{\partial \mathcal{D}}{\partial \rho} \right) \quad (\text{A78})$$

In equations (A76) to (A78) the partial derivatives of \mathcal{S}_1 are

$$\frac{\partial \mathcal{S}_1}{\partial \sigma_\ell} = \sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial \sigma_\ell}, \quad \ell = 1, \dots, \text{NRS or NS} \quad (\text{A79})$$

$$\frac{\partial \mathcal{S}_1}{\partial T} = \sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial T} \quad (\text{A80})$$

$$\frac{\partial \mathcal{S}_1}{\partial \rho} = \sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial \rho} \quad (\text{A81})$$

The terms $\partial \mathcal{S}_2 / \partial \sigma_\ell$, $\partial \mathcal{S}_2 / \partial \rho$, and $\partial(c_p/R) / \partial T$ are given, respectively, by equations (A66), (A67), and (A68). Finally, because HT1 is the only heat transfer model allowed for this problem type, the derivatives of the heat transfer terms are given by equations (A69) to (A71).

Flow Problem

Assigned pressure.—In this case the required partial derivatives are calculated from equations (A42) to (A44), after first replacing the term σ_m/ρ by $1/(\rho RT)$ in equation (A42), and are as follows:

$$\frac{\partial f_{\text{NT}}}{\partial \sigma_\ell} = -\frac{1}{c_p/R} \left[T \left(\frac{\partial \mathcal{S}_2}{\partial \sigma_\ell} + \frac{\partial \mathcal{D}}{\partial \sigma_\ell} \right) + f_{\text{NT}} \frac{c_{p,\ell}}{R} \right], \quad \ell = 1, \dots, \text{NRS} \quad (\text{A82})$$

$$\frac{\partial f_{\text{NT}}}{\partial T} = -\frac{1}{c_p/R} \left[T \left(\sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} \frac{\partial f_i}{\partial T} + \frac{1}{\dot{m}RT} \frac{V}{\chi} \frac{\partial \dot{Q}'}{\partial T} \right) + \sum_{i=1}^{\text{NRS}} \frac{c_{p,i}}{R} f_i + f_{\text{NT}} \frac{\partial c_p}{\partial T} \frac{1}{R} \right] \quad (\text{A83})$$

$$\frac{\partial f_{\text{NT}}}{\partial \rho} = -\frac{T}{c_p/R} \left(\frac{\sigma_m}{\rho} \frac{1}{p} \frac{dp}{d\xi} + \frac{\partial \mathcal{S}_2}{\partial \rho} + \frac{\partial \mathcal{D}}{\partial \rho} \right) \quad (\text{A84})$$

$$\frac{\partial f_{\text{NT}}}{\partial V} = -\frac{T}{c_p/R} \left(\frac{\partial \mathcal{S}_2}{\partial V} + \frac{\partial \mathcal{D}}{\partial V} \right) \quad (\text{A85})$$

$$\frac{\partial f_{\text{NR}}}{\partial \sigma_\ell} = -\rho \left(\frac{\partial \mathcal{S}_1}{\partial \sigma_\ell} + \frac{1}{T} \frac{\partial f_{\text{NT}}}{\partial \sigma_\ell} \right), \quad \ell = 1, \dots, \text{NRS} \quad (\text{A86})$$

$$\frac{\partial f_{\text{NR}}}{\partial T} = -\rho \left[\frac{\partial \mathcal{S}_1}{\partial T} + \frac{1}{T} \left(\frac{\partial f_{\text{NT}}}{\partial T} - \frac{f_{\text{NT}}}{T} \right) \right] \quad (\text{A87})$$

$$\frac{\partial f_{\text{NR}}}{\partial \rho} = \frac{f_{\text{NR}}}{\rho} - \rho \left(\frac{\partial \mathcal{S}_1}{\partial \rho} + \frac{1}{T} \frac{\partial f_{\text{NT}}}{\partial \rho} \right) \quad (\text{A88})$$

$$\frac{\partial f_{\text{NR}}}{\partial V} = -\rho \left(\frac{\partial \mathcal{S}_1}{\partial V} + \frac{1}{T} \frac{\partial f_{\text{NT}}}{\partial V} \right) \quad (\text{A89})$$

$$\frac{\partial f_{\text{NV}}}{\partial \sigma_\ell} = 0, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A90})$$

$$\frac{\partial f_{\text{NV}}}{\partial T} = 0 \quad (\text{A91})$$

$$\frac{\partial f_{\text{NV}}}{\partial \rho} = -\frac{f_{\text{NV}}}{\rho} \quad (\text{A92})$$

$$\frac{\partial f_{\text{NV}}}{\partial V} = -\frac{f_{\text{NV}}}{V} \quad (\text{A93})$$

In equations (A82) to (A84) and (A86) to (A88) the required partial derivatives of \mathcal{S}_1 , \mathcal{S}_2 , and c_p/R are given by equations (A63) to (A68) and

$$\frac{\partial \mathcal{S}_1}{\partial V} = M_w \sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial V} \quad (\text{A94})$$

$$\frac{\partial \mathcal{S}_2}{\partial V} = \sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} \frac{\partial f_i}{\partial V} \quad (\text{A95})$$

where $\partial f_i / \partial V$ depends on the independent variable (see eq. (A56)). The heat transfer terms depend on the heat transfer model and the independent variable (see eq. (A41)). For HT1 the required derivatives are as follows:

A. Ordinary Differential Equations and Jacobian Matrix Elements

$$\frac{\partial \mathcal{D}}{\partial \sigma_\ell} = 0, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A96})$$

$$\frac{\partial \dot{Q}'}{\partial T} = H_{T,1} + 2H_{T,2} T + 3H_{T,3} T^2 + 4H_{T,4} T^3 \quad (\text{A97})$$

$$\frac{\partial \mathcal{D}}{\partial p} = 0 \quad (\text{A98})$$

$$\frac{\partial \mathcal{D}}{\partial V} = \begin{cases} \frac{\mathcal{D}}{V}, & \xi = t \\ 0, & \xi = x \end{cases} \quad (\text{A99})$$

For HT2 and laminar flow $\partial \mathcal{D} / \partial \sigma_\ell$, $\partial \mathcal{D} / \partial p$, and $\partial \mathcal{D} / \partial V$ are, respectively, given by equations (A96), (A98), and (A99), and

$$\frac{\partial \dot{Q}'}{\partial T} = 2\sqrt{\pi A \mathcal{H}} \quad (\text{A100})$$

The expressions for $\partial \mathcal{D} / \partial \sigma_\ell$ and $\partial \dot{Q}' / \partial T$ remain unchanged in the turbulent case but

$$\frac{\partial \mathcal{D}}{\partial p} = 0.4 \frac{\mathcal{D}}{p} \quad (\text{A101})$$

$$\frac{\partial \mathcal{D}}{\partial V} = \begin{cases} 1.4 \frac{\mathcal{D}}{V}, & \xi = t \\ 0.4 \frac{\mathcal{D}}{V}, & \xi = x \end{cases} \quad (\text{A102})$$

Assigned area.—The partial derivatives of f_{NT} and f_{NR} are obtained from equations (A48) and (A49), but those of f_{NV} are computed by using the simplified expression, equation (A52), for this derivative:

$$\frac{\partial f_{\text{NT}}}{\partial \sigma_\ell} = \frac{T}{\mathcal{M}^2 - 1} \left[(\gamma - 1) \left(\mathcal{M}^2 \frac{\partial \mathcal{S}_1}{\partial \sigma_\ell} + \frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \frac{\partial \mathcal{M}^2}{\partial \sigma_\ell} \right) - \mathcal{M}^2 \left(\frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) \frac{\partial \gamma}{\partial \sigma_\ell} + (1 - \gamma \mathcal{M}^2) \left(\frac{\partial \mathcal{B}}{\partial \sigma_\ell} + \frac{\partial \mathcal{D}}{\partial \sigma_\ell} \right) \right],$$

$$\ell = 1, \dots, \text{NRS} \quad (\text{A103})$$

$$\frac{\partial f_{\text{NT}}}{\partial T} = \frac{T}{\mathcal{M}^2 - 1} \left\{ (\gamma - 1) \left(\mathcal{M}^2 \frac{\partial \mathcal{S}_1}{\partial T} + \frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \frac{\partial \mathcal{M}^2}{\partial T} \right) - \mathcal{M}^2 \left(\frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) \frac{\partial \gamma}{\partial T} + (1 - \mathcal{M}^2) \left[\frac{\gamma - 1}{\gamma} M_w \right. \right.$$

$$\left. \times \left(\frac{1}{T} \sum_{i=1}^{\text{NRS}} \frac{c_{p,i}}{R} f_i + \sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} \frac{\partial f_i}{\partial T} + \frac{1}{\dot{m} RT} \frac{V}{\chi} \frac{\partial \dot{Q}'}{\partial T} \right) + \frac{\mathcal{B} + \mathcal{D}}{\gamma(\gamma - 1)} \frac{\partial \gamma}{\partial T} \right\} \quad (\text{A104})$$

$$\frac{\partial f_{\text{NT}}}{\partial p} = \frac{T}{\mathcal{M}^2 - 1} \left[(\gamma - 1) \mathcal{M}^2 \frac{\partial \mathcal{S}_1}{\partial p} + (1 - \gamma \mathcal{M}^2) \left(\frac{\partial \mathcal{B}}{\partial p} + \frac{\partial \mathcal{D}}{\partial p} \right) \right] \quad (\text{A105})$$

$$\frac{\partial f_{NT}}{\partial V} = \frac{T}{\mathcal{M}^2 - 1} \left[(\gamma - 1) \left(\mathcal{M}^2 \frac{\partial \mathcal{S}_1}{\partial V} + \frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) \frac{\partial \mathcal{M}^2}{\partial V} \right. \\ \left. + (1 - \gamma \mathcal{M}^2) \left(\frac{\partial \mathcal{B}}{\partial V} + \frac{\partial \mathcal{D}}{\partial V} \right) \right] \quad (\text{A106})$$

$$\frac{\partial f_{NR}}{\partial \sigma_\ell} = \frac{\rho}{\mathcal{M}^2 - 1} \left(\frac{\partial \mathcal{S}_1}{\partial \sigma_\ell} - \frac{\partial \mathcal{B}}{\partial \sigma_\ell} - \frac{\partial \mathcal{D}}{\partial \sigma_\ell} + \frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) \frac{\partial \mathcal{M}^2}{\partial \sigma_\ell}, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A107})$$

$$\frac{\partial f_{NR}}{\partial T} = \frac{\rho}{\mathcal{M}^2 - 1} \left(\frac{\partial \mathcal{S}_1}{\partial T} - \frac{\partial \mathcal{B}}{\partial T} - \frac{\partial \mathcal{D}}{\partial T} + \frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) \frac{\partial \mathcal{M}^2}{\partial T} \quad (\text{A108})$$

$$\frac{\partial f_{NR}}{\partial \rho} = \frac{\rho}{\mathcal{M}^2 - 1} \left(\frac{\partial \mathcal{S}_1}{\partial \rho} - \frac{\partial \mathcal{B}}{\partial \rho} - \frac{\partial \mathcal{D}}{\partial \rho} \right) + \frac{f_{NR}}{\rho} \quad (\text{A109})$$

$$\frac{\partial f_{NR}}{\partial V} = \frac{\rho}{\mathcal{M}^2 - 1} \left(\frac{\partial \mathcal{S}_1}{\partial V} - \frac{\partial \mathcal{B}}{\partial V} - \frac{\partial \mathcal{D}}{\partial V} + \frac{1}{A} \frac{dA}{d\xi} - \mathcal{A} + \mathcal{D} \right) \frac{\partial \mathcal{M}^2}{\partial V} \quad (\text{A110})$$

$$\frac{\partial f_{NV}}{\partial \sigma_\ell} = -\frac{V}{\rho} \frac{\partial f_{NR}}{\partial \sigma_\ell}, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A111})$$

$$\frac{\partial f_{NV}}{\partial T} = -\frac{V}{\rho} \frac{\partial f_{NR}}{\partial T} \quad (\text{A112})$$

$$\frac{\partial f_{NV}}{\partial \rho} = \frac{V}{\rho} \left(\frac{f_{NR}}{\rho} - \frac{\partial f_{NR}}{\partial \rho} \right) \quad (\text{A113})$$

$$\frac{\partial f_{NV}}{\partial V} = \frac{f_{NV}}{V} - \frac{V}{\rho} \frac{\partial f_{NR}}{\partial V} \quad (\text{A114})$$

In equations (A103) to (A110) the partial derivatives of \mathcal{S}_1 are given by equations (A63) to (A65) and (A94), and those of \mathcal{M}^2 , γ , and \mathcal{B} are as follows:

$$\frac{\partial \mathcal{M}^2}{\partial \sigma_\ell} = \mathcal{M}^2 \left((\gamma - 1) \frac{c_{p,\ell}/R}{c_p/R} - \gamma M_w \right), \quad \ell = 1, \dots, \text{NRS} \quad (\text{A115})$$

$$\frac{\partial \mathcal{M}^2}{\partial T} = -\mathcal{M}^2 \left(\frac{1}{T} + \frac{1}{\gamma} \frac{\partial \gamma}{\partial T} \right) \quad (\text{A116})$$

$$\frac{\partial \mathcal{M}^2}{\partial V} = 2 \frac{\mathcal{M}^2}{V} \quad (\text{A117})$$

$$\frac{\partial \gamma}{\partial \sigma_\ell} = \gamma(\gamma - 1) \left(M_w - \frac{c_{p,\ell}/R}{c_p/R} \right), \quad \ell = 1, \dots, \text{NRS} \quad (\text{A118})$$

$$\frac{\partial \gamma}{\partial T} = -\frac{\gamma(\gamma - 1)}{c_p/R} \frac{\partial}{\partial T} \frac{c_p}{R} \quad (\text{A119})$$

A. Ordinary Differential Equations and Jacobian Matrix Elements

$$\frac{\partial \mathcal{B}}{\partial \sigma_\ell} = \frac{\gamma-1}{\gamma} M_w \frac{\partial \mathcal{S}_2}{\partial \sigma_\ell} - \mathcal{B} \frac{c_{p,\ell}/R}{c_p/R}, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A120})$$

$$\frac{\partial \mathcal{B}}{\partial T} = M_w \frac{\gamma-1}{\gamma} \left(\frac{1}{T} \sum_{i=1}^{\text{NRS}} \frac{c_{p,i}}{R} f_i + \sum_{i=1}^{\text{NRS}} \frac{h_i}{RT} f_i \right) + \mathcal{B} \left(\frac{1}{\gamma(\gamma-1)} \frac{\partial \gamma}{\partial T} - \frac{1}{T} \right) \quad (\text{A121})$$

$$\frac{\partial \mathcal{B}}{\partial \rho} = M_w \frac{\gamma-1}{\gamma} \frac{\partial \mathcal{S}_2}{\partial \rho} \quad (\text{A122})$$

$$\frac{\partial \mathcal{B}}{\partial V} = M_w \frac{\gamma-1}{\gamma} \frac{\partial \mathcal{S}_2}{\partial V} \quad (\text{A123})$$

In these equations the partial derivatives of \mathcal{S}_2 and c_p/R are given by equations (A66) to (A68) and (A95).

The partial derivatives $\{\partial \mathcal{D}/\partial \sigma_\ell\}$ and $\partial \mathcal{D}/\partial T$ in equations (A103), (A107), and (A108) are given by

$$\frac{\partial \mathcal{D}}{\partial \sigma_\ell} = -\mathcal{D} \frac{c_{p,\ell}/R}{c_p/R}, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A124})$$

$$\frac{\partial \mathcal{D}}{\partial T} = M_w \frac{\gamma-1}{\gamma} \frac{V}{\chi} \frac{1}{\dot{m}RT} \frac{\partial \dot{Q}'}{\partial T} + \frac{\mathcal{D}}{\gamma(\gamma-1)} \frac{\partial \gamma}{\partial T} - \frac{\mathcal{D}}{T} \quad (\text{A125})$$

The partial derivatives $\partial \dot{Q}'/\partial T$, $\partial \mathcal{D}/\partial \rho$, and $\partial \mathcal{D}/\partial V$ in equations (A104) to (A106), (A109), (A110), and (A125) depend on the heat transfer model. For HT1, $\partial \dot{Q}'/\partial T$ is given by equation (A97) and

$$\frac{\partial \mathcal{D}}{\partial \rho} = -\frac{\mathcal{D}}{\rho} \quad (\text{A126})$$

$$\frac{\partial \mathcal{D}}{\partial V} = \begin{cases} 0, & \xi = t \\ -\frac{\mathcal{D}}{V}, & \xi = x \end{cases} \quad (\text{A127})$$

For HT2, $\partial \dot{Q}'/\partial T$ is given by equation (A100) for both laminar and turbulent flows, but the other partial derivatives depend on the nature of the flow. Equations (A126) and (A127) apply in the case of laminar flow, but for turbulent flow

$$\frac{\partial \mathcal{D}}{\partial \rho} = -0.2 \frac{\mathcal{D}}{\rho} \quad (\text{A128})$$

$$\frac{\partial \mathcal{D}}{\partial V} = \begin{cases} 0.8 \frac{\mathcal{D}}{V}, & \xi = t \\ -0.2 \frac{\mathcal{D}}{V}, & \xi = x \end{cases} \quad (\text{A129})$$

Assigned area and temperature.—For this problem type the required partial derivatives are computed from equations (A51) and (A52) and are as follows:

$$\frac{\partial f_{\text{NR}}}{\partial \sigma_\ell} = -\frac{\rho}{\gamma \mathcal{M}^2 - 1} \left[\left(\frac{1}{A} \frac{dA}{d\xi} + \frac{f_{\text{NR}}}{\rho} \right) \frac{\partial \gamma \mathcal{M}^2}{\partial \sigma_\ell} - \frac{\partial \mathcal{S}_1}{\partial \sigma_\ell} \right], \quad \ell = 1, \dots, \text{NRS} \quad (\text{A130})$$

$$\frac{\partial f_{\text{NR}}}{\partial \rho} = \frac{\rho}{\gamma \mathcal{M}^2 - 1} \frac{\partial \mathcal{S}_1}{\partial \rho} + \frac{f_{\text{NR}}}{\rho} \quad (\text{A131})$$

$$\frac{\partial f_{\text{NR}}}{\partial V} = -\frac{\rho}{\gamma \mathcal{M}^2 - 1} \left[\left(\frac{1}{A} \frac{dA}{d\xi} + \frac{f_{\text{NR}}}{\rho} \right) \frac{\partial \gamma \mathcal{M}^2}{\partial V} - \frac{\partial \mathcal{S}_1}{\partial V} \right] \quad (\text{A132})$$

$$\frac{\partial f_{\text{NV}}}{\partial \sigma_\ell} = -\frac{V}{\rho} \frac{\partial f_{\text{NR}}}{\partial \sigma_\ell}, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A133})$$

$$\frac{\partial f_{\text{NV}}}{\partial \rho} = -\frac{V}{\gamma \mathcal{M}^2 - 1} \frac{\partial \mathcal{S}_1}{\partial \rho} \quad (\text{A134})$$

$$\frac{\partial f_{\text{NV}}}{\partial V} = \frac{f_{\text{NV}}}{V} - \frac{V}{\rho} \frac{\partial f_{\text{NR}}}{\partial V} \quad (\text{A135})$$

In equations (A130) to (A132) and (A134) the partial derivatives of \mathcal{S}_1 are given by equations (A63), (A65), and (A94), and

$$\frac{\partial \chi \mu^2}{\partial \sigma_\ell} = -M_w \chi \mu^2, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A136})$$

$$\frac{\partial \chi \mu^2}{\partial V} = 2 \frac{\chi \mu^2}{V} \quad (\text{A137})$$

Sensitivity Analysis

The numerical solution procedure described in chapter 4 of part I for the sensitivity coefficients requires the partial derivatives with respect to the rate coefficient parameters of the temporal derivatives of the dependent variables. These variables depend on the problem type—assigned pressure or constant density—and whether temperature has been assigned. The partial derivatives of the species formation rates are independent of the problem type and are derived from equation (A31), where $\chi = 1$ because $\xi = t$. The derivatives with respect to the rate coefficient parameters of reaction j , A_j , n_j , and E_j or c_j (see eqs. (8.3) and (8.4)), are as follows:

$$\frac{\partial f_i}{\partial A_j} = \frac{\omega_{ij}}{\rho A_j}, \quad i = 1, \dots, \text{NRS} \quad (\text{A138})$$

$$\frac{\partial f_i}{\partial n_j} = \frac{\omega_{ij} \ln T}{\rho}, \quad i = 1, \dots, \text{NRS} \quad (\text{A139})$$

$$\frac{\partial f_i}{\partial E_j} = -\frac{\omega_{ij}}{\rho RT}, \quad i = 1, \dots, \text{NRS} \quad (\text{A140})$$

$$\frac{\partial f_i}{\partial c_j} = \frac{\omega_{ij} T}{\rho}, \quad i = 1, \dots, \text{NRS} \quad (\text{A141})$$

In these equations ω_{ij} is given by equation (A33).

Assigned-Pressure Static Problem

For this problem type the additional partial derivatives required are those of f_{NT} , if T is not assigned, and f_{NR} and are obtained from equations (A37) and (A38):

$$\frac{\partial f_{\text{NT}}}{\partial \eta_j} = -\frac{T}{c_p/R} \frac{\partial \mathcal{S}_2}{\partial \eta_j} \quad (\text{A142})$$

$$\frac{\partial f_{\text{NR}}}{\partial \eta_j} = -\rho \left(\frac{\partial \mathcal{S}_1}{\partial \eta_j} - \frac{1}{T} \frac{\partial f_{\text{NT}}}{\partial \eta_j} \right) \quad (\text{A143})$$

In these equations \mathcal{S}_1 and \mathcal{S}_2 are given by equations (A34) and (A35), respectively, η_j is any one of A_j , n_j , E_j , and c_j , and

$$\frac{\partial \mathcal{S}_1}{\partial \eta_j} = M_w \sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial \eta_j} \quad (\text{A144})$$

$$\frac{\partial \mathcal{S}_2}{\partial \eta_j} = \sum_{i=1}^{\text{NRS}} \frac{k_i}{RT} \frac{\partial f_i}{\partial \eta_j} \quad (\text{A145})$$

where $\partial f_i / \partial \eta_j$ depends on η_j (see eqs. (A138) to (A141)). Finally, in equation (A143) $\partial f_{\text{NT}} / \partial \eta_j$ is either given by equation (A142) or zero for an assigned-temperature problem.

Constant-Density Static Problem

In this case the only additional partial derivatives required are those of f_{NT} , if T is not assigned, and are derived from equation (A40)

$$\frac{\partial f_{\text{NT}}}{\partial \eta_j} = \frac{T}{c_p/R - \sigma_m} \left(\frac{\partial \mathcal{S}_1}{\partial \eta_j} - \frac{\partial \mathcal{S}_2}{\partial \eta_j} \right) \quad (\text{A146})$$

where \mathcal{S}_1 and \mathcal{S}_2 are given by equations (A39) and (A35), respectively,

$$\frac{\partial \mathcal{S}_1}{\partial \eta_j} = \sum_{i=1}^{\text{NRS}} \frac{\partial f_i}{\partial \eta_j} \quad (\text{A147})$$

$\partial \mathcal{S}_2 / \partial \eta_j$ is given by equation (A145), and $\partial f_i / \partial \eta_j$ depends on η_j (see eqs. (A138) to (A141)).

Perfectly Stirred Reactor Problem

The numerical solution procedure for this problem requires the partial derivatives of the N ($= \text{NRS} + 1$) functionals $\{f_i\}$, equations (7.14a) and (7.14b) in chapter 7 of part I, with respect to the dependent variables. These variables are the reacting species mole numbers, σ_i ($i = 1, \dots, \text{NRS}$) and either temperature for an assigned-mass-flow-rate problem or mass

A. Ordinary Differential Equations and Jacobian Matrix Elements

flow rate for an assigned-temperature problem. In addition, for reasons discussed in chapter 7 (section 7.2) of part I, the partial derivatives with respect to the density must be included. The partial derivatives with respect to the species mole numbers are independent of the problem type. They are derived from equations (7.14a) and (7.14b) and are as follows:

$$\frac{\partial b_i}{\partial \sigma_\ell} = \sum_{j=1}^{NR} (v''_{ij} - v'_{ij}) \frac{\partial r_j}{\partial \sigma_\ell} - \rho M_w \sum_{j=1}^{NR} (v''_{ij} - v'_{ij}) \frac{\partial r_j}{\partial \rho} - \frac{\dot{m}}{\mathcal{V}} \delta_{i\ell},$$

$$i, \ell = 1, \dots, \text{NRS} \quad (\text{A148})$$

$$\frac{\partial b_N}{\partial \sigma_\ell} = h_\ell, \quad \ell = 1, \dots, \text{NRS} \quad (\text{A149})$$

In equation (A148), $\partial r_j / \partial \sigma_\ell$ depends on the reaction type and whether species ℓ is a reactant or a product (see eqs. (A6), (A7), (A21), and (A22)); $\partial r_j / \partial \rho$ also depends on the reaction type (see eqs. (A9) and (A23)); \mathcal{V} is the reactor volume; and $\delta_{i\ell}$ is the Kronecker symbol:

$$\delta_{i\ell} = \begin{cases} 0, & i \neq \ell \\ 1, & i = \ell \end{cases} \quad (\text{A150})$$

Assigned-Mass-Flow-Rate Problem

For this problem type $\{\partial b_i / \partial T\}$ and $\partial b_N / \partial T$ must also be computed. They are given by

$$\frac{\partial b_i}{\partial T} = \sum_{j=1}^{NR} (v''_{ij} - v'_{ij}) \frac{\partial r_j}{\partial T} - \frac{\rho}{T} \sum_{j=1}^{NR} (v''_{ij} - v'_{ij}) \frac{\partial r_j}{\partial \rho},$$

$$i = 1, \dots, \text{NRS} \quad (\text{A151})$$

$$\frac{\partial b_N}{\partial T} = \sum_{j=1}^{NS} \sigma_j c_{p,j} + \frac{1}{\dot{m}} \frac{\partial \dot{Q}}{\partial T} \quad (\text{A152})$$

In these equations $\partial r_j / \partial T$ is given by equation (A8), $\partial r_j / \partial \rho$ is given by either equation (A9) or (A23), and because HT1 is the only heat transfer model allowed for PSR problems, $\partial \dot{Q} / \partial T$ is given by equation (A70).

Assigned-Temperature Problem

In this case the additional partial derivatives required are those with respect to \dot{m} and are as follows:

$$\frac{\partial b_i}{\partial \dot{m}} = \frac{\sigma_i^* - \sigma_i}{\mathcal{V}}, \quad i = 1, \dots, \text{NRS} \quad (\text{A153})$$

$$\frac{\partial b_N}{\partial \dot{m}} = -\frac{\dot{Q}}{\dot{m}^2} \quad (\text{A154})$$

In equation (A153) the asterisk denotes the reactor inlet state.

Appendix B

Accessing System Clock

To measure the execution time required by LSENS, the user must provide a subprogram, REAL FUNCTION SECCPU, which computes the incremental central processing unit (CPU) time in seconds. This function calls the system clock and is given in figures B.1 to B.8 for several common computing systems. If the call to the system clock is not available or known, the dummy version of SECCPU given in figure B.9 may be used.

```
REAL FUNCTION SECCPU(TIME)
C
C   INTEGER ITIME
C
C   REAL TIME
C   REAL CPTIME
C
C   CALL CPUTIM(ITIME)
C
C   CPTIME = FLOAT(ITIME)*1.0E-03
C   SECCPU = CPTIME - TIME
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END
```

Figure B.1.—Listing of function SECCPU for IBM 370 computer.

```
REAL FUNCTION SECCPU(TIME)
C
C   CHARACTER*23 DATTIM
C
C   REAL TIME
C   REAL ECPU,ETCPU,ETIME
C
C   CALL DATETM(DATTIM,ECPU,ETIME,ETCPU)
C
C   SECCPU = ECPU - TIME
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END
```

Figure B.2.—Listing of function SECCPU for Amdahl 5870 computer and VM operating system.

```
REAL FUNCTION SECCPU(TIME)
C
C   REAL TIME
C   REAL CPTIME
C
C   CALL CLOCK(CPTIME,1,1)
C
C   CPTIME = CPTIME*1.0E-03
C   SECCPU = CPTIME - TIME
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END
```

Figure B.3.—Listing of function SECCPU for Amdahl 5870 computer, UTS operating system, and Fujitsu 77 compiler.

```

REAL FUNCTION SECCPU(TIME)
C
C   INTEGER ITIME
C
C   REAL TIME
C   REAL CPTIME
C
C   INCLUDE '($JPIDEF)'
C
C   CALL LIB$GETJPI(JPI$_CPUTIM,,,ITIME,,)
C
C   CPTIME = FLOAT(ITIME)*1.0E-02
C   SECCPU = CPTIME - TIME
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END

```

Figure B.4.—Listing of function SECCPU for VAX computers.

```

REAL FUNCTION SECCPU(TIME)
C
C   REAL TIME
C   REAL CPTIME
C   REAL DUM
C
C   DIMENSION CPTIME(2)
C
C   EXTERNAL f77lid
C
C   DUM = ETIME(CPTIME)
C
C   SECCPU = CPTIME(1) - TIME
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END

```

Figure B.7.—Listing of function SECCPU for Sun SPARCstation 1.

```

REAL FUNCTION SECCPU(TIME)
C
C   REAL TIME
C   REAL CPTIME
C
C   CALL SECOND(CPTIME)
C
C   SECCPU = CPTIME - TIME
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END

```

Figure B.5.—Listing of function SECCPU for CDC Cyber and Cray computers.

```

REAL FUNCTION SECCPU(TIME)
C
C   INTEGER ITIME
C
C   REAL TIME
C   REAL CPTIME
C
C   ITIME = MCLOCK()
C
C   CPTIME = FLOAT(ITIME)*1.0E-02
C   SECCPU = CPTIME - TIME
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END

```

Figure B.8.—Listing of function SECCPU for IBM RISC System/6000.

```

REAL FUNCTION SECCPU(TIME)
C
C   REAL TIME
C   REAL CPTIME
C   REAL DUM
C
C   DIMENSION CPTIME(2)
C
C   DUM = ETIME(CPTIME)
C
C   SECCPU = CPTIME(1) - TIME
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END

```

Figure B.6.—Listing of function SECCPU for Convex C220 minicomputer, Alliant FX/S computer, and IRIS workstations.

```

REAL FUNCTION SECCPU(TIME)
C
C   REAL TIME
C
C   SECCPU = 0.0
C
C   RETURN
C-----END OF FUNCTION SECCPU-----
END

```

Figure B.9.—Listing of dummy function SECCPU that can be used if call to system clock is not available or known.

Appendix C

Code Modifications

The present version of LSENS has built-in values for the following parameters: maximum number of elements (atoms, etc.), maximum number of species, maximum number of reactions, maximum number of third-body collisional reactions, maximum number of species with third-body collisional efficiencies different from unity for any one reaction, maximum number of tabular values for the assigned variable (pressure or area) and temperature, and maximum number of print stations. The necessary modifications to change the present values of these quantities are described in this appendix. Information is also given on changing logical unit numbers and some parameters that control the integration and sensitivity analysis computation.

All of the above quantities are set in the BLOCK DATA module. Table C.1 lists these variables, describes them, and gives their current values. All variables are integers, except SMEST, which is a double-precision variable, and TESTNO, which is real. Although any quantity may be reset by the user, we strongly recommend against changing the current values of JDMJAC, JDMOPT, and MESFLG. In particular, resetting MESFLG to zero will switch off all error messages from subroutine XERRWV (see table 9.1). The user will then be unaware of any difficulty experienced by the LSODE package, including illegal input. The reason for any premature execution termination may also be a mystery. The integer JDMOPT controls how the integration is to be continued after every solution output. The current value ($= 0$) indicates that the integration is to be continued normally. If JDMOPT is set to any nonzero value, LSODE will be reinitialized after every printout, which may result in significantly increased execution times and a reduction in the accuracy of both the kinetics and sensitivity analysis results. The integer JDMJAC controls the frequency of Jacobian matrix update while performing sensitivity analysis. The current value ($= 0$) means that the Jacobian matrix is to be updated on every step. Any nonzero value for JDMJAC will result in the matrix being updated only on those steps for which LSODE found this update necessary for the kinetics problem. Be-

cause the accuracy of the sensitivity coefficients is not tested, it may deteriorate if this option is used.

The maximum number of elements NLMAX that make up the chemical species can be changed by using table C.2. Table C.3 shows how to reset the maximum number of (reacting plus inert) species LSMAX that the reaction mechanism can involve. In addition to the modifications listed in this table, if LSMAX is increased, the lengths LRW of the real work array RWORK and LIW of the integer work array IWORK must be reset in the MAIN program. However, if LSMAX is reduced, the current dimensions for the two arrays need not be reset, although doing so will decrease storage requirements. These arrays are used for both integration of kinetics problems and sensitivity analysis. When sensitivity analysis is required, the length of RWORK will have to be reset if the maximum number of reactions LRMAX is increased.

The minimum length of RWORK (i.e., minimum LRW) depends on the number N of ordinary differential equations (ODE's) and the method flag MF (see eq. (9.1) and table 9.4). For sensitivity analysis computations the total number of sensitivity parameters NDMSSEN must also be taken into account. The minimum length of IWORK (i.e., minimum LIW) depends on N , the corrector iteration technique MITER (see table 9.5), and whether sensitivity analysis is needed.

For a kinetics-only problem (i.e., no sensitivity analysis) the required minimum LRW for each MF is given in table C.4. In this table MAXORD is the maximum method order to be used and has the default values of 12 for the Adams-Moulton method and 5 for the backward differentiation formula method. The maximum number of ODE's solved by the code is $LSMAX + 3$. The required minimum LIW is given in table C.5 for each MITER.

The LRW value used in the current version of LSENS is the minimum required for $MF = 21$, the default method. Thus, even if LSMAX is not increased, the user may have to reset LRW if a different MITER is selected. In addition, if

TABLE C.1.—USER-ADJUSTABLE VARIABLES IN BLOCK DATA MODULE ^a

Variable name	Description	Current value
JDMJAC	Switch to specify when Jacobian matrix is to be updated for sensitivity computations. JDMJAC = 0 causes this matrix to be computed at every time step. JDMJAC = 1 causes the Jacobian matrix to be computed only when it is updated by LSODE.	0
JDMOPT	Switch to specify mode of continuation of integration after every solution printout. JDMOPT = 0 instructs LSODE to continue integration normally. JDMOPT = 1 causes LSODE to restart solution after every printout.	0
LDAT	Logical unit number for temporary storage of input data for each individual case	7
LREAD	Logical unit number for problem data file	5
LRMAX	Maximum number of reactions	250
LSCR	Logical unit number for temporary storage of thermodynamic data included in problem data file	9
LSMAX	Maximum number of reacting-plus-inert species	50
LTHM	Logical unit number for standard thermodynamic data file	4
LTRAP	Logical unit number for transport properties data file	8
LUNIT	Logical unit number for (a) error messages from subroutine XERRWV for both kinetics and sensitivity analysis computations and (b) information regarding storage and computational work requirements for sensitivity analysis	6
LWRITE	Logical unit number for all output, except the quantities listed for LUNIT	6
MESFLG	Control flag for printing error messages from subroutine XERRWV. MESFLG = 0 tells the routine not to print any error message. MESFLG = 1 tells it to print all messages.	1
NBLANK	Identification number to denote no species in reactant or product location in a reaction (used internally)	52
NLMAX	Maximum number of elements	15
NPHOTO	Identification number to denote "reactant" HNU for photochemical reaction (used internally)	53
NPMAX	Maximum number of print stations	100
NTBMAX	Maximum number of tabular values for both the assigned variable (pressure or area) and temperature	100
NTHRD	Identification number to denote third-body species M (used internally)	51
SMEST	In normalizing sensitivity coefficients variable values less than SMEST in magnitude are set equal to SMEST.	10 ⁻³⁵
TBRMAX	Maximum number of third-body collisional reactions	35
TBSMAX	Maximum number of species with third-body collisional efficiencies different from 1.0 for any one reaction	10
TESTNO	Value (essentially minus infinity) to which all print stations and tabular values of assigned variable are initialized	-10 ³⁵

^aAll variables are integers, except SMEST, which is a double-precision variable, and TESTNO, which is real.

TABLE C.2.—MODIFICATIONS NEEDED TO CHANGE MAXIMUM NUMBER OF ELEMENTS (NLMAX)

1. Change NLMAX in BLOCK DATA to required value.	
2. Change variable dimensions in the following common blocks:	
Common block	Variable (dimension) ^a
ELMNTS	ELNAM(NLMAX)
MISC	ELNO(NLMAX)
MATX	GA(max(LSMAX+1, NLMAX+2), max(LSMAX+2, NLMAX+2)), GX(max(LSMAX+1, NLMAX+2))
SPECES	ELSP(NLMAX, LSMAX)

^aLSMAX is the maximum number of reacting-plus-inert species (see table C.3).

MF is equal to 14, 15, 24, or 25, the user must set IWORK(1) = ML and IWORK(2) = MU in the MAIN program before the call to subroutine SENDDM. Here ML and MU are, respectively, the lower and upper half-bandwidths of the banded Jacobian matrix. If MITER = 4 (i.e., if MF is 14 or 24), the user must replace subroutine PEDERV with a routine satisfying the requirements described in reference 22.

If sensitivity analysis is required, only a value of 21 may be specified for the method flag. The code assumes a maximum method order of five, even if a smaller value is set in namelist SOLVER (see section 11.3.9 and table 11.16). The

array RWORK will require an additional $N(6\text{NDMSEN} + N + 1) + 2$ words for the sensitivity analysis computations. In this case the maximum number of ODE's solved is $\text{LSMAX} + 2$. The minimum LRW needed for the kinetics-plus-sensitivity-analysis computation is given in table C.4. Here NDMSEN is the total number of initial conditions and rate coefficient parameters with respect to which sensitivity coefficients are to be computed and has a maximum value of $\text{LSMAX} + 2 + 3\text{LRMAX}$. The dimension of the array IWORK must be increased by at least $N + 20$, and the minimum LIW value that must be set is given in table C.5.

For either computation (i.e., kinetics only or kinetics plus sensitivity) the user must reset the LRW and LIW values given in the MAIN program. In addition, the dimensions specified in this routine for RWORK and IWORK must be changed to reflect the new LRW and LIW.

The modifications needed to change the maximum number of reactions LRMAX are given in table C.6. If LRMAX is increased and the sensitivity analysis option is used, LRW may have to be increased, as described previously (see table C.4).

The necessary actions to assign new values for the maximum number of third-body collisional reactions TBRMAX and the maximum number of species with third-body collisional efficiencies different from unity TBSMAX are listed in tables C.7 and C.8, respectively. Table C.9 shows how to change the maximum number of tabular values NTBMAX for the assigned variable and temperature. Finally, to reset the maximum number of print stations NPMAX, table C.10 may be consulted.

TABLE C.3.—MODIFICATIONS NEEDED TO CHANGE MAXIMUM NUMBER OF SPECIES (LSMAX)

<p>1. Changes in BLOCK DATA</p> <p>a. Reset LSMAX to desired maximum number of species.</p> <p>b. Reset following three variables to indicated values: NTHRD = LSMAX+1 NBLANK = LSMAX+2 NPHOTO = LSMAX+3</p>	
<p>2. Change variable dimensions in the following common blocks:</p>	
Common block	Variable (dimension) ^{a,b}
COND	SIGMA(LSMAX)
DFDPA	DFDPJ(LSMAX+2, 3LRMAX)
GHSC	GRT(LSMAX), HRT(LSMAX), SR(LSMAX), CPR(LSMAX), DCPR(LSMAX)
INERT1	DISNM(LSMAX)
MATX	GA(max(LSMAX+1, NLMAX+2), max(LSMAX+2, NLMAX+2)), GX(max(LSMAX+1, NLMAX+2))
PDDTRM	PDDSIG(LSMAX)
SAVRAT	FF(LSMAX+3)
SENNAM	SNames(2, LSMAX+3)
SENNOR	SIVNP(LSMAX+2), YINV(LSMAX+2)
SENPAR	SCIV(LSMAX+3, LSMAX+2), SCRPL(LSMAX+3, LRMAX, 3), IDPAR(2, LSMAX+2)
SENSOL	DYDYO(LSMAX+2, LSMAX+2), SENSE(LSMAX+3, LRMAX, 3)
SPECONC	C(LSMAX)
SPECES	EN(LSMAX), ENLN(LSMAX), DELN(LSMAX), ELSP(NLMAX, LSMAX)
SPEC1 ^c	DSPNM(LSMAX)
SPEC2	MW(LSMAX), W(LSMAX), STOICH(LSMAX, LRMAX), OMEGA(LSMAX, LRMAX)
TCOF	TC(LSMAX, 7, 2)
TRAN	VTC(4, LSMAX, 2), ICV(LSMAX), VCSP(LSMAX, 2)
<p>3. Change variable dimensions in DIMENSION statements in the following routines:</p>	
Routine	Variable (dimension) ^b
MAIN	Y0(LSMAX+3), ATOL(LSMAX+3)
GAUSS	COEFX(LSMAX+1)
HETRAN	DLVC(LSMAX)
KINP	C(LSMAX)
OUT2	PRC(LSMAX)
PEDERV	PBBSIG(LSMAX), PGSIG(LSMAX), PM2SIG(LSMAX), PS1SIG(LSMAX), PS2SIG(LSMAX)
RXNTAB	NRIMP(LSMAX, LRMAX), OMOUT(LSMAX, LRMAX)
WSOUT	EN(LSMAX), FMOUT(4, LSMAX)
WSR	ES(LSMAX), DLSIG(LSMAX), SIGLN(LSMAX), TZ(LSMAX+1)
SNSTAB	NRIMP(LSMAX+3, LRMAX, 3)
<p>4. Rewrite one EQUIVALENCE statement in subroutine KINP as follows:</p> <p>EQUIVALENCE (EFFM, DSPNM(LSMAX+1), (BLANK, DSPNM(LSMAX+2)), (HNU, DSPNM(LSMAX+3)))</p>	

^aNLMAX is the maximum number of elements (see table C.2).

^bLRMAX is the maximum number of reactions (see table C.6).

^cIn subroutines KINP and OUT1 the dimension of DSPNM in common block SPEC1 must be set equal to LSMAX+3 and not that given in the table.

TABLE C.4.—MINIMUM LENGTH REQUIRED BY REAL WORK ARRAY
RWORK (i.e., MINIMUM LRW)

Problem type	Method flag, MF	Minimum length required for RWORK, minimum LRW
Kinetics only	10, 20	$20 + N(\text{MAXORD} + 4)$
	11, 12, 21, 22	$22 + N(\text{MAXORD} + N + 4)$
	13, 23	$22 + N(\text{MAXORD} + 5)$
	14, 15, 24, 25 ^a	$22 + N(\text{MAXORD} + 2\text{ML} + \text{MU} + 5)$
Kinetics and sensitivity analysis ^b	21	$24 + N(6\text{NDMSEN} + 2N + 10)$

^aML and MU are, respectively, the lower and upper half-bandwidths of the banded Jacobian matrix. If MF = 14 or 24 is chosen, subroutine PEDERV must be replaced; otherwise, numerical instability may result.

^bOnly MF = 21 is allowed.

TABLE C.5.—MINIMUM LENGTH
REQUIRED BY INTEGER WORK
ARRAY IWORK (i.e.,
MINIMUM LIW)

Problem type	Iteration method, MITER ^a	Minimum length required for IWORK, minimum LIW
Kinetics only	0, 3	20
	1, 2, 4, 5	$20 + N$
Kinetics and sensitivity analysis ^b	1	$40 + 2N$

^aSee table 9.5.

^bOnly MITER = 1 is allowed.

TABLE C.6.—MODIFICATIONS NEEDED TO CHANGE MAXIMUM NUMBER OF REACTIONS (LRMAX)

1. Change LRMAX in BLOCK DATA to required value.	
2. Change variable dimensions in the following common blocks:	
Common block	Variable (dimension) ^a
DFDPA KOUT2 RATLOG REAC2 RRAT SENPAR SENSOL SPEC2 STCS ZERCON	DFDPI(LSMAX+2, 3LRMAX) DELH(LRMAX) DAFLOG(LRMAX) LSR(4, LRMAX), XX(LRMAX), DPX(2, LRMAX), RATE(LRMAX), BRATE(LRMAX), LRTYPE(LRMAX) A(LRMAX), N(LRMAX), EACT(LRMAX), NUM(LRMAX) SCRP(LSMAX+3, LRMAX, 3), DTRDRP(2, LRMAX, 3), NREAC(LRMAX) SENSE(LSMAX+3, LRMAX, 3) STOIC(LSMAX, LRMAX), OMEGA(LSMAX, LRMAX) NSTOIC(4, LRMAX), NSPRP(2, LRMAX) FBRATE(2, LRMAX)
3. Change variable dimensions in DIMENSION statements in the following subroutines:	
Subroutine	Variable (dimension) ^a
OUT2 RXNTAB SENSIN SNSTAB	PRX(LRMAX), EQUIL(LRMAX) NRIMP(LSMAX, LRMAX), OMOUT(LSMAX, LRMAX) RXNUM(LRMAX) NRIMP(LSMAX+3, LRMAX, 3)

^aLSMAX is the maximum number of reacting-plus-inert species (see table C.3).

TABLE C.7.—MODIFICATIONS NEEDED TO CHANGE MAXIMUM NUMBER OF THIRD-BODY REACTIONS (TBRMAX)

1. Change TBRMAX in BLOCK DATA to required value.	
2. Change variable dimensions in the following common blocks:	
Common block	Variable (dimension) ^a
REAC2 RRAT SPEC1	MM(TBRMAX), LR3RD(TBRMAX), NS3RD(TBRMAX), I3RD(TBSMAX, TBRMAX) M(TBSMAX, TBRMAX) TBSPNM(TBSMAX, TBRMAX)

^aTBSMAX is the maximum number of third-body species with efficiencies different from unity (see table C.8).

TABLE C.8.—MODIFICATIONS NEEDED TO CHANGE
 MAXIMUM NUMBER OF SPECIES WITH THIRD-
 BODY EFFICIENCIES DIFFERENT FROM
 UNITY (TBSMAX)

1. Change TBSMAX in BLOCK DATA to required value.	
2. Change variable dimensions in the following common blocks:	
Common block	Variable (dimension) ^a
REAC2	I3RD(TBSMAX, TBRMAX)
RRAT	M(TBSMAX, TBRMAX)
SPEC1	TBSPNM(TBSMAX, TBRMAX)

^aTBRMAX is the maximum number of third-body reactions (see table C.7).

TABLE C.9.—MODIFICATIONS NEEDED TO CHANGE MAXIMUM NUMBER OF
 TABULAR VALUES FOR ASSIGNED VARIABLE (PRESSURE OR AREA)
 AND TEMPERATURE (NTBMAX)

1. Change NTBMAX in BLOCK DATA to required value.		
2. Change variable dimensions in the following common blocks:		
Common block	Variable (dimension)	Alternative name (dimension), if any, for variable and subprograms where alternative name is used
CUBASV	CUBX(NTBMAX)	
	CUBY(NTBMAX)	
	CUBM(NTBMAX)	
CUBTMP	CUBXT(NTBMAX)	
	CUBYT(NTBMAX)	
	CUBMT(NTBMAX)	
SAVTMP	XTB(NTBMAX)	DUMST1(NTBMAX) in MAIN
	TTB(NTBMAX)	DUMST2(NTBMAX) in MAIN
	TMPTB(NTBMAX)	DUMST3(NTBMAX) in MAIN
SAVVAR	CXTB(NTBMAX)	DUMSV1(NTBMAX) in MAIN and OUT1
	CATB(NTBMAX)	DUMSV2(NTBMAX) in MAIN and OUT1
	CXTTB(NTBMAX)	DUMSV3(NTBMAX) in MAIN and OUT1
	CTMPTB(NTBMAX)	DUMSV4(NTBMAX) in MAIN and OUT1
XVSA2	XTB(NTBMAX)	XTBSAV(NTBMAX) in KINP
	ATB(NTBMAX)	ATBSAV(NTBMAX) in KINP
XVST2	XTTB(NTBMAX)	XTTBSV(NTBMAX) in TINP
	TMPTB(NTBMAX)	TTBSAV(NTBMAX) in TINP
3. Change variable dimensions in DIMENSION statements in the following subroutines:		
Subroutine	Variable (dimension)	
CUBS	X(NTBMAX), Y(NTBMAX), CUBX(NTBMAX), CUBY(NTBMAX), CUBM(NTBMAX)	
KINP	XTB(NTBMAX), TTB(NTBMAX), ATB(NTBMAX)	
SPLINE	X(NTBMAX), Y(NTBMAX), M(NTBMAX), SPS(NTBMAX), SPT(NTBMAX), U(NTBMAX), V(NTBMAX), TX(NTBMAX)	

TABLE C.10.—MODIFICATIONS NEEDED TO CHANGE MAXIMUM NUMBER OF PRINT STATIONS (NPMAX)

1. Change NPMAX in BLOCK DATA to desired value.		
2. Change variable dimensions in the following common blocks:		
Common block	Variable (dimension)	Alternative name (dimension) for variable and subprograms where alternative name is used
PRIN PRIN2 SAVTMP	PRINT(NPMAX) APRINT(NPMAX) PRINT(NPMAX) TPRINT(NPMAX)	CPRINT(NPMAX) in KINP and TINP DUMP21(NPMAX) in MAIN DUMP22(NPMAX) in MAIN DUMST6(NPMAX) in MAIN

References

1. Gardiner, W.C., Jr., ed.: *Combustion Chemistry*. Springer-Verlag, New York, 1984, pp. 1–19.
2. Basevich, V.Y.: *Chemical Kinetics in the Combustion Processes: A Detailed Kinetics Mechanism and Its Implementation*. *Prog. Energy Combust. Sci.*, vol. 13, no. 3, 1987, pp. 199–248.
3. Baulch, D.L.; and Drysdale, D.D.: An Evaluation of the Rate Data for the Reaction $\text{CO} + \text{OH} \rightarrow \text{CO}_2 + \text{H}$. *Combust. Flame*, vol. 23, 1974, pp. 215–225.
4. Pratt, G.L.: *Gas Kinetics*. Wiley, New York, 1969.
5. Forsythe, G.E.; Malcolm, M.A.; and Moler, C.B.: *Computer Methods for Mathematical Computations*. Prentice-Hall, Englewood Cliffs, NJ, 1977.
6. Stoer, J.; and Bulirsch, R.: *Introduction to Numerical Analysis*. Second Ed. Springer-Verlag, New York, 1992.
7. Bittker, D.A.; and Scullin, V.J.: GCKP84—General Chemical Kinetics Code for Gas-Phase Flow and Batch Processes Including Heat Transfer Effects. NASA TP–2320, 1984.
8. Holman, J.P.: *Heat Transfer*. Fifth ed., McGraw-Hill, New York, 1981.
9. Zeleznik, F.J.; and McBride, B.J.: *Modeling the Internal Combustion Engine*. NASA RP–1094, 1984.
10. Gordon, S.; and McBride, B.J.: *Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouguet Detonations*. NASA SP–273, 1976.
11. Gordon, S.; and McBride, B.J.: *Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications. Part I—Analysis*. NASA RP–1311, 1994.
12. McBride, B.J.; and Gordon, S.: *Computer Program for Calculating and Fitting Thermodynamic Functions*. NASA RP–1271, 1992.
13. Chase, M.W., et al.: *JANAF Thermochemical Tables*. Third ed., American Chemical Society, Washington, DC, 1986.
14. Callen, H.B.: *Thermodynamics and an Introduction to Thermostatistics*. Second ed., Wiley, New York, 1985.
15. Gordon, S.; McBride, B.J.; and Zeleznik, F.J.: *Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications. Supplement I: Transport Properties*. NASA TM–86885, 1984.
16. Maitland, G.C.; and Smith, E.B.: Critical Reassessment of Viscosities of 11 Common Gases. *J. Chem. Eng. Data*, vol. 17, no. 2, 1972, pp. 150–156.
17. Wilke, C.R.: A Viscosity Equation for Gas Mixtures. *J. Chem. Phys.*, vol. 18, no. 4, 1950, pp. 517–519.
18. Bromley, L.A.; and Wilke, C.R.: Viscosity Behavior of Gases. *Ind. Eng. Chem.*, vol. 43, no. 7, 1951, pp. 1641–1648.
19. Lindsay, A.L.; and Bromley, L.A.: Thermal Conductivity of Gas Mixtures. *Ind. Eng. Chem.*, vol. 42, no. 8, 1950, pp. 1508–1511.
20. Hindmarsh, A.C.: LSODE and LSODI, Two New Initial Value Ordinary Differential Equation Solvers. *ACM SIGNUM Newsletter*, vol. 15, no. 4, 1980, pp. 10–11.
21. Hindmarsh, A.C.: ODEPACK: A Systematized Collection of ODE Solvers. R.S. Stepleman et al., eds., *Scientific Computing*, North Holland Publishing Co., Amsterdam, 1983, pp. 55–64. (Also UCRL–88007, Lawrence Livermore Laboratory, Livermore, CA, 1982.)
22. Radhakrishnan, K.; and Hindmarsh, A.C.: Description and Use of LSODE, the Livermore Solver for Ordinary Differential Equations. NASA RP–1327, 1993. (Also UCRL–ID–113855, Lawrence Livermore Laboratory, Livermore, CA, 1993.)
23. Hindmarsh, A.C.; and Byrne, G.D.: EPISODE: An Effective Package for the Integration of Systems of Ordinary Differential Equations. Report UCID–30112, Rev. 1, Lawrence Livermore Laboratory, Livermore, CA, 1977.
24. Nordsieck, A.: On Numerical Integration of Ordinary Differential Equations. *Math. Comp.*, vol. 16, 1962, pp. 22–49.
25. Radhakrishnan, K.: Comparison of Numerical Techniques for Integration of Stiff Ordinary Differential Equations Arising in Combustion Chemistry. NASA TP–2372, 1984.
26. Radhakrishnan, K.: New Integration Techniques for Chemical Kinetic Rate Equations. I. Efficiency Comparison. *Combust. Sci. and Tech.*, vol. 46, no. 1–2, 1986, pp. 59–81.
27. Pratt, D.T.; and Radhakrishnan, K.: CREK1D: A Computer Code for Transient, Gas-Phase Combustion Kinetics. NASA TM–83806, 1984.
28. Pratt, D.T.; and Radhakrishnan, K.: Physical and Numerical Sources of Computational Inefficiency in Integration of Chemical Kinetic Rate Equations: Etiology, Treatment, and Prognosis. NASA TP–2590, 1986.
29. Radhakrishnan, K.; and Pratt, D.T.: Fast Algorithm for Calculating Chemical Kinetics in Turbulent Reactive Flow. *Combust. Sci. and Tech.*, vol. 58, no. 1–2, 1988, pp. 155–176.
30. Radhakrishnan, K.: *Combustion Kinetics and Sensitivity Analysis Computations. Numerical Approaches to Combustion Modeling*. E.S. Oran and J.P. Boris, eds., AIAA, Washington, DC, 1991, pp. 83–128.
31. Radhakrishnan, K.: Integrating Combustion Kinetics Rate Equations by Selective Use of Stiff and Nonstiff Methods. *AIAA J.*, vol. 25, no. 11, 1987, pp. 1449–1455. (Also AIAA Paper 85–0237 and NASA TM–86923, 1985.)
32. Dunker, A.M.: The Decoupled Direct Method for Calculating Sensitivity Coefficients in Chemical Kinetics. *J. Chem. Phys.*, vol. 81, no. 5, 1984, pp. 2385–2393.
33. Radhakrishnan, K.: Decoupled Direct Method for Sensitivity Analysis in Combustion Kinetics. *Advances in Computer Methods for Partial Differential Equations—VI*, R. Vichnevetsky and R.S. Stepleman, eds., IMACS, New Brunswick, NJ, 1987, pp. 479–486. (Also NASA CR–179636, 1987.)
34. Dunker, A.M.: A Computer Program for Calculating Sensitivity Coefficients in Chemical Kinetics and Other Stiff Problems by the

References

- Decoupled Direct Method. Report GMR-4831, Env. 192, General Motors Research Laboratories, Warren, MI, 1985.
35. Mirels, H.: Test Time in Low-Pressure Shock Tubes. *Phys. Fluids*, vol. 6, no. 9, 1963, pp. 1201-1214.
 36. Mirels, H.: Shock Tube Test Time Limitation due to Turbulent-Wall Boundary Layer. *AIAA J.*, vol. 2, no. 1, 1964, pp. 84-93.
 37. Mirels, H.: Flow Nonuniformity in Shock Tubes Operating at Maximum Test Times. *Phys. Fluids*, vol. 9, no. 10, 1966, pp. 1907-1912.
 38. Radhakrishnan, K.: A Critical Analysis of the Accuracy of Several Numerical Techniques for Chemical Kinetic Rate Equations. NASA TP-3315, 1993.
 39. Brabbs, T.A.; and Musiak, J.D.: Ignition Delay Measurements and Proposed Kinetic Model for Hydrogen-Oxygen. NASP CR-1030, 1988.
 40. Bittker, D.A.: Detailed Mechanism for Oxidation of Benzene. *Combust. Sci. and Tech.*, vol. 79, no. 1-2, 1991, pp. 49-72.
 41. Brabbs, T.A.; Lezberg, E.A.; Bittker, D.A.; and Robertson, T.F.: Hydrogen Oxidation Mechanism With Applications to (1) Chaperon Efficiency of Carbon Dioxide and (2) Vitiated Air Testing. NASA TM-100186, 1987.
 42. Westbrook, C.K.; and Dryer, F.L.: Chemical Kinetic Modeling of Hydrocarbon Combustion. *Prog. Energy Combust. Sci.*, vol. 10, no. 1, 1984, pp. 1-57.
 43. Warnatz, J.: Rate Coefficients in the C/H/O System. *Combustion Chemistry*, W.C. Gardiner, ed., Springer-Verlag, New York, 1984, pp. 197-360.

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 February 1994	3. REPORT TYPE AND DATES COVERED Reference Publication	
4. TITLE AND SUBTITLE LSENS, A General Chemical Kinetics and Sensitivity Analysis Code for Homogeneous Gas-Phase Reactions II. Code Description and Usage		5. FUNDING NUMBERS WU-505-62-52	
6. AUTHOR(S) Krishnan Radhakrishnan and David A. Bittker		8. PERFORMING ORGANIZATION REPORT NUMBER E-5140-2	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135-3191		10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA RP-1329	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, D.C. 20546-0001		11. SUPPLEMENTARY NOTES Krishnan Radhakrishnan, NYMA, Inc., Lewis Research Center Group, 2001 Aerospace Parkway, Brook Park, Ohio 44142; David A. Bittker, NASA Lewis Research Center. Responsible person, Edward J. Mularz, organization code 2650, (216) 433-5850.	
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Categories 25, 61, and 64		12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) LSENS, the Lewis General Chemical Kinetics and Sensitivity Analysis Code, has been developed for solving complex, homogeneous, gas-phase chemical kinetics problems and contains sensitivity analysis for a variety of problems, including nonisothermal situations. This report is part II of a series of three reference publications that describe LSENS, provide a detailed guide to its usage, and present many example problems. Part II describes the code, how to modify it, and its usage, including preparation of the problem data file required to execute LSENS. Code usage is illustrated by several example problems, which further explain preparation of the problem data file and show how to obtain desired accuracy in the computed results. LSENS is a flexible, convenient, accurate, and efficient solver for chemical reaction problems such as static system; steady, one-dimensional, inviscid flow; reaction behind incident shock wave, including boundary layer correction; and perfectly stirred (highly backmixed) reactor. In addition, the chemical equilibrium state can be computed for the following assigned states: temperature and pressure, enthalpy and pressure, temperature and volume, and internal energy and volume. For static problems the code computes the sensitivity coefficients of the dependent variables and their temporal derivatives with respect to the initial values of the dependent variables and/or the three rate coefficient parameters of the chemical reactions. Part I (NASA RP-1328) derives the governing equations and describes the numerical solution procedures for the types of problems that can be solved by LSENS. Part III (NASA RP-1330) explains the kinetics and kinetics-plus-sensitivity-analysis problems supplied with LSENS and presents sample results.			
14. SUBJECT TERMS Gas-phase chemical kinetics; Numerical solution of chemical kinetics ODE's; Sensitivity analysis; Chemical equilibrium; Incident shock; Perfectly stirred reactor		15. NUMBER OF PAGES 156	16. PRICE CODE A08
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT

