Rocketdyne Division
Rockwell International

LIQUID ROCKET COMBUSTION COMPUTER MODEL

WITH

DISTRIBUTED ENERGY RELEASE

DER COMPUTER PROGRAM DOCUMENTATION

AND USER'S GUIDE

VOLUME I

Prepared By

L. P. Combs

Prepared For

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Contract NAS7-746

Jet Propulsion Laboratory

Pasadena, California

15 December 1971

Revised 1 January 1974

FOREWORD

This computer program documentation was prepared by Rocketdyne, a Division of North American Rockwell Corporation, in accordance with and in partial fulfillment of Article I, Task IV and Task VIII (b) of Contract NAS7-746 with the National Aeronautics and Space Administration. The contract was administered for NASA by the Jet Propulsion Laboratory, Pasadena, California, whose Technical Manager was Dr. Raymond Kushida. The Rocketdyne Program Manager was Mr. T. A. Coultas and Dr. D. T. Campbell was the Project Manager.

This document accompanies the 15 December 1971 version of the DER computer program. Both the documentation and computer program are intended to supercede and replace their counterparts delivered to JPL in September 1970. Both that September 1970 version and this December 1971 version are based on the evaporation coefficient (k') approach to propellant spray combustion, which is limited in application to subcritical conditions. This document is not concerned with, nor does it supercede, a March 1971 version of DER which is based on a transient droplet heating and diffusion approach to spray combustion and which is intended for analyzing peri·rmance of systems whose chamber pressures are supercritical for one or both propellants.

REVISIONS

First Revision by W. D. Chadwick   January 1974

CONTENTS

REVISION PAGES

January 1974 - i, ii, iii, 1, 1a, 2, 2a, 6a, 6b, 7, 23, 25, 30, 30a, 31, 31a,
40, 41, 43, 46, 52, 53, 54, 55, 57, 58, 59, 98

NOMENCLATURE

A	=	area
a	=	local sound speed
$a, b, C_1 \text{---} C_6$	=	empirical spray coefficients, Eq. (2)
C_D	=	drag coefficient
C_F	=	thrust coefficient
$C_{k'}$	=	approximate evaporation coeffi nt
C_{ND}	=	nozzle discharge coefficient
c	=	mixture ratio
c^*	=	characteristic exhaust velocity
c_p	=	specific heat at constant pressure
D	=	droplet diameter
$\bar{D}$	=	mass median droplet diameter
E_m	=	Rupe mixing efficiency factor (Ref. 7)
F	=	drag force
g_c	=	gravitational coefficient
I_{SP}	=	specific impulse
ΔH_v	=	heat of vaporization
k, k_g	=	thermal conductivity
k_s'	=	droplet evaporation coefficient
M_w	=	molecular weight
m	=	droplet mass or spray size group mass
$\dot{m}$	=	rate of change of mass
N	=	droplet concentration (no/volume)
$\hat{N}$	=	number flowrate of droplets
N_{EL}	=	number of injection elements
Nu	=	Nusselt number
n_t	=	number of stream tubes
P, p	=	pressure

p^*	=	pressure corresponding to sonic flow
Re	=	Reynolds number
R_u	=	universal gas constant
R_R	=	nozzle throat radius ratio (curvature/opening)
R_T	=	nozzle throat opening radius
r	=	radial coordinate
s	=	stream tube path length
T	=	temperature
u	=	velocity
$\hat{W}$	=	propellant mass flux at a spatial mesh point
$\hat{w}$	=	propellant mass flux contribution from an injection element to a mesh point
$\dot{w}$	=	flowrate
$\hat{w}_{001}$	=	$w(x,y,z)$ for $x=y=0$, $z=1$
$\dot{W}_g$	=	stream tube gas flowrate
X	=	non dimensional distance from nozzle throat
x,y,z	=	rectangular coordinates (referenced to an injection element origin)
z	=	axial coordinate
z_0	=	initial plane for beginning spray combustion analysis

GREEK LETTERS

α	=	nozzle wall angle (from chamber axis)
α,β,γ	=	element rotational angles (see page 72)
γ	=	ratio of specific heats or adiabatic expansion coefficient
δ_G, δ_L	=	pseudo-impingement points for gaseous and liquid propellants for a gas/liquid injection element
ϵ_c	=	chamber contraction ratio
ϵ_e	=	nozzle expansion ratio
ϵ_{A_t}	=	decimal tolerance, convergence on throat area
η	=	efficiency factor
θ	=	angular coordinate
μ	=	viscosity
ρ	=	density
σ	=	surface tension

SUPERSCRIPTS

$(\bar{\ })$	=	average value or concerned with one-dimensional solution
n	=	concerned with the nth droplet size group
prime,( )'	=	value reduced by evaporation

SUBSCRIPTS

c	=	chamber
cr	=	critial point property
d	=	droplet
E	=	element
e	=	expansion section
f	=	fuel, spray fan
g	=	combustion gas
i	=	concerned with i^{th} stream tube, summation index
inj	=	injection
j	=	concerned with j^{th} propellant
ℓ	=	liquid
o	=	oxidizer
o,0	=	initial or stagnation value
s	=	surface or stream tube path
STC	=	stream tube combustion model
T	=	throat
TDK	=	two-dimensional kinetic model
v,vap	=	vapor or vaporization

1. INTRODUCTION

The Distributed Energy Release (DER) combustion analysis computer program described in this report has been developed to fill a need in, and to become a part of, the JANNAF system of related computer programs for performing bipropellant liquid rocket performance analyses. The DER program is concerned with formation, distribution, flow and combustion of liquid sprays and combustion product gases in conventional (cylindrical) rocket combustion chambers. The main spray flow direction must be essentially parallel to the chamber axis, but the injector surface can deviate from a simple flat plate normal to the chamber axis.

Although an earlier evaporation coefficient version of DER (Ref.1 and 2) also had a capability to analyze non-equilibrium, two-dimensional supersonic exhaust nozzle flow, this December 1971 version simply provides output data for subsequent input to another computer program (TDK) for performing that analysis. Major modifications included in the current version are:

1. Simplified injection element specification for inputting the LISP program block,

2. Replacement of the spray flux and mixture ratio distribution plotting subprograms with simpler, reduced-run-time routines,

3. Provision of a partial LISP capability for analyzing bipropellant injection involving one gaseous propellant,

4. Removal of a modified, long-form-option revision of the TDK program block from DER,

5. Provision of a subroutine for punching out nozzle throat-plane data in the form of a partial data deck for subsequent use in running an improved multi-stream tube TDK program.

6. Interface option to punch cards in LISP for 3DC and to read punched cards from 3DC in STC.

Regarding engine performance calculation, DER computes only characteristic velocity efficiency; all other performance parameters are presumed to result from the subsequent TDK program analysis.

A technical description of the DER computer program is given in Section 2 of this report. Beginning with Section 3, the remainder of this report is concerned with DER computer program usage. Section 3.1 shows the recommended

program overlay structure. Section 3.2 contains a brief statement about each of DER's subroutines, indicating the major functions that each performs. Section 3.3 is concerned with the provision of the rather considerable amount of input data needed to make DER computer program runs. Sample data code sheets show the correct sequence and formats for variable values and include notes concerning options to bypass the input of certain data. A separate list defines the variables and indicates their required dimensions. Section 3.4 contains some notes concerning program operation, e.g., core size requirements, output limits and execution times. Section 3.5 describes the computer program output. Literature cited in the body of the report is listed in Section 4 and correlation coefficient values used in the LISP subprogram are tabulated in Ref. 6. Finally, appended to the report, as a separate volume, are a listing of the computer program (Rev. Jan. 74), an example data deck listing and portions of typical output data.

2. COMPUTER MODEL ANALYSIS

Basically, two major subprogram blocks comprise the DER computer progr ·
These subprogram blocks can be executed individually or fully coup.   as a   ·ςle
program. The first is the Liquid Injection Spray Pattern (LISP) proςram. Lι.·
utilizes correlations for the propellant mass flux distributions for sinςle-
injection elements and, based on element design, location and orientatιon data,
combines outputs from all injection elements to calculate a complete mass flux
distribution in a plane normal to the combustor's axis and a distance z_0 down-
stream of the injector. Other outputs from this program block include mean pro-
pellant droplet sizes and velocity vectors, approximate calculations of the per-
centages of propellants gasified upstream of z_0, and data regarding the resultant
combustion gas flowing through the plane z_0. Cylindrical combustor coordinates
are used. The output from LISP provides the necessary description of the two-
phase flowfield for initializing the second major subprogram block, the Stream
Tube Combustion (STC) program. STC is designed to accept propellant flux data
at r, θ, z_0 mesh points from LISP, combine them into several axisymmetric stream
tubes and analyze spray gasification and combustion in all stream tubes
simultaneously as the flow progresses downstream of z_0. STC's basic method
involves finite difference solution of coupled ordinary differential equations
by marching along the chamber length.

An optional procedure is provided to supplement the DER analysis by executing
a 2-phase, 3-dimensional combustion program, 3DC[*], in between the LISP and STC
subprogram analyses. In the normal DER analysis, spray is considered to travel
in straight flow paths completely independent of gas flow paths throughout the
region modeled by the LISP subprogram and then instantly forced at the start of
the STC analysis to change direction to paths coincident with gas streamline
flow. The 3DC analysis physically models the mixing of spray and gas as the
original spray flow paths are redirected by the transverse drag force with the
gas flow.

[*]This is the 3C-CØMBUST (3DC model) reported on in Ref. 5, only with slight
modifications in order to accept punched card output from LISP and to provide
punched card output for STC.

At some distance upstream of the combustor exhaust nozzle's throat, the STC program block calls upon subprogram block TRANS to provide information about the transonic flow's spatial pressure distribution. The geometric relationships among LISP, STC and TRANS program block analyses are indicated in Fig. 4 (page 23). Application of the Two-Dimensional Kinetic (TDK) nozzle analysis computer program (Ref. 3) is also indicated there, although this is no longer an integral part of DER but must be accomplished by subsequent use of DER output data. It is because TRANS and TDK are formulated in two dimensions (r, z) that STC is designed to assemble asixymmetric stream tubes at plane z_0.

Descriptions of the analytical bases for the major subprogram blocks are given in this section.

2.1 INJECTION AND ATOMIZATION:
LISP SUBPROGRAM BLOCK

The LISP computer program block has been developed, applied and improved in several successive stages. It was initially structured as the first of several individual programs in a system of computer programs for analyzing injector/chamber compatibility for liquid/liquid bipropellant combinations (Ref. 4). That version was adapted to form a LISP subprogram block in the initial DER computer program (Ref. 1). Subsequently, the LISP input data requirements were simplified, subroutines for producing computer-plotted mass flux contour and profile graphs were simplified and the formulation was expanded to include coaxial jet injection of gaseous fuel/liquid oxidizer propellants (Ref. 5). Similarly, gas/liquid triplet injection was added during the current DER program effort.

The LISP computer program calculates spray mass fluxes at mesh points (r,θ,z_0) by straightforward summation of mass fluxes from all individual injector elements

$$\hat{W}(r,\theta,z_0) = \sum_{i=1}^{N_{EL}} \hat{w}_i(r,\theta,z_0) \tag{1}$$

The method can be used if: (1) the individual injector elements have predictable spray flux patterns which can be measured and correlated with the main direction of spray flow parallel to the chamber axis; (2) the individual spray patterns of the various elements are not destroyed between the element impingement points and the plane z_0 by collisions between neighboring fans; and (3) vaporization of injected spray mass between the element impingement and plane z_0 can be estimated. A detailed accounting is not made for any back-flow of spray toward the injector; overall axial flow continuity is enforced at the LISP collection plane.

The necessary correlation for $\hat{w}_i(r,\theta,z_0)$ for use in Eq. (1) is based upon the shape of single element spray flux distributions determined from cold-flow spray measurements. Flux patterns were fitted to the generalized expression:

$$\hat{w}_i(x,y,z) = \frac{\hat{w}_{001}}{z^2} \left\{ \left[1 + c_1\left(\frac{y}{z}\right) + c_2\left(\frac{y}{z}\right)^2 \right] + \left[c_3\left(\frac{x}{z}\right) \right. \right.$$
$$\left. \left. + c_4\left(\frac{x}{z}\right)^2 \right] \left[1 + c_5\left(\frac{y}{z}\right) + c_6\left(\frac{y}{z}\right)^2 \right] \right\} e^{-a\left(\frac{x}{z}\right)^2 - b\left(\frac{y}{z}\right)^2} \qquad (2)$$

which is applied separately to each propellant from an element. The (x,y,z) coodinate system in Eq. (2) is referenced to the element's impingement point from which its spray is presumed to emanate. The empirical coefficients a, b, w_{001} and c_1 through c_6 are functions of such parameters as element type (doublet, triplet, etc.) impingement angle, orifice diameter, impinging stream momenta, orifice length, and manifold effects. The coefficients are obtained by correlating data from injector cold-flow simulation tests, in which immiscible propellant simulants are collected in a plane beneath the injector and measured. The form of Eq. (2) was chosen because it satisfies continuity, predicts the observed inverse square relationship between mass flux and distance from the impingement point, and because closed form integrals of Eq. (2) and its x and y moments over the x, y plane allow straightforward evaluation of the empirical coefficients, using experimental cold-flow data. A detailed discussion of the coefficient evaluation procedure, including an actual example computation, is given in Ref. 6. That document is primarily a catalog of the correlation coefficients which are currently in the LISP library of coefficient subroutines.

LISP transforms the x,y,z coordinate systems of the individual injection elements to the r, θ,z coordinates of the combustor and then sums the propellant mass fluxes from the individual elements to each r,θ mesh point in plane z_0. Advantage can be taken of injector design symmetry to define a mesh system which occupies only a fraction of the chamber. This reduces the amount of input data and the LISP program execution times, but care must be exercised in defining the injection elements, to ensure that the mesh system receives the correct geometric proportion of the total propellant injection rates.

LISP assumes that droplets travel as rays from the element origins (e.g., impingement points) to the mesh points and that injection kinetic energy is

conserved. The velocity of a given drop is therefore given by its injection velocity and the direction cosine of its path from the injector impingement point to the mesh point. When expressed in a common reference frame (the chamber geometry) the mass-weighted drop velocity vectors from individual elements can be summed at a mesh point to give mean velocity vectors u_{dz}, u_{dr}, and $u_{d\theta}$.

Propellant fluxes from neighboring elements are considered to pass through one another without interference. For many rocket designs, with moderately small elements or moderately widely-spaced elements, this assumption appears to be reasonably valid. When a spray was examined from a non-continuum, kinetic theory viewpoint, the mean-free-path of a drop was estimated to be about 1/2 to 1-inch for a typical rocket injection condition of about 1 lb/sec-in^2. Operation at higher injection densities (higher chamber pressures and/or lower contraction ratios) would tend to invalidate this assumption, but no limit conditions have been established.

Gaseous propellant injection is currently treated in LISP as if it were a low density, non-continuum spray. Modifications to the LISP logic were thus minimized, but this treatment obviously limits the program to analyzing single-element injectors or multiple-element injectors whose elements are far enough apart that their gas flows do not interact strongly. With closely-spaced elements, the gas distribution can be rather poorly predicted; a symptom of this condition may be the calculation of axial gas velocities between elements that are comparable with those downstream of the elements.

A very essential part of the combustion field initialization performed by LISP is the assignment of propellant droplet size distributions. In the DER computer program, LISP computations are concerned only with a mass median diameter ($\bar{D}$) for each propellant's spray. Later, during STC program block's initialization of stream tubes, the sprays are distributed into a discrete number of droplet size groups. The magnitudes of the $\bar{D}$'s frequently have a direct, strong influence on the propellant combustion efficiency computed by DER. Thus, an effort should be made to ensure that the most realistic values of $\bar{D}$ are supplied to LISP.

For some injector types, mean droplet sizes are calculated within LISP if they are not supplied by the program user. Here, empirical equations are used which

relate $\bar{D}$ to injection element hole sizes and injection velocities. The basic formulae were derived from cold-flow studies of molten wax jets; empirical adjustments are included to account approximately for differences in liquid and gas properties between those experimental and combustor conditions. A discussion of LISP's $\bar{D}$ formulae and guidelines for selecting $\bar{D}$'s to input to LISP are given in Ref. 6.

Partial propellant evaporation upstream of z_0 is calculated by a simplified, integrated evaporation expression

$$\hat{w}'(r,\theta,z_0) = \hat{w}(r,\theta,z_0)\left[1 - \frac{C_{k'}\,\Delta z}{\bar{D}^2 u_d}\right]^{3/2} \qquad (3)$$

where $\hat{w}'$ is the liquid spray flux actually arriving at the point (r, θ, z_0). The coefficient $C_{k'}$ is related to the evaporation coefficient k' used in the subsequent spray combustion analysis. However, because the liquid sprays are not fully atomized over the entire Δz distance, values of $C_{k'}$, including a convective Nusselt number, are usually assumed to be only about 1/5 to 1/4 of the stagnant values of k'. The propellant vapors said to be generated by this calculation are summed over all mesh points to yield a single overall vapor flow rate for each propellant. Use of such a simplified evaporation expression is, to some extent, justified by the relatively small percentage of evaporation in the spray formation zone.

To ensure continuity of both propellants, the LISP program performs summations over all mesh points of local propellant flow rates and then scales each propellant's mesh point flows and fluxes by the ratio of its total injected flow to its mesh point sum. This normalization process thus corrects the total flows for computational simplifications and errors.

Input to the LISP computer program consists of the number, location orientation, size, geometry, and type of injector elements, together with the geometry of combustion zone mesh network and general data concerning the propellant densities and injector pressure drops. Up to 60 injector elements can be considered. As many as 400 combustion zone mesh points can be prescribed.

6

2.1A 2-PHASE, 3-DIMENSIONAL SUBSONIC FLOW: 3DC INTERFACE WITH DER
       COMPUTER PROGRAM

Two distinct models for analyzing flow in the combustion chamber exist in the
DER computer programs: one is the LISP model which analyzes a region starting
at the injector face and the other is the STC model which analyzes the region
downstream of the LISP region. These models are integrated in DER to be
executed sequentially in a single computer run. However, by executing the two
DER models separ. ely, the overall analysis can be expanded to include the 3DC
computer program for modeling 2-phase, 3-dimensional flow in an intermediate
region. Interfacing of the DER models with 3DC is not fully automated. To
assist the user in running these models in sequence, the bul'; of the flow data
which must pass between them is transmitted via computer-generated punched data
cards: LISP generates data cards for 3DC, 3DC accepts these cards and punches
cards for STC, and STC reads the 3DC-generated cards.

Inclusion of 3DC imposes a severe requirement on the mesh size used in LISP.
The sector which is representative of the cross section must contain seven
angular mesh positions and 15 or less radial mesh positions. LISP can accomo-
date a mesh size up to 20 by 20 when 3DC is not used, but must have the same size
mesh system as 3DC when the latter is used. For some injector designs, the 3DC
mesh size may be too course to obtain satisfactory , rformance predictions.

The length of the LISP region is especially critical in determining the propel-
lant mixing efficiency when the 3DC analysis is bypassed, because the length
controls the amount of overlap between neighboring spray fans. Ideally, this
length is not critical when it is kept short and the spray spreading is handled
by the 3DC analysis. However, due to practical considerations of modeling
spray flow with a finite element representation, the LISP region length may or
may not be very critical, depending on the particular spray-spreading character-
istics. In the LISP analysis, spray at any mesh point is moving in as many
directions as there are injection elements. In interfacing with 3DC, a single,
mean spray velocity vector replaces the multiple spreading spray velocity

vectors at each mesh point. Therefore, the modeling of spray spreading using
3DC with LISP loses some validity when propellant is injected from more than
one injection element. In general, user judgment is required to select an
appropriate length for the LISP region, whether 3DC is used or not; however,
in some cases, the judgmental selection is not critical.

2.2 SUBSONIC FLOW AND COMBUSTION:

STC SUBPROGRAM BLOCK

Selected data computed by LISP are transferred (via scratch data unit) to the STC subprogram block. There, by one of three alternate methods, several mesh points' propellant fluxes and flow areas are combined to form one of the stream tube flows to be analyzed by STC.[*] Model solutions for spray gasification and combustion are obtained numerically for several systems (one for each stream tube) of simultaneous ordinary differential and algebraic equations by starting from known conditions at the LISP collection plane and marching downstream in small axial steps.

Input to the STC computer program consists of chamber wall profile, propellant properties, combustion gas properties and either (1) initial-plane gaseous flowrate and mixture ratio and spray flowrates, velocities, and droplet diameters for all spray size-groups entering each stream tube or (2) data from LISP from which these variables can be calculated. Up to 40 stream tubes can be initialized with as many as 12 spray size-groups (fuel and oxidizer combined).

[*]When 3DC is used, these data are transferred via punched cards generated by 3DC.

2.2.1 Stream Tube Initialization from LISP Data

Data transferred to STC from LISP are: propellant mass fluxes in their injected states, mean droplet velocities and mass median diameters at each mesh point; mesh point coordinates; and total initial plane flow and how much of it is gasified for each propellant. If there is no spray present for one of the propellants, its gaseous mass flux at each mesh point is retained, otherwise, simple bulk propellant vapor flows are transferred. In either case, the gas mixture ratio is considered to be uniform (constant) across the r,θ,z_0 plane. Axisymmetric stream tube flows may be initialized from these data by means of the following options:

(1) All mesh points along each circle (r = constant) of LISP's mesh points are combined into one stream tube. Gasified propellants are handled in one of two ways, depending upon whether one propellant is completely gasified or not:

a) If there is no spray of one propellant, the bipropellant gas flux distributions to the mesh points computed by LISP may either be retained or be averaged out to form a uniform bulk gas flux before the mesh points are combined into stream tubes.

b) If there is spray of both propellants, the gas mass fluxes are initially approximated as being uniform:

$$\dot{w}_{g_{ij}} = \left(\dot{W}_{gf} + \dot{W}_{go}\right) \frac{A_{ij}}{\sum\limits_{i,j} A_{ij}} \tag{4}$$

Then the gas mixture ratio at each mesh point is said to be equal to the spray mixture ratio there:

$$c_{ij} = \frac{\dot{W}_{o_{i,j}}}{\dot{W}_{f_{ij}}} \tag{5}$$

In general, however, these two assumptions will not be compatible with conservation of propellant species flow rates, e.g.:

$$\dot{W}_{gf} \neq \sum\limits_{ij} \frac{\dot{w}_{g_{ij}}}{1 + c_{ij}} \tag{6}$$

8

Therefore, the fuel and oxidizer contributions to each mesh point's gas flow are scaled separately to preserve species continuity:

$$\dot{w}_{gf_{ij}} = \frac{\dot{w}_{g_{ij}}}{1 + c_{ij}} \left[\frac{\dot{W}_{gF}}{\sum\limits_{ij} \frac{\dot{w}_{g_{ij}}}{1 + c_{ij}}} \right] \qquad (7)$$

$$\dot{w}_{go_{ij}} = \frac{c_{ij}\dot{w}_{g_{ij}}}{1 + c_{ij}} \left[\frac{\dot{W}_{gO}}{\sum\limits_{ij} \frac{c_{ij}\dot{w}_{g_{ij}}}{1 + c_{ij}}} \right] \qquad (8)$$

These definitions complete the specification of propellant flows at each mesh point.

This initialization method may be appropriate for injectors that form essentially axisymmetric flows. When applied to injectors which produce angular gradients in local propellant mixture ratio, however, it can effect substantial mixture ratio averaging and result in overcalculation of combustion efficiency.

(2) Mesh points are combined into stream tubes on the basis of position and local mesh point mixture ratio. Distribution of the gaseous propellant flows is the same as described in (1a) or (1b), above. Following distribution of the gases amont the mesh points, a wall boundary layer stream tube is established by combining all the mesh points at the wall. If that stream tube does not contain more than a specified percentage of the total flow, the next inward circle of mesh points will also be combined into it, etc., until it does. Then the remaining LISP circles of mesh point are divided into a specified few (2 to 4, perhaps) circular or annular zones having roughly equal propellant flow rates, as illustrated in Fig. 1a.

Within each of these zones, the mesh point flows are accumulated into stream tubes according to their total propellant mixture ratios, rather than to their positions. The number of stream tubes per zone is specified and they are assigned roughly equal propellant flow rates. The lowest mixture ratio mesh points are combined into the first stream tube until its fraction of the zonal flow rate is reached, the next lowest mixture ratio mesh points are assigned to the second stream tube, etc., as illustrated in Fig. 1b.

9
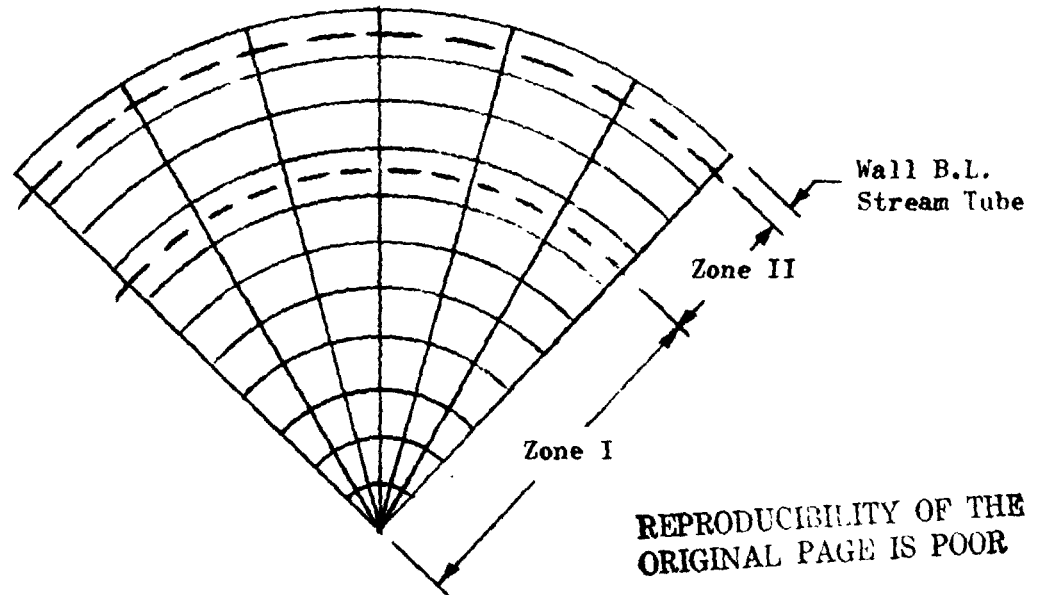
a. Geometric Assignment of Mesh Points to a Wall Boundary
   Layer Flow and to Two Approximately-Equal Flow Rate Zones.
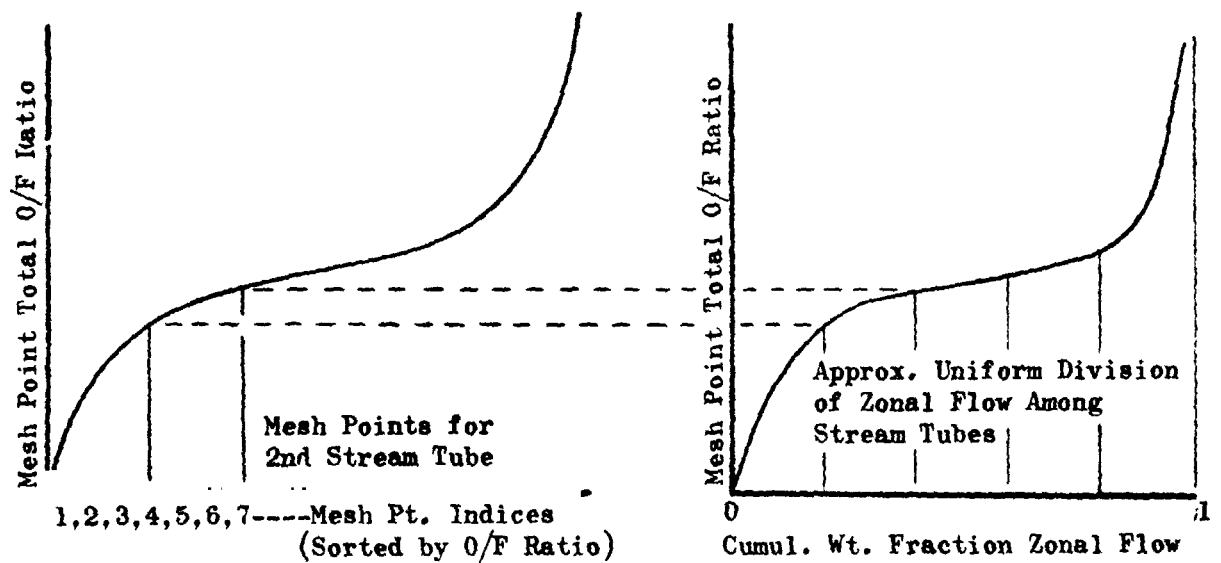
b. Mesh Point Assignments to Stream Tubes within a Zone

Figure 1. Illustration of Assignment of Mesh Point Flows
           to Stream Tubes of Like Mixture Ratio

Finally, the resultant stream tubes are arbitrarily assigned radial positions within their respective zones, with the fuel-rich stream tubes lying inside of the oxidizer-rich ones.

This method preserves the angular averaging objected to before only at the wall and is accepted for a fraction of the flow in order to get a wall-bounding stream tube that is characteristic of the mean wall mixture ratio. For the remainder of the flow, the non-physical combining of mesh points on the basis of mixture ratio has been found to effect only modest changes in calculated mixing efficiencies from those based on the full LISP distributions.

2.2.2 System of Equations

The system of equations for the $i\underline{\text{th}}$ stream tube is:

Gas Phase
Continuity:

$$\frac{d}{ds}\left(\rho_i u_i A_{s_i}\right) = A_{s_i} \sum_{n,j} \left(\dot{m}_j^n\right)_i \tag{9}$$

Momentum:

$$\frac{d}{ds}\left(\rho_i u_i^2 A_{s_i}\right) = A_{s_i}\left[-g_c\left(\frac{dp}{ds} + \sum_{j,n} \left(F_j^n\right)_i\right)\right.$$

$$\left. + \sum_{j,n} \left(\dot{m}_j^n\right)_i \left(u_{dj}^n\right)_i\right] \tag{10}$$

Adiabatic Energy Equation:

$$T_i = T_{oi}\left[1 - \frac{\gamma_i - 1}{2}\left(\frac{u_i}{a_{oi}}\right)^2\right] \tag{11}$$

where

$$T_{oi} = T_o(c_i), \quad \gamma_i = \gamma(c_i), \quad \text{and} \quad M_{wi} = M_w(c_i)$$

10

are tabulated* and

$$a_{oi} = \left[\frac{\gamma_i R_u T_{oi} g_c}{M_{wi}}\right]^{1/2} \tag{12}$$

This corresponds to frozen expansion to local conditions from stagnation equilibrium. **Frozen values of the specific heat ratio, γ, are used.**

The local stream tube gas mixture ratio is obtained simply by integrating the evaporation rates to get gasified flowrates:

$$\dot{w}_{ji}(z) = \dot{w}_{ji}(z_0) + \int_{z'=z_0}^{z} A_i \sum_n (\dot{m}_j^n)_i dz' \tag{13}$$

Mixture Ratio:

$$c_i = \frac{\dot{w}_{oxid,i}(z)}{\dot{w}_{fuel,i}(z)} \tag{14}$$

State:

$$\rho_i = \frac{p M_{wi}}{R_u T_i} \tag{15}$$

Spray Phase (n^{th} droplet size group of jth propellant)

Mass Continuity:

$$\frac{d}{ds}\left[(\rho_{dj}^n)_i (u_{dj}^n)_i A_{s_i}\right] = -\Lambda_{s_i}(\dot{m}_j^n)_i \tag{16}$$

Drop Number Continuity:

$$\frac{d}{ds}\left[(N_{dj}^n)_i (u_{dj}^n)_i A_{s_i}\right] = 0 \tag{17}$$

---

*These combustion gas properties are obtained from separate calculation of equilibrium chamber conditions for several mixture ratios and the nominal chamber pressure for a particular case being analyzed. They are also relatively weak functions of chamber pressure, but this dependence is neglected.

11

or, equivalently,

$$\left(\dot{N}_{dj}{}^n\right)_i = \left(N_{dj}{}^n\right)_i \left(u_{dj}{}^n\right)_i A_{s_i} = \text{constant*} \tag{18}$$

Momentum:

$$\frac{d}{ds}\left[\left(\rho_{dj}{}^n\right)_i \left(u_{dj}{}^n\right)_i^2 A_{s_i}\right] = A_{s_i}\left[g_c \left(F_j{}^n\right)_i - \left(\dot{m}_j{}^n\right)_i \left(u_{dj}{}^n\right)_i\right] \tag{19}$$

The independent variable in these one-dimensional flow equations is the stream tube path length or flow direction, s_i. This variable is related to the stream tube's cylindrical (r,z) coordinates through the differential expression

$$ds_i = \left(dz_i{}^2 + d\bar{r}_i{}^2\right)^{1/2} \tag{20}$$

where $\bar{r}_i$ is the stream tube's mean radius. For numerical stability in the STC solution, however, approximations are used that $ds_i = dz$ where the chamber wall is parallel to the axis and that

$$ds_i = dz \left\{\frac{\left[\bar{r}_i{}^2 + (z_I - z)^2\right]^{1/2}}{|z_I - z|}\right\} \tag{21}$$

in the nozzle. The basis for Eq. 21 may be seen by examining Fig. 2.

In this formulation, A_{s_i} appears as a dependent variable for which a solution is to be found. The gas phase equations are constrained, however, in terms of z-plane area:

$$\sum_i A_{z_i}(z) = A_c(z)$$

---

*This equation states that the number flowrate of droplets in each propellant spray group size is constant. However, if only a single size group (i.e., a monodisperse spray) is specified for a propellant, the program holds the droplet diameter constant for that size group and its number flowrate is decreased as spray vaporization proceeds.
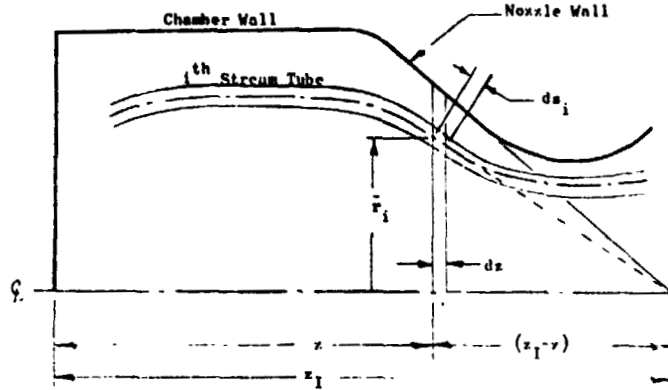
Figure 2. Schematic Illustration of Assur ?cal Conical Convergence of Stream 7

Therefore, the foregoing equations were modified for the ...r·uter program to permit direct solution for A_{z_i} by substituting:

$$A_{s_i} = A_{z_i} \frac{dz}{ds}$$

The sets of gas and liquid phase equations are coupled through mass and momentum exchange between phases. For droplet gasification, the simple evaporation coefficient model is utilized:

$$\dot{m}_j^{\ n} = N_j^{\ n} \left(\frac{\pi}{8}\right) \rho_{\ell_j}^{\ n} D_j^{\ n} Nu_j^{\ n} k'_{s_j}^{\ n} \qquad (22)$$

where the evaporation coefficient is

$$k'_{s_j}^{\ n} = \frac{8}{\rho_{\ell_j}^{\ n}} \int_{T_d}^{T} \frac{k_g}{\Delta H_v + \int_{T_d}^{T} c_{p_v} dT} dT \qquad (23)$$

and

$$Nu_j^{\ n} = 2 + 0.53\, Re_j^{\ n} \qquad (24)$$

Drag forces on spray droplets are expressed by

$$F_{j_i}^{\ n} = \frac{\pi}{8 g_c} N_{d_j}^{\ n} \rho_i D_j^{\ n} C_{D_j}^{\ n} \left(u_i - u_{d_j}^{\ n}\right) \left| u_i - u_{d_j}^{\ n}\right| \qquad (25)$$

13

with the drag coefficient specified as

$$C_{D_j}^{n} = 24 \left(Re_j^{n}\right)^{-0.84} ; \quad Re_j^{n} \leq 80$$

$$= 0.271 \left(Re_j^{n}\right)^{0.217} ; \quad Re_j^{n} > 80 \qquad (26)$$

2.2.2.1 Performance Parameters

Two separate parameters are calculated which are indicative of the overall degree of propellant mixing. These are calculated once in LISP, based on the flowrates associated with the LISP mesh points, and once in STC, based on the initial flowrates supplied to the stream tubes. One parameter is E_m, a mixing efficiency factor due to Rupe (Ref. 7) which expresses a mass-weighted average approach of local oxidizer mass fractions to the overall injected mass fraction:

$$E_m = 100 \left[1 - \sum_{i=1}^{n} \frac{\dot{w}_i (R-r_i)}{\hat{W}R} - \sum_{i=1}^{\bar{n}} \frac{\dot{w}_i (R-\bar{r}_i)}{\hat{W}(R-1)} \right] \qquad (27)$$

where:

n = number of samples with $R > r$

$\bar{n}$ = number of samples with $R < r$

$\dot{w}$ = local propellant flowrate, lb_m/sec

$\hat{W}$ = total propellant flowrate, lb_m/sec

$r, \bar{r}$ = local oxidizer mass fractions, $\dot{w}_o/\dot{w}$

R = injection oxidizer mass fraction, $\hat{W}_o/\hat{W}$

The second parameter is a mixing c* efficiency, $\eta_{c*,mix}$, which represents the maximum attainable c* efficiency corresponding to complete propellant gasification:

$$\dot{\eta}_{c*,mix} = \frac{\displaystyle\sum_{i=1}^{n+\bar{n}} c^*(c_i)\dot{w}_i}{c^*(c_{inj})\hat{W}} \qquad (28)$$

where c_{inj} is the injection mixture ratio $(\hat{W}_o/\hat{W}_F)$, c_i is local mixture ratio $(\dot{w}_{O_i}/\dot{w}_{F_i})$ and $\dot{w}_i$, $\hat{W}$, n and $\bar{n}$ have the same meanings as above. Theoretical characteristic velocity is tabulated as a function of mixture ratio.

14

During STC's multiple stream tube analysis, a single value of c* efficiency is calculated from the n_t stream tubes' data at the throat plane:

$$\eta_{c*} = \frac{\sum_{i=1}^{n_t} c*(c_{g_i})\dot{w}_{g_i}}{c*(c_{inj})\dot{W}} \tag{29}$$

Note the distinction between Eq. 29 and 28: local _gasified_ propellant mixture ratios and flowrates are used in Eq. 29 rather than local total mixture ratios and flowrates, as in Eq. 28.

2.2.3 Method of Solution

The numerical integration scheme used to solve each stream tube's system of equations is the simplest first-order Runge-Kutta (or Euler) method. Selected for its simplicity, minimal data storage requirements, low execution times and numerical stability, this method's accuracy is strongly dependent upon using sufficiently small step sizes. This limitation is reduced in importance by using backwards differencing in writing finite-difference equations and by solving the equations twice, using predicted values from the first, or predictor, solution as input data for a second, corrector, solution.

The STC program is first run in a single stream tube mode, i.e., a one-dimensional subsonic combustion analysis is made for the entire chamber using appropriate sums and averages of initial stream tube variables. This is done for two reasons: (1) to verify consistency of input data (initial-plane pressure is adjusted until the one-dimensional throat velocity is within a small tolerance of the calculated throat sound speed), and (2) to provide a mean adiabatic expansion coefficient, $\bar{\gamma}$, for combustion gas flow in the convergent part of the exhaust nozzle.

The latter coefficient is given by:

$$\bar{\gamma} = \left(\frac{\ln \frac{\bar{p}*}{\bar{p}_1}}{\ln \frac{\bar{\rho}*}{\bar{\rho}_1}} \right) \tag{30}$$

where the subscript 1 refers to the beginning of nozzle convergence, the variables p* and ρ* are at sonic conditions and the over-bars refer to the

one-dimensional flow analysis. It is used by the TRANS computer program
(described in the next subsection) to calculate the coordinates of constant
pressure surfaces (isobars) for transonic flow in the nozzle. TRANS isobars
are generated and transferred to STC in non-dimensional terms, so their use in
STC requires knowledge of the nozzle throat radius, R_T (an input parameter),
and sonic flow pressure, p^*. Discussion of the evaluation of p^* is included
in the following outline of STC's multiple stream tube solution.

Following STC average single stream tube analysis and TRANS analysis, the
initial plane is reinitialized with its original input and the STC program
is run in a multiple stream tube mode. Sequentially:

1. The main iteration loop performs the z direction marching. It begins with
   estimates that the changes in chamber pressure, stream tube gas velocities
   and densities across the next Δz increment are equal to their gradients
   in the preceding single stream tube analysis.

2. These estimated properties are used to calculate predicted values for all
   spray behavior parameters. Drag, evaporation and other spray droplet
   size group parameters are computed with controls to: (a) limit evaporation
   if it is found to exceed the amount of spray available, and (b) avoid
   having the sign of $\left[u_s - (u_{d_j}{}^n)_s \right]_{z_1 \to z_2}$ change in a non-physical way due to
   overestimation of drag forces, i.e., droplets cannot accelerate or
   decelerate past the gas velocity within a given Δz.

3. Evaporated spray weights are added to the previous gasified propellant sums
   and the gas phase mixture ratios are computed. Combustion gas properties
   corresponding to these mixture ratios are obtained by linear interpolation
   in the properties table. Corrected estimates are then made of gas
   temperatures, densities, and stream tube areas.

4. Spray gasification and drag terms are next treated as known constants in
   an implicit-explicit two-step solution for the gas phase properties, stream
   tube flow areas and chamber pressure in plane z_2. In the first step, a
   pressure level in plane z_2 is assumed in order to calculate a predicted
   distribution of stream tube areas in that plane. In the second step, that
   distribution of areas is assumed to be valid, making possible an explicit
   solution for the gas temperatures, densities and pressure at z_2.

16

5. The foregoing paragraphs 2, 3, and 4 describe a predictor cycle. Their calculations are repeated in a corrector cycle, using the predictor cycle's calculated z_2-plane results instead of the estimated properties (paragraph 1). If evaporation of a spray group is calculated in the corrector cycle to exceed the total mass of that group, the group is said to be completely gasified. (For improved accuracy, a user may elect to perform additional corrector cycles.)

6. At this point, normal program flow is printout of computed data at plane z_2, reinitialization of plane z_1 at plane z_2 for a new Δz and progressing again through the paragraph 1 through paragraph 5 computations. This procedure would continue for non-axisymmetric stream tubes until nozzle throat plane was reached. With DER's axisymmetric annular stream tubes, however, the procedure is changed (as follows) at the position in the nozzle where curvature of the isobaric surfaces is introduced, and marching continues for as many as 25 Δz's past the throat plane.

7. An approximate value of p* is estimated from the nozzle throat plane pressure of the preceding averaged, single stream tube analysis: $p* = \bar{p}* \, p(z_0)/\bar{p}(z_0)$. By multiplying the reduced pressures, p/p*, of the TRANS isobars by this value of p*, absolute pressures are calculated for the transonic flow field. These are imposed upon the multiple stream tube nozzle flow.

The furthest upstream TRANS isobar may be planar or curved, depending upon the nozzle's radius ratio and shape of its convergent section. If it is curved, it is desirable to introduce a gradual transition from planar isobars to that first curved isobar which the solution encounters. Also, a gradual transition is desirable to smooth out any discontinuity in pressure levels between those solved for upstream and those imposed downstream of the transition. The gradual transition is provided by stopping the solution for pressure level at a position that is upstream of the nozzle throat by 1.3 times the axial distance that the furthest-upstream TRANS isobar intersects the nozzle wall, and using linear interpolation to impose absolute pressures over the transition interval.

In the transonic region, imposition of absolute pressures makes the paragraph 4 solution for pressure level redundant, so that the gas phase calculation is reduced to a one-step, explicit solution for stream tube areas. Because absolute pressures have been imposed, the solution row provides absolute values of stream tube areas and these may or may not conform to the local nozzle flow area (i.e., satisfy area continuity). By this method, area continuity can be satisfied only by finding the appropriate value of p* to define the proper nozzle pressure level. This is done only for the minimum flow area, as follows.

8. As the solution marches through the transonic flow regime, the minimum value of the sum of the calculated stream tube areas at any preceding z-plane is stored. Later, after the solution has reached a z-plane that's wholly downstream of the TRANS isobar which intersects the nozzle wall at the throat, this minimum area sum is compared with the input geometric area of the nozzle throat. It is the match of these areas that constitutes satisfaction of the nozzle throat boundary condition. If the fractional deviation is less than some input tolerance, ε_{A_t}, the solution is complete and data for subsequent input to the TDK computer program are punched out. If the deviation lies between one and three times ε_{A_t}, the value of p* is adjusted and the solution is recalculated from the point in the nozzle where absolute pressures were imposed. For deviations exceeding three ε_{A_t}, the STC initial-plane pressure is adjusted and the entire multiple stream tube solution is recalculated.

2.3 TRANSONIC NOZZLE FLOW:
TRANS SUBPROGRAM BLOCK

A transonic flow analysis section was adapted from the reference TDK computer program (Ref. 8), as modified (Ref. 9) to utilize an elliptic coordinate transformation solution method (Ref. 10). This section was removed from the TDK program and modified so that it would generate a family of isobaric lines throughout the transonic flow regime and provide a computer-plotted graph of that family. The necessary input data are obtained from the averaged, single stream tube solution of STC, so this TRANS subprogram block gives a homogeneous flow solution. For homogeneous flow, TRANS solutions are stable with radius ratios as small as 5/8.

18

As input data, TRANS needs values only of the nozzle throat radius, R_R, and a mean expansion coefficient, $\bar{\gamma}$. Isobaric coordinates are calculated in terms of axial distance, X, from the throat plane and radial distance, R, from the nozzle axis; both dimensions are normalized to the throat radius. Multiple isobars are generated, one at a time, by starting downstream of the throat and marching upstream with equal intervals, $\Delta\alpha$, in the angle between the nozzle axis and a line tangent to the nozzle wall at the isobar/wall intersection point. The program is structured such that that intersection point for the fifth isobar is at the throat; this isobar later becomes the TDK start-li . . Four isobar/wall intersection points lie downstream of the throat ($\alpha > 0$) and the remainder lie upstream of the throat ($\alpha < 0$). The angular interval between isobars is given by:

$$\Delta\alpha = -\left(1 + \frac{2}{R_R}\right) \tag{31}$$

Generation of isobars continues until either (1) there are twenty of them or (2) an isobar exhibits significant reverse, or upstream curvature. In the latter case, that last upstream-curving isobar is replaced with a planar surface.

Two computer-plotted examples from TRANS analyses are shown in Fig. 3, where: the nozzle axis is at the bottom (R/RT = 0); flow direction is from left to right; a portion of the nozzle wall, defined by $-30° \leq$ wall angle $\leq + 8°$, is shown as the upper curve; isobars are generated from right to left at nozzle wall angle intervals of 2.00° (Eq. 31 was not used in these runs). The monotonic downstream curvature of the constant pressure surfaces is apparent, as is its accentuation by lowering the nozzle radius ratio. Included on the figure are tables which list the pressure ratio, p/p*, and Mach number for each isobar.

The TRANS program also calculates a nozzle discharge coefficient using the 3rd order equation given in Ref. 10:

$$C_{ND} = 1 - \frac{\bar{\gamma}+1}{(1+R_R)^2}\left[\frac{1}{96} - \frac{(8\bar{\gamma}-27)}{2304(1+R_R)}\right. \tag{32}$$
$$\left. + \frac{(754\bar{\gamma}^2 - 757\bar{\gamma} + 3633)}{276,408(1+R_R)^2}\right]$$

19

a) Radius Ratio = 2.00

b) Radius Ratio = 1.00
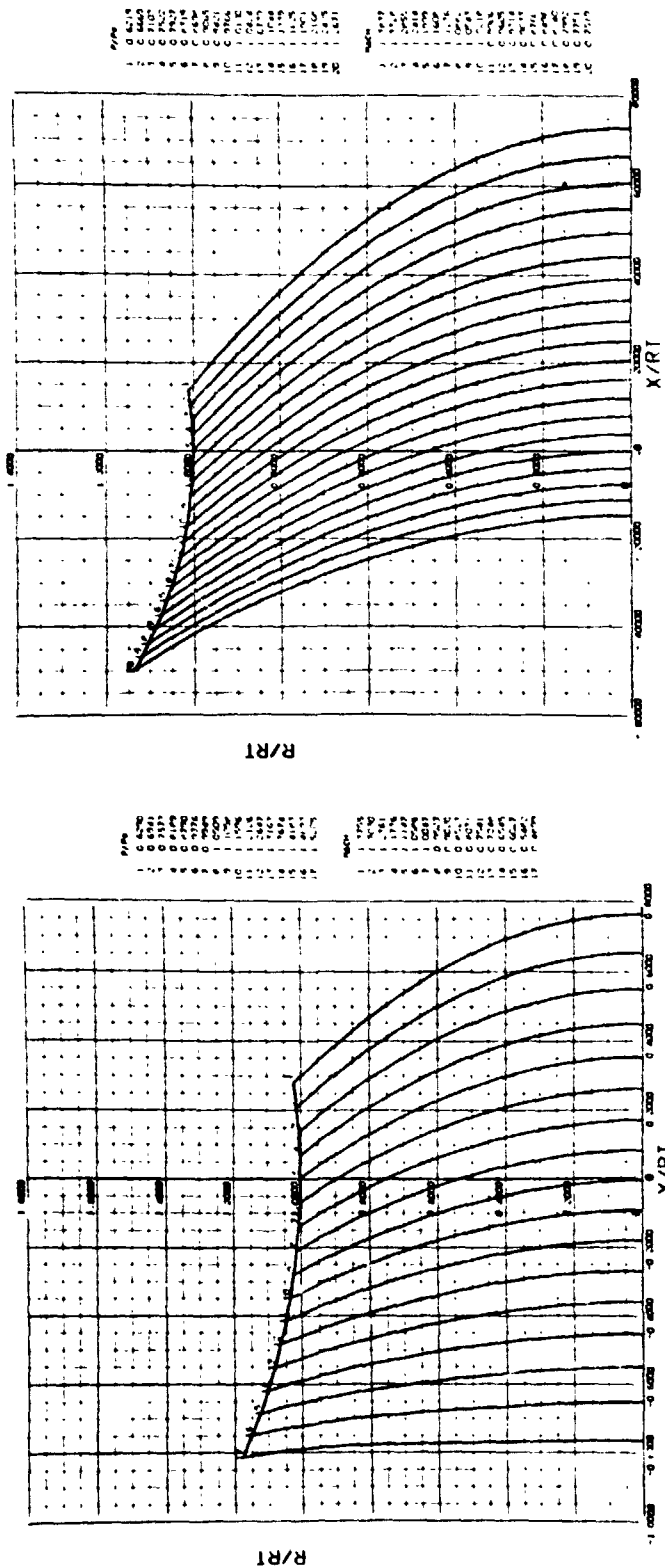
Figure 3. Nozzle Pressure Distributions Calculated
by the TRANS Computer Program

20

Before proceeding to multiple stream tube STC program analysis, the STC initial-plane pressure which satisfied one-dimensional sonic throat flow is divided by C_{ND} to obtain an improved estimate of initial plane pressure and expedite convergence on the nozzle throat boundary condition.

2.4 SUPERSONIC NOZZLE FLOW:
INTERFACE WITH THE TDK COMPUTER PROGRAM

The previous version (Ref. 2) of DER included a TDK subprogram block for performing supersonic nozzle expansion of multiple, axisymmetric, gaseous stream tube flows. That subprogram block was based on minimum modifications of the TDK long-form option, described in Ref. 8, wherein only supersonic analysis is performed. A supersonic start line was initialized from STC computed data and from equilibrium computations using the ODE (one-dimensional equilibrium) section of TDK. The TDK start line data generated by STC were printed out at the end of STC computations and a corresponding punched-card deck was generated. Thus, TDK could be run, optionally, either in sequence with STC using data transferred via scratch data units or separately using punched card input data.

A substantial discrepancy was found to exist between the start line gas temperatures and densities computed by STC and those computed by TDK as start line equilibrium. The discrepancy is caused by the difference between STC's frozen expansion from stagnation equilibrium and TDK's static equilibrium. Under another program (Ref. 11) a modified STC solution method has been developed which eliminated this discrepancy. This method involved: expanding the input tables of equilibrium gas properties to be functions of both mixture ratio and flow Mach number; using static equilibrium properties rather than stagnation; forming (internally) pseudo-stagnation temperature arrays from the input static temperatures; and, interpolating to local mixture ratio and Mach number in these tables.

At the same time, however, an improved version of TDK was also developed (Ref. 3) which is capable of performing kinetic expansion analyses for multiple gaseous stream tubes in subsonic and transonic flow regimes as well as in the supersonic nozzle flow. While it would have been possible to adopt the modified

21

STC subprogram block from Ref. 11 and modify the improved TDK to accept STC data as input, these actions were not taken. Rather, the TDK subprogram block was removed from DER and the frozen expansion from stagnation equilibrium was retained. A subroutine was added which prints and punches, in NAMELIST format, the throat plane data needed from STC to continue the multiple stream tube analysis via the improved TDK computer program.

Throat plane data punched out (and the TDK parameters to which they correspond) are: the number of stream tubes (NZØNES), a stream-tube-area-weighted mean stagnation pressure (P(1)),

$$\overline{P_o^*} = \frac{\displaystyle\sum_{i=1}^{n_t} P_{o\,i}^* A_i}{\displaystyle\sum_{i=1}^{n_t} A_i} \tag{33}$$

and, for each stream tube, the gasified propellant mixture ratio (ØFSKED) and mass fraction of the total gasified propellant flow within that stream tube (XM).

It is anticipated that these stream tube data may be used to initiate TDK nozzle expansion analysis at the throat plane or, if kinetic effects are believed to be important in the subsonic flow, at some plane upstream of the nozzle throat. Thus, it is now possible to overlap the spray gasification analysis of STC with the non-equilibrium combustion analysis of TDK, but in an uncoupled manner. The location of the TDK initial plane is designated by the program user, by specifying the contraction ratio (ECRAT) for that start plane and also the subsonic area ratio for that plane (SUBAR(1), equal to the contraction ratio). This non-physical stream tube data transfer is illustrated in Fig. 4. Other data needed for initializing TDK are*: reactant cards (pages 6-9 and 10), reactions cards (pages 6-27 to 31), nozzle design parameters (pages 6-32 to 34), integration and print controls (pages 6-35 to 37) and TDK controls and nozzle divergence geometry (pages 6-42 to 46).

---

*Page numbers in parentheses refer to the Program User's Manual Section of Ref. 3.

22
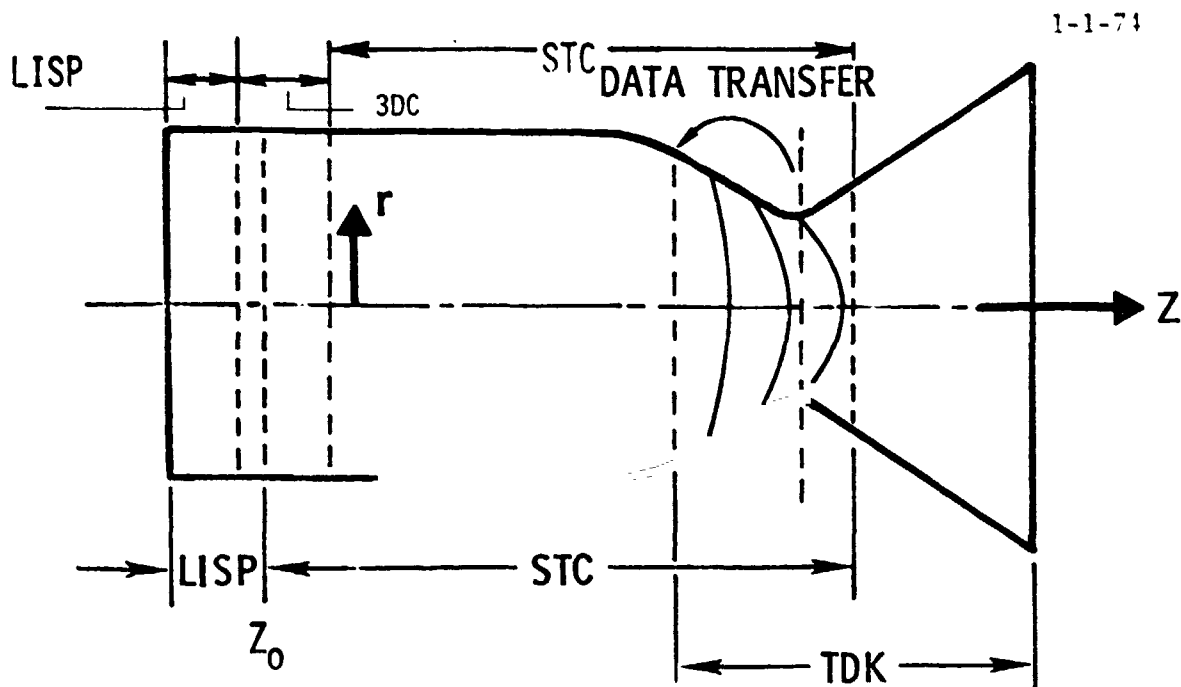
Fig. 4.   Revised STC/TDK Interface Permitting TDK
Analysis to Begin at Subsonic Flow Conditions

Modifications have been made to the improved TDK program to make it compatible
with DER. TDK analyzes only the gasified propellant flow. Account is taken
of the propellant mass loss represented by residual sprays passing through the
throat, assuming no continued evaporation or acceleration that such sprays
might undergo in the supersonic nozzle expansion section. Residual spray is
not added to the TDK initial plane momentum.

3. DER PROGRAM USER'S GUIDE

The DER computer program was developed for operation on Rocketdyne's IBM System 360, Mod 50/65 computer which is designed to run programs written in Fortran H. So that this program would be compatible with other computers, however, it was written in Fortran IV (which is a subset of Fortran H). There are, of course, some sub-programs which may not be operable on other than the Rocketdyne computer; for most other computers, these are probably restricted to the data-plotting functions and can be replaced by dummy subroutines without detriment to the rest of the program functions. In order to run the program on any computer, a user must supply program control cards that are compatible with his compiler, link editor, etc. The program makes extensive use of overlay in order to reduce computer storage requirements; the overlay structure is described in Section 3.1. Storage required and other operational considerations are discussed in Section 3.4.

Operation of the DER computer program also depends upon a user-supplied data deck, through which he specifies details for the particular combustor and propellants he desires to analyze. Assembling a data deck usually involves expenditure of a substantial effort. There are separate sections of the data deck for each major subprogram block; they are assembled in the order in which the program calculations proceed. Details concerning data deck assemblage are given in Section 3.3.

3.1 PROGRAM OVERLAY STRUCTURE

The DER computer program can be overlayed to operate on a computer having only moderate storage capacity. The recommended overlay structure is shown in Fig. 5, where the branches are set us such that most of them are required only once. The program length with this structure is 42,450 words; maximum length occurs when the branch comprising CSPRAY, et seq., is loaded. Total storage used during execution on Rocketdyne's IBM System 360, Mod. 65 computer is about 47,200 words.
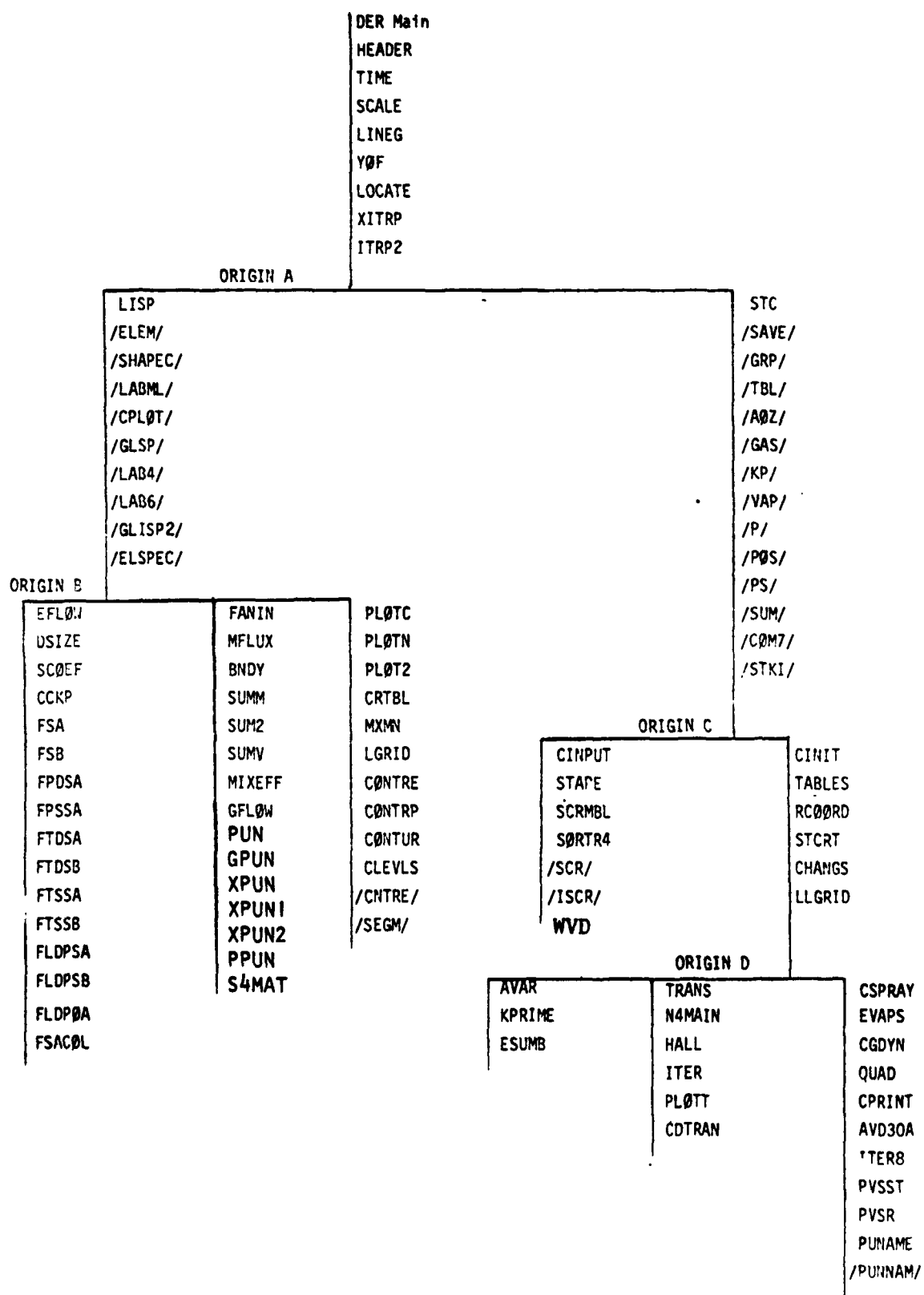
24

```
                                    DER Main
                                    HEADER
                                    TIME
                                    SCALE
                                    LINEG
                                    YØF
                                    LOCATE
                                    XITRP
                                    ITRP2

            ORIGIN A

    LISP                                              STC
    /ELEM/                                            /SAVE/
    /SHAPEC/                                          /GRP/
    /LABML/                                           /TBL/
    /CPLØT/                                           /AØZ/
    /GLSP/                                            /GAS/
    /LAB4/                                            /KP/
    /LAB6/                                            /VAP/
    /GLISP2/                                          /P/
    /ELSPEC/                                          /PØS/
                                                      /PS/
ORIGIN B                                              /SUM/
    EFLØW       FANIN       PLØTC                     /CØM7/
    USIZE       MFLUX       PLØTN                     /STKI/
    SCØEF       BNDY        PLØT2
    CCKP        SUMM        CRTBL
    FSA         SUM2        MXMN
    FSB         SUMV        LGRID           ORIGIN C
    FPDSA       MIXEFF      CØNTRE
    FPSSA       GFLØW       CØNTRP      CINPUT        CINIT
    FTDSA       PUN         CØNTUR      STAPE         TABLES
    FTDSB       GPUN        CLEVLS      SCRMBL        RCØØRD
    FTSSA       XPUN        /CNTRE/     SØRTR4        STCRT
    FTSSB       XPUN1       /SEGM/      /SCR/         CHANGS
    FLDPSA      XPUN2                   /ISCR/        LLGRID
    FLDPSB      PPUN                    WVD
    FLDPØA      S4MAT                           ORIGIN D
    FSACØL                              AVAR      TRANS       CSPRAY
                                        KPRIME    N4MAIN      EVAPS
                                        ESUMB     HALL        CGDYN
                                                  ITER        QUAD
                                                  PLØTT       CPRINT
                                                  CDTRAN      AVD30A
                                                              ITER8
                                                              PVSST
                                                              PVSR
                                                              PUNAME
                                                              /PUNNAM/
```

Figure 5.   DER Program Overlay Structure

Label CØMMØN blocks needed only within a particular segment are included in
the overlay structure. Each is denoted in Fig. 5 by enclosing the CØMMØN
block name between /'s, e.g.,/CNTRE/. There are several label CØMMØN blocks
which are not shown in Fig. 5; these are understood to be part of the root
segment, i.e., that including DER, HEADER, et seq.

3.2 PROGRAM SUBROUTINES

Brief statements are given about each of the subroutines in the DER computer
program, indicating the major functions performed. These are intended to assist
the interested user in finding quickly where particular operations are per-
formed. No attempt is made to give detailed descriptions or to delineate
fully the equations solved or the solution methods used. The subprograms are
divided into three groups, corresponding to the main stem (DEF, et seq.), the
LISP branch and the STC branch, respectively, of the overlay structure. The
sequence of subroutines in this section conforms to the structure of the
overlay chart, Fig. 5.

3.2.1 DER (Main Stem) Subroutines

The DER main program is a brief executive program that calls subroutine HEADER
and, based on values read-in there for certain control parameters, calls
sequentially subprograms LISP, STC, TRANS and PUNAME. If a nonzero value of
the integer INERR is returned from LISP, the rest of the calls are bypassed
and execution is terminated.

Subroutine HEADER reads a set of four comment cards, descriptive of the
particular case being run, and a set of program flow control integers. It
then prints out a title page for the case being run.

Subroutine TIME prints out the date, time of day and elapsed clock (real) time
since the previous call of TIME. This subroutine calls library subroutines
CLØCK and CDATEV; if either of these is unavailable in the library of a
particular computer, subroutine TIME should be replaced by one which utilizes
available library routines or by a dummy subroutine TIME.

26

Subroutine SCALE determines required limits and increments for scaling the axes of computer-plotted graphs. It is referred to by subprograms CLEVLS, LGRID, PLØTT, and LLGRID.

Subroutine LINEG connects successive points of a computer-plotted function with straight line segments. Library subroutine LINEV is called by LINEG.

Function subroutine YØF (X, XT, YT, N, NP) performs an NP-point Lagrangian interpolation to determine $Y = YØF = f(X)$ corresponding to the point (X,Y) in the N-point array (XT, YT). Subroutine LØCATE is called by YØF to find the first of the points (XT(K), YT(K)) for interpolation, where $1 \leq K < N$. Subroutine XITRP is called by YØF to perform the actual point-to-point interpolation.

Subroutine ITRP2 (X1, T1, N1, I1, X2, T2, N2, I2, YT, LI, Y) performs a double interpolation to find $Y = f(X1,X2)$ from the YT(N1,N2) array corresponding to the position X1 in the T1(1) to T1(N1) array and the position X2 in the T2(1) to T2(N2) array. Subroutine LØCATE is called to determine the appropriate first points and subroutine XITRP performs the actual interpolation.

3.2.2 LISP Subroutines

Subroutine LISP is quite large and, as the executive subroutine of this subprogram block, performs many functions. The first of these is reading-in all of the data required for any of its associated subroutines. Some of the input data are tested, as they are read-in, for consistency. If there are inconsistencies in the data which would invalidate LISP's computations, an error message is written out and the case is deleted immediately. Given acceptable data, LISP next sets up the coordinates of the network of mesh points in plane z_0. Then, sequentially, subroutines EFLØW, DSIZE, SCØEFF, CCKP, FANIN, MFLUX and BNDY are called to perform their functions, if appropriate. Subroutine LISP then scales the computed mesh point mass fluxes to ensure conservation of total propellant flowrates, writes out the computed data (both before and after evaporation), generates a scratch data record of information to be transferred to STC, calls subroutine MIXEFF, calculates

27

and prints mean properties for a combustion gas flow at z_0 resulting from gasi-
fication of propellants upstream of that plane. Subroutines PLØTN, CRTLBL,
PLØT2 and PLØTC are called 'f controls have been set to generate computer
plotted graphs of data and, finally, the entire LISP procedure is repeated if
analysis at a second (or third) LISP collection plane has been called for by
reading in non-zero ZØM2 (and ZØM3).

Subroutine EFLØW calculates propellant injection flowrates and velocities for
each injection element of Types 1 through 5 and 7 and 8. For the gas/liquid
element types (Types 6, 9 and 10), the injection flowrates are supplied as
input data. Variables denoting the flowrates of each propellant from each
orifice (or equivalent orifice) of each element are also assigned.

Another function performed in EFLØW is calculation of correlation coefficient
a for the gaseous (fuel) propellant of a Type 6 (gas/liquid coaxial jet)
element. This is done on the basis that the vaporized portion of the oxidizer
spray is diffused and mixed uniformly with the gaseous fuel and that gas phase
momentum is conserved.

Subroutine DSIZE contains equations for calculation of propellant mass median
droplet diameters for elements of Type 1 through 5. These computations are
bypassed if non-zero droplet diameters are supplied as input data.

Subroutine SCØEF provides, for Type 1 through 6 and Type 9 elements, val· ·s
of the mass flux correlation coefficients, C_1 through C_6, a and b for both
propellants. These variables are read-in for Type 8 and 10 elements. SCØEF
calls upon appropriate coefficient library subroutines FSA through FSACØL.

Subroutine CCKP calculates modified evaporation coefficients, C_k, (CKP1 and 2),
in the event that these variables were both read-in with values greater than 1.
In that case, the read-in data are taken to be the propellant latent heats of
vaporization and this subroutine replaces them with a calculated $C_{k'}$. If
both read-in values are less than unity, CCKP is bypassed.

28

Function subroutine FSA provides values for correlation coefficients a and C_4 for Type 1 (unlike doublet) and 2 (like doublet) elements. Function sub-routine FSB provides values for coefficients b, C_1, C_2, C_3, C_5 and C_6 for Type 1 and 2 elements. Both propellants are included.

Function subroutines FPDSA and FPSSA provide values for coefficients a and C_4 for the outer orifices and the central orifice, respectively, of Type 5 (liquid/liquid 4-on-1) elements.

Similarly, function subroutines FTDSA and FTSSA evaluate coefficients a and C_4 for the outer (fuel) orifices and the central (oxidizer) orifice, res: .tively, of Type 3 (liquid/liquid triplet) elements. In like manner, coefficients b, C_2 and C_6 for the triplet's outer and central orifices are obtained from function subroutines FTDSB and FTSSB, respectively.

Spray distribution coefficients for Type 3 (like doublet pair) injection elements are obtained from function subroutines FLDPSA, FLDPSB and FLDPØA. The a coefficients for each propellant differ in the $(-x)$ direction from those in the $(+x)$ direction. The value of a for the "same side" of the origin as a given propellant's doublet lies is obtained from FLDPSA and that for the "opposite side" of the origin from FLDPØA. The b coefficients are calculated by FLDPSB.

Function subroutine FSACØL calculates coefficients a and C_4 for the central liquid jet of a gas/liquid coaxial jet element (Type 6).

Subroutine FANIN was originally intended to account for interference between the spray trajectories of closely-spaced, neighboring spray fans. This function has not been provided, however, so the subroutine's only function is assigning and writing out values of coefficient a for positions to the right $(x>0)$ and left $(x\leqslant0)$ of the element origins.

Subroutine MFLUX performs primary functions of calculating each injection element's contributions to each collection plane mesh point, summing them to obtain mesh point propellant fluxes (both before and after accounting for

partial spray gasification) and calculating mean mesh point spray velocity vectors. Secondary functions are concerned with computing and printing some variables that are indicative of spray flowrate collection efficiencies. MFLUX calls upon a number of computer library trigonometric functions. Arithmetic statement functions define functions SIND(X) and CØSD(X) for angular arguments X in degrees; these definitions should be removed if the library contains these functions.

Subroutine BNDY provides for folding propellant flows which fall outside of the defined chamber segment being analyzed back into particular mesh points. Flow computed to pass radially outward through the chamber wall is folded back into the wall mesh points. Treatment of flow which passes azimuthally through radial planes of symmetry depends upon the value of JSYM and upon whether there are physical barriers (e.g., radial baffles) there. BNDY calls upon subroutines SUMM to sum the propellants folded into the outer wall mesh points, SUM2 to accomplish the folding at radial lines of symmetry and SUMV to calculate appropriate adjustments of the propellant mean velocity vectors at mesh points which have received folded flows.

Subroutine MIXEFF computes values of Rupe's mixing efficiency factor, E_m, and a mass weighted mean characteristic velocity efficiency, $n_{c*,mix}$, based on mesh point propellant flows and a read-in table of c* vs mixture ratio (O/F).

For Type 6 or 10 gas/liquid injection elements, subroutine GFLØW calculates and prints out local mesh point combustion gas conditions. It is assumed that the vaporized portions of the liquid sprays are mixed with the initial gaseous propellant at uniform mixture ratio but the gaseous mass flux can be non-uniform. Combustion gas properties are obtained as functions of that mixture ratio by interpolation in read-in tables.

A package of subroutines is required for punching data cards for 3DC. LISP calls PUN to produce cards to specify spray drop size, velocity components and flowrates at each mesh point. Also, it calls GPUN to produce cards to specify gas mixture ratio, density, velocity and temperature at each mesh point.

Subroutine PUN uses some generalized punch subroutines: PPUN, XPUN, XPUN1 and XPUN2. A variable format is used in PPUN which is set up by subroutine S4MAT. This latter subroutine is dependent on the type of computer used, i.e., in respect to the number of Hollerith characters stored in each word location.

The remaining LISP subroutines are all concerned with generation of computer-plotted graphical output. They are designed to utilize Stromberg-Carlson SC-4010 equipment in producing 9x9-in CRT plots. Subroutine PLØTN produces a cross-sectional graph of the chamber section being analyzed if NCRT $\neq$ 0. The location of each element's origin is indicated thereon. Subroutine PLØT2

is called if NCRT > 0;it plots propellant mass flux vs θ along specified
r = constant arcs. The basic rectangular grid is established by subroutine
LGRID. Subroutine CRTLBL writes labels on the abscissa and ordinate of each
PLØTN or PLØT2 graph.

Subroutine PLØTC constructs circular plots of the system of LISP mesh points
and, on successive graphs, plots contour level maps of fuel, oxidizer and
total propellant flux and one of a function of the basic mixture ratio. If the
maximum and minimum magnitudes of the contour variables are not specified,
they are determined by subroutine MXMN. Subroutine CLEVLS establishes the
actual contour levels plotted, assigning some convenient interval between
levels. Subroutine CØNTUR is called to do the actual contour line plotting.
It, in turn, calls upon subroutine CØNTRE to establish the contours position
between four mesh points and CØNTRP to plot and label the contours in
repeating pie-slice-shaped chamber segments.

3.2.3 STC Subroutines

Subroutine STC is the main executive subroutine for the STC subprogram block
of DER. After calling subroutines CINPUT and CINIT for reading-in data and
initializing the start-plane conditions, respectively, a main DØ-loop is
established which performs the marching from z_0 through the nozzle throat.
Within this DØ, the combustion model equations are solved by stepping from
known conditions in one plane and calculating conditions in a second plane
a small Δz downstream. A number of other subroutines are called upon to carry
out the actual computations. Flow control for predictor-corrector cycle
calculations resides in STC. The logic for testing to see if the single
stream tube analysis has converged on a solution, for reinitializing multiple
stream tube conditions and for testing whether the multiple stream tube analysis
has converged are also in STC.

Subroutine CINPUT reads all of the punched card data needed as input by the
STC subprogram block and, if appropriate, calls upon subroutine STAPE to
read the scratch data unit recorded by subroutine LISP and set up the initial
stream tube areas, propellant flowrates, gas phase mixture ratio and spray
properties. If the option is specified for starting the STC analysis from

31

3DC data, STAPE reads data from cards punched by 3DC rather than from a scratch data unit and droplet flowrate, velocity and diameter are averaged at each mesh point by subroutine WVD.

Subroutine STAPE reads data transferred from LISP via scratch data unit 2 and uses it to set up the stream tubes' initial cross-sectional areas and propellant flowrates and conditions at the STC start plane. STAPE calls subroutine SCRMBL to perform the assignment of mesh points to particular stream tubes and to do the appropriate summation and averaging of areas and propellant parameters. Data returned from SCRMBL are multiplied by an integer scale factor to convert from a pie-slice-shaped sector to a full circular chamber cross-section. Stream tube initialization data are punched-out so they may be used as input for subsequent STC subprogram block runs.

Subroutine SCRMBL is a fairly complicated subroutine, as it contains the logic for accomplishing the several optional ways of combining mesh point flows into stream tube flows described on page 8 . When mesh points are to be combined on the basis of mixture ratio, subroutine SØRTR4 is called to sort the mesh points into a sequence with increasing mixture ratio. This is done by tagging each mesh point with one value of a subscripted integer label IC(K) whose subscript K is uniquely related to the mesh points r,θ subscripts.

Subroutine CINIT performs several functions in preparing for STC's main DØ-loop marching solution. Dummy arrays are defined for saving stream tube initialization data for subsequent occasions when the start plane is reinitialized and STC's computations are restarted. Subroutine KPRIME is called to compute, store and print tables of propellant evaporation coefficients as functions of combustion gas static temperature. Subroutine AVAR is called to set up a table of chamber cross-sectional area vs chamber length and to define some other geometric parameters. If STC single stream tube analysis is being prepared for, the stream tubes are (temporarily) combined into one; thus, only the first of NST stream tubes is properly defined for this option and it carries the total injected propellant flowrates and occupies the entire combustion chamber cross-section. CINIT also contains the logic for reestablishing the multiple stream tube conditions upon completion of single stream tube analysis. Other functions include initialization of the stream tube combustion gas properties and printing of some initial stream tube data.

Subroutine TABLES interpolates in read-in tables of combustion gas properties at local stream tube mixture ratio to find local combustion gas stagnation temperature, gamma, molecular weight and viscosity. Local stagnation sound speed and throat static sound speeds are calculated.

Subroutine RCØØRD establishes the radial coordinates of the dividing stream lines between stream tubes and the incremental stream tube path length corresponding to a Δz axial increment. Downstream of the nozzle throat, the dividing stream lines' intersections with the fifth TRANS isobar are solved for and printed out.

Subroutine STCRT records the dividing stream line radii as functions of chamber length and, upon completion of the chamber length marching analysis, generates an axial chamber cross-section plot.

Subroutine CHANGS reassigns the values of stream tube parameters in the upstream z-plane as being equal to those in the downstream z-plane, as preparation for another Δz step. (When called from CINIT, subroutine CHANGS' function is inverted, assigning upstream conditions to the downstream plane.)

Subroutines KPRIME and AVAR were commented upon along with subroutine CINIT.

Subroutine ESUBM calculates a value of the Rupe mixing efficiency factor, E_m, for the stream-tube-striated combustor flow. Total stream tube propellant flows are considered, so this value corresponds to complete propellant gasification.

Subroutine TRANS is the executive program for the TRANS subprogram block. It is called by the DER main program. TRANS defines a number of control parameters, then writes out a table of input and control data before calling subroutine N4MAIN, which performs the transonic analysis.

Subroutine N4MAIN solves for the radial and axial coordinates of points on isobaric surfaces in an axisymmetric nozzle's transonic flow regime. Subroutine HALL is called upon to perform a modified Hall solution of transonic flow equations for a homogeneous constant flowrate gas. Subroutine ITER is

33

called to solve for the axial coordinate roots. The isobaric surfaces' coordinates are plotted by subroutine PLØTT and are tabulated, together with each line's reduced pressure, Mach number and flow directions, by N4MAIN.

Subroutine CDTRAN is called by TRANS to calculate and print a nozzle discharge coefficient for the foregoing transonic solution.

Subroutine CSPRAY calculates the behavior of propellant spray size groups. Subroutine EVAPS is called to compute the size group gasification rates. CSPRAY then determines whether those rates are permissible, adjusts them if necessary, reduces the liquid spray flowrates and calculates the size group droplet diameters and velocities. If there are multiple size groups of a propellant, the droplet number flowrate is constant and droplet diameter is a variable. However, if there is a single size group for a propellant, the diameter is held constant and the number flowrate is reduced as gasification proceeds.

Subroutine EVAPS calculates size group gasification. In doing so, it calculates droplet Reynolds numbers based on mean droplet film properties. Film viscosity is approximated via Wilke's equation.

Subroutine CGDYN solves the stream tube gas dynamic equations. Four finite-difference equations (continuity, momentum, state and adiabatic) in four variables (velocity, temperature, density and either area or pressure) are combined to form a single quadratic equation for gas velocity. A gas velocity root is determined explicitly, via function QUAD, and is used in the individual equations to solve explicitly for the other variables. Different quadratic equations are used, depending upon whether pressure or stream tube area is (assumed to be) known at plane 2. In the multiple stream tube case, pressure is first assumed to be known in plane 2 in order to solve for a distribution of areas. In a second step, that distribution of downstream areas is assumed to be valid and the downstream pressures are solved for; an area-weighted mean pressure is assigned to the plane. Obviously, in the case of a single stream tube, the downstream area is known and the first step is not needed.

34

Similarly, imposition of absolute pressures, based on the TRANS distribution in the nozzle, makes it possible to bypass the second step for the multiple stream tube case.

Function subprogram QUAD (A,B,C,X1) solves the quadratic equation $AX^2 + BX + C = 0$ for the root whose value is closest to the argument X1. If the discriminant $B^2 - 4AC < 0$, it is set equal to zero and the single root is returned.

Subroutine CPRINT's function is to print computed stream tube pressure, area, gas stream and propellant spray data at specified multiples of Δz from the STC start plane. Different formats are used for the single and multiple stream tube cases.

Subroutine AVD30A calculates a value of volume-number mean droplet diameter, D_{30}, for each propellant. It is called by subroutine CPRINT for single stream tube print out.

Subroutine ITER8 records data, on scratch data unit No. 3, at the plane immediately preceding that plane where a radial distribution of pressures is imposed on the multiple stream tube flow. These data may be used to re-initialize stream tube combustion analysis with adjusted values of pressure, gas densities and velocities at that plane if the computed minimum stream tube area sum is not sufficiently close to the nozzle throat area. Another data record is recorded at the throat plane; it is read by subroutine PUNAME and used for preparing input data for the improved TDK.

Subroutine PVSST calculates the radial distribution of pressure in a z-plane and assigns a pressure to each stream tube corresponding to its mean radial position. Subroutine PVSR is called to perform the interpolation between TRANS isobars.

Subroutine PUNAME reads data from data unit No. 3 and calculates data for initializing the improved version of TDK. A punched card deck is generated, using the NAMELIST format appropriate for TDK input data. The deck is also printed out. Additionally, values of a number of other variables, which may by needed in a future revision of TDK, are calculated and printed out.

35

3.3 PROGRAM INPUT DATA

In general, each of the two major program blocks LISP and STC performs its own
input and output functions, and they can be run sequentially or individually.
When executed sequentially, coupling data are transmitted via scratch data
unit from LISP to STC and other data are supplied directly to STC as required
via punched card input. When run individually,all data to a block are supplied
via card input. Punched card output of the coupling data are also provided
(optionally for STC) for convenience in running subsequent STC analyses
without having to rerun LISP and for subsequent analyses using the improved
TDK program (Ref. 3). (For example, the same LISP output might be used for
several STC runs with different chamber and nozzle lengths, or TDK might be
run more than once with different nozzle contours or expansion ratios using
a single STC output.)

Input data requirements are indicated in Tables 1 and 2. Table 1 shows, in
the required order, Fortran code sheets for all of the required data cards.
The "DESCRIPTION" column gives the DER program's Fortran name for each
variable; definitions of these input variables and their units are listed
alphabetically in Table 2. Values of the variables are entered in appropriate
fields occupying columns 1 through 72, using the format designated in the
"IDENTIFICATION" space at the bottom of each card in Table 1. Sequence numbers
are denoted in columns 73 through 80 which both leave room for the maximum
sizes of subscripted arrays and ensure that the data deck is sortable. In
some instances, where multiple sets of the same cards may need to be entered,
blank spaces appear in the sequence numbers.

Whether or not certain data are actually required for a particular case depends
upon values read-in earlier in the case for particular control parameters.
These controls are indicated in Table 1 either as a condition attending major
blocks of cards or parenthetically beside the card specification.

Discussions guiding the selection of input data are subdivided under LISP
and STC subheadings following Table 2.

TABLE 1.   DER Computer Program, Input Data Cards

A.   Data Read by Subroutine Header, Every Case

	NUMBER	DESCRIPTION
1		1st DER Comment Card
13		
25		
37		
49		
61		
	IDENTIFICATION 18A4 73	1 0 80
1		2nd DER Comment Card
13		
25		
37		
49		
61		
	IDENTIFICATION 18A4 73	2 0 80
1		3rd DER Comment Card
13		
25		
37		
49		
61		
	IDENTIFICATION 18A4 73	3 0 80
1		4th DER Comment Card
13		
25		
37		
49		
61		
	IDENTIFICATION 18A4 73	4 0 80

37

TABLE 1 (CONT'D)

NUMBER	DESCRIPTION
1	ILISP
13	ISTC
25	ITRANS
37	ITDK
49	
61	
IDENTIFICATION 6I12 73	5 0 80

B. Data Read by Subroutine LISP, If (ILISP ≠ 0)

(If ILISP = 0, omit cards 2010 through 4630)

NUMBER	DESCRIPTION
1	1st LISP Comment Card
12	
25	
37	
49	
61	
IDENTIFICATION T8A4 73	2 0 1 0 80

NUMBER	DESCRIPTION
1	2nd LISP Comment Card
13	
25	
37	
49	
61	
IDENTIFICATION 18A4 73	2 0 2 0 80

NUMBER		DESCRIPTION	
1		NEL	NRML
13		NTHML	NRWALL
25		NTHR	NTHL
37		JSYM	NRBAFR
49		NRBAFL	NLSPEC
61		NCRT	IPUN
IDENTIFICATION 12I6 73		2 0 3 0 80	

38

TABLE 1 (CONT'D)

NUMBER		DESCRIPTION	
1		IPUNR	IPUNL
13		KFCRT	KØCRT
25		KTCRT	KFFCRT
37		NCSTR	
49			
61			
IDENTIFICATION 12I6 73		2 0 4 0	80

1		IRCRT(I)	(If
13		[=] NCRT	NCRT≤0
25			omit
37			this
49			card)
61			
IDENTIFICATION 12I6 73		2 0 5 0	80

1		DRADM
13		DTHETM
25		ZØM
37		THETAR
49		DPINJ1
61		DPINJ2
IDENTIFICATION 6E12.8 73		2 0 6 0	80

1		DENS1
13		DENS2
25		CKP1
37		CKP2
49		ZØM2
61		ZOM3
IDENTIFICATION 6E12.8 73		2 0 7 0	80

TABLE 1 (CONT'D)

1-1-74

	NUMBER		DESCRIPTION		
1			RMNØM		Cards must be
13			DNSAT1		Included
25			DNSAT2		for any case.
37			PRPR1		Values are
49			PRPR2		needed only
61			RHØG		if Type 2 or 3 elements
	IDENTIFICATION 6E12.8 73		2 0 8 · 80		are speci-
1			EPS		fied.
13			TPRØP1		
25			TPRØP2		
37					
49					
61					
	IDENTIFICATION 6E12.8 73		2 0 9 0 80		

Provide a Set of Cards 2-10 through 2-95 for Each Element Specification

	NUMBER		DESCRIPTION	
1			NTYPE	NPRØP1
13			NPRØP2	IDBAR
25				
37				
49				
61			Elem. Spec. ↓	
	IDENTIFICATION 12I6 73		2 1 0 80	
1			DIA1 (Not needed for Type 6)	
13			DIA2	
25			CDDIA1	
37			CDDIA2	
49			ZE	
61			GAME	
	IDENTIFICATION 6E12.8 73		2 2 0 80	

40

TABLE 1 (CONT'D)

1-1-74

NUMBER	DESCRIPTION	
1	BETA	
13	GAMMA	
25	DELTZG	} (Only for
37	DELTZL	NTYPE=10
49		See p. 72b,
61		δ_G and δ_L)
IDENTIFICATION 6E12.8 73	2 3 0 |80	

NUMBER	DESCRIPTION	
1	GAMFAN	(Only for
13	SPFAN	NTYPE=3)
25	SPEL	Otherwise, omit this card.
37		
49		
61		
IDENTIFICATION 6E12.8 73	2 4 0 |80	

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NUMBER	DESCRIPTION	
1	DBAR1	(For IDBAR≠0
13	DBAR2	and for
25		NTYPE≥6)
37		Otherwise,
49		omit this
61		card.
IDENTIFICATION 6E12.8 73	2 5 0 |80	

NUMBER	DESCRIPTION	
1	WØT1	(Only for
13	AØ1	NTYPE=9)
25		Otherwise,
37	(For NTYPE=9, Only)	omit this
49		card.
61		
IDENTIFICATION 6E12.8 73	2 5 5 |80	

41

TABLE 1 (CONT'D)

NUMBER	DESCRIPTION	
1	SC11	(Only for NTYPE=8 or 10) Otherwise, omit these cards
13	SC21	
25	SC31	
37	SC41	
49	SC51	
61	SC61	
IDENTIFICATION 6E12.8 73	2 6 0 80	
1	SC12	
13	SC22	
25	SC32	
37	SC42	
49	SC52	
61	SC62	
IDENTIFICATION 6E12.8 73	2 7 0 80	
1	SA1	
13	SB1	
25	SA2	
37	SB2	
49		
61		
IDENTIFICATION 6E12.8 73	2 8 0 80	
1	WØT1	(Only for NTYPE=10) Otherwise, omit this card.
13	WØT2	
25		
37		
49		
61		
IDENTIFICATION 6E12.8 73	2 9 0 80	
1	WØT1	(Only for NTYPE=6) Otherwise, omit this card.
13	WØT2	
25	AØ1	
37	PSTREC	
49		
61		
IDENTIFICATION 6E12.8 73	2 9 5 80	

42

TABLE 1 (CONT'D)

1-1-74

Enter One of the Following Cards for Each Injection Element

NUMBER	DESCRIPTION	
1	LSPEC	
13	RADE	(1st Element)
25	THETAE	
37	ALFA	
49		
61	Add Seq. No.	
IDENT	I12, 3E12.8 73	3 .1 0 80
1	LSPEC	
13	RADE	(2nd Element)
25	THETAE	
37	ALFA	
49		
61		
IDENT	I12, 3E12.8 73	3 .2 0 80
1	Etc.	
13		(Continue
25		Through
37		NEL
49		Elements)
61		
IDENT	I12, 3E12.8 73	3 .3 0 80
1	W1F	(Only if
13	W2F	KFCRT≠0
25	W1Ø	KØCRT≠0
37	W2Ø	KTCRT≠0
49	W1T	or
61	W2T	KFFCRT≠0
IDENTIFICATION	6E12.8 73	4 0.1 0 80

43

TABLE 1 (CONT'D)

NUMBER	DESCRIPTION
	W1FF
	W2FF

IDENTIFICATION 6E12.8 73   4 0 2 0   80

Otherwise, omit these cards.

Provide Cards 4110 through 4630 Only if NCSTR≠0

(If NCSTR= 0, omit cards 4110 through 4630)

NUMBER	DESCRIPTION
	TMR(I).
	I=1,NCSTR

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

IDENTIFICATION 6E12.8 73   4 1 1 0 80

	TMR(I)
	(Continued)

(If NCSTR>6)

IDENTIFICATION 6E12.8 73   4 1 2 0 80

	TMR(I).
	(Continued)

(If NCSTR>12)

IDENTIFICATION 6E12.8 73   4 1 3 0 80

TABLE 1 (CONT'D)

	NUMBER			DESCRIPTION	
1				TGØ(I),	(Plus 4220
13				I=1,NCSTR	& 4230, if
25					appropriate)
37					
49					
61					
IDENTIFICATION 6E12.8 73				4.2.1.0	80
1				TMU(I),	(Plus 4320
13				I=1,NCSTR	& 4330, if
25					appropriate)
37					
49					
61					
IDENTIFICATION 6E12.8 73				4.3.1.0	80
1				TGM(I)	(Plus 4420
13				I=1,NCSTR	& 4430, if
25					appropriate)
37					
49					
61					
IDENTIFICATION 6E12.8 73				4.4.1.0	80
1				TMW(I),	(Plus 4520
13				I=1,NCSTR	& 4530, if
25					appropriate)
37					
49					
61					
IDENTIFICATION 6E12.8 73				4.5.1.0	80

TABLE 1 (CONT'D)

1-1-74

NUMBER	DESCRIPTION	
1		CST(I)
13		I=1,NCSTR
25		
37		
49		
61		
IDENTIFICATION 6E12.8 73	4 6 1 0	80

(Plus 4620 & 4630, if appropriate)

C. Data Read by Subroutine CINPUT, if ISTC≠0

(If ISTC= 0, this is the end of the data deck)

	DESCRIPTION	
1	NØZØN	
13	NSTPZ	
25	NUG	
37	IPUN3D	
49		
61		
IDENTIFICATION 6I12 73	5 0 1 0	80

	DESCRIPTION	
1	NP	
13	NAP	
25	NMR	
37	NTK	
49		
61		
IDENTIFICATION 6I12 73	5 0 2 0	80

	DESCRIPTION	
1	APRØF(1,1)	
13	APRØF(1,2)	
25	APRØF(2,1)	
37	APRØF(2,2)	
49	APRØF(3,1)	
61	APRØF(3,2)	
IDENTIFICATION 6E12.8 73	5 0 3 0	80

46

TABLE 1 (CONT'D)

NUMBER	DESCRIPTION	
APRØF(4,1)		(If NAP>3)
APRØF(4,2)		
⋮		
(NAP Pairs)		(Through 5060 if appropriate
IDENTIFICATION 6E12.8 73	5 0 4 0 80	

	TITLE P.	
	Propellant	
	Comment	
	Card	
IDENTIFICATION 18A4 73	5 1 0 0 80	

The following cards 5110 through 5630 are identical to cards
4110 through 4630, preceding, if NMR = NCSTR

	TMR(I),	
	I=1,NMR	
IDENTIFICATION 6E12.8 73	5 1 1 0 80	

	TMR(I),	(If NMR>6)
	(Continued)	
IDENTIFICATION 6E12.8 73	5 1 2 0 80	

47

TABLE 1 (CONT'D)

NUMBER	DESCRIPTION	
	TMR(I)	(If NMR>12)
	(Continued)	
IDENTIFICATION 6E12.8 73	5 1 3 0 80	
	TT0(I)	(Plus 5220 &
	I=1,NMR	5230, if
		appropriate)
IDENTIFICATION 6E12.8 73	5 2 1 0 80	
	TVIS(I)	(Plus 5320 &
	I=1,NMR	5330, if
		appropriate)
IDENTIFICATION 6E12.8 73	5 3 1 0 80	
	TGAM(I)	(Plus 5420 &
	I=1,NMR	5430, if
		appropriate)
IDENTIFICATION 6E12.8 73	5 4 1 0 80	

48

TABLE 1 (CONT'D)

	NUMBER	DESCRIPTION	
1		TMW(I)	
13		I=1,NMR	
25			
37			
49			(Plus 5520 & 5530, if appropriate)
61			
IDENTIFICATION 6E12.8 73		5.5.1 0 80	
1		CSTR(I),	
13		I=1,NMR	
25			
37			
49			(Plus 5620 & 5630, if appropriate)
61			
IDENTIFICATION 6E12.8 73		5.6.1 0 80	
1		TVF(I)	
13		I=1,NTK	
25			
37			
49			(Through 5740 if appropriate)
61			
IDENTIFICATION 6E12.8 73		5 7 1 0 80	
1		CPVF(I)	
13		I=1,NTK	
25			
37			
49			(Through 5840 if appropriate)
61			
IDENTIFICATION 6E12.8 73		5.8.1 0 80	

TABLE 1 (CONT'D)

	NUMBER	DESCRIPTION	
1		TCØNVF(I)	
13		I=1,NTK	
25			
37			
49			(Through 5940, if appropriate)
61			
IDENTIFICATION **6E12.8** 73		5 9 1 0	80
1		TVØ(I),	
13		I=1,NTK	
25			
37			
49			(Through 6040, if appropriate)
61			
IDENTIFICATION **6E12.8** 73		6 0 1 0	80
1		CPVØ(I)	
13		I=1,NTK	
25			
37			
49			(Through 6140, if appropriate)
61			
IDENTIFICATION **6E12.8** 73		6 1 1 0	80
1		TCØNVØ(I)	
13		I=1,NTK	
25			
37			
49			(Through 6240, if appropriate)
61			
IDENTIFICATION **6E12.8** 73		6 2 1 0	80

50

TABLE 1 (CONT'D)

NUMBER		DESCRIPTION	
1		TNBF	
13		TBF	
25		RHØNBF	
37		RHØLF	
49		WTMLLF	
61		WTMLVF	
IDENTIFICATION 6E12.8 73		6.5.1 0	80
1		TNBØ	
13		TBØ	
25		RHØNBØ	
37		RHØLØ	
49		WTMLLØ	
61		WTMLVØ	
IDENTIFICATION 6E12.8 73		6.5.2 0	80
1		TCRITF	
13		TCRITØ	
25		DHVF	
37		DHVØ	
49			
61			
IDENTIFICATION 6E12.8 73		6.5.3 0	80
1		IST	
13		NSSTI	
25		NMSTI	
37		ICRC	
49		IPRSST	
61		IPRMST	
IDENTIFICATION 6I12 73		6.5.4 0	80

51

TABLE 1 (CONT'D)

1-1-74

NUMBER		
1		CRTØL
13		ARTØLD
25		
37		
49		
61		
IDENTIFICATION 6E12.8 73		6 5 5 0 80

1		PCI
13		ZSTART
25		PCTBL
49		
61		
IDENTIFICATION 6E12.8 73		6 5 6 0 80

(For ILISP≠0
or IPUN3D=0

1		NGT
13		NGF
25		ND
37		
49		
61		
IDENTIFICATION 6I12 73		6 5 7 0 80

1		FRACUM(I)
13		I=1,ND
25		
37		
49		
61		
IDENTIFICATION 6E12.8 73		6 5 8 0 80

Otherwise, omit these cards.

(For ILISP≠0,
ND>0)
(Plus 6590, if
ND>6)

52

TABLE 1 (CONT'D)   1-1-74

NUMBER	DESCRIPTION	
1		DØDBAR(I), I=1, ND
13		(Continued)
25		
37		
49		
61		
IDENTIFICATION 6E12.8 73	6 6 1 0 80	

Otherwise, omit these cards →

(Plus 6620, if ND > 6)

This is the end of the data deck if ILISP≠0.

If IPUN3D=1, the data deck is completed with the punched card output from 3DC.

1		PCI PC
13		ZSTART
25		
37		
49		
61		
IDENTIFICATION 6E12.8 73	6 7 1 0 80	
1		NGT
13		NGF
25		NST
37		NASEG
49		
61		
IDENTIFICATION 6I12 73	6 7 2 0 80	

(Include cards only if ILISP=0 and IPUN3D=0)

53

TABLE 1 (CONT'D)   1-1-74

The following cards (7010 et seq.) are punched by subroutine STAPE if ILISP≠0 and IPUN≠0 in LISP input, as preparation for this input when ILISP=0

NUMBER	DESCRIPTION
1	AREA1(1)
13	GASFL(1)
25	STRG(1)
37	SNN(1)
49	SR(1)
61	STH(1)
IDENTIFICATION 6E12.8 73	7 0 1 0 80
1	GWSPR(I,J)
13	GVELD1(I,J)
25	GDIAD1(I,J)
37	GWSPR(I+1,J)
49	GVELD1(I+1,J)
61	GDIAD1(I+1,J)
IDENTIFICATION 6E12.8 73	7 0 1 1 80
1	GWSPR(I+2,J)
13	etc.
25	Continue through I=1,
37	NGT spray groups
49	
61	
IDENTIFICATION 6E12.8 73	7 0 1 2 80
1	AREA1(J+1),
13	etc.
25	
37	Next Stream Tube Starts
49	on a New Card
61	
IDENTIFICATION 6E12.8 73	7 0 2 0 80

(Include cards only if ILISP=0 and IPUN3P=0)

Continue for J=1, NST Stream Tubes

54

TABLE 2. DER INPUT DATA DEFINITIONS AND UNITS
(ALPHABETICAL ORDER) CARD NO.

Fortran Variable Name	Math Symbol	Definition	Units
ALFA(I) $1 \leq I \leq NEL$	α	The counterclockwise angle of element rotation about its z axis (see page 73)	degrees
APRØF(I,J) $1 \leq I \leq NAP$ $1 \leq J \leq 2$		Combustion Chamber Geometry Specification APRØF(I,1) = Axial Distance APRØF(I,2) = Chamber Diameter If APRØF(1,1) $\neq$ 0, APRØF(1,1) = throat radius ratio	in in
AØ1(I) $1 \leq I \leq NLSPEC$		Cross-sectional area of orifice No. 1 of Type 6 or Type 9 element	in^2
AREA1(J) $1 \leq J \leq NST$	A_z	Stream tube cross-sectional area at z = ZSTART	in^2
ARTØLD	ϵ_{A_t}	Decimal tolerance, deviation of computed minimum stream tube area sum from nozzle throat area, multiple stream tube analysis	–
BETA(I) $1 \leq I \leq NLSPEC$	β	Counterclockwise angle of element rotation about its y-axis (see page 73)	degrees
CDDIA1,2(I) $1 \leq I \leq NLSPEC$	C_D	Discharge coefficient for orifice 1,2 of Ith element specification	–
CKP1,2	$C_{k'}$	Modified k' evaporation coefficient for propellant 1(fuel), 2(oxidizer) or Latent heat of vaporization (see page 78)	in^2/sec BTU/lbm
CPVF,Ø(J) $1 \leq I \leq NTK$	$C_{\Gamma v}$	Fuel, oxidizer vapor specific heat at constant pressure	$\dfrac{BTU}{lbm-°R}$
CRTØL		Decimal tolerance, deviation of computed single stream tube throat contraction ratio from unity	–
CST(I) $1 \leq I \leq NCSTR$	c^*	Theoretical characteristic velocity at mixture ratio TMR(I)	$\dfrac{ft}{sec}$
CSTR(I) $1 \leq I \leq NMR$	c^*	Same as CST	$\dfrac{ft}{sec}$
DBAR1,2(I) $1 \leq I \leq NLSPEC$	$\bar{D}$	Mass median droplet diameter for whichever propellant flows from orifice No. 1,2 of Ith element specification	in

55

TABLE 2 (CONT'D

Fortran Variable Name	Math Symbol	Definition	Units
DELTZG,L(I) I = 6 or 10	δ_G, δ_L	Axial location of pseudo-impingement point for gas (fuel), liquid (oxidizer)of Ith element specification (see page 72a)	in
DENS1,2	ρ	Density of propellant 1 (fuel), 2 (oxidizer) at injector temperature	$\frac{lb_m}{ft^3}$
DHVF, $\emptyset$	ΔH_v	Fuel, oxidizer latent heat of vaporization	$\frac{BTU}{lb_m}$
DIA1,2(I) $1 \leq I \leq NLSPEC$	D	Diameter of injection orifice No. 1, 2 of Ith element specification	in
DNSAT1,2	ρ_s	Density of propellant 1 (fuel), 2 (oxidizer) at saturation temperature corresponding to the chamber pressure	$\frac{lb_m}{ft^3}$
DØDBAR(I) $1 \leq I \leq ND$		Spray size group diameter, D, divided by $\bar{D}$: with FRACUM(I), gives spray distribution	–
DPINJ1,2	ΔP	Injection pressure drop for propellant 1 (fuel), propellant 2 (oxidizer)	$\frac{lb_f}{in^2}$
DRADM	Δr	Spacing between circumferential mesh lines	in
DTHETM	$\Delta \theta$	Spacing between radial mesh lines	degrees
EPS	ϵ_c	Combustion chamber contraction ratio	–
FRACUM(I) $1 \leq I \leq ND$		Mass fraction of spray assigned to Ith size group, whose diameter is given by DØDBAR(I)	–
GAME(I) $1 \leq I \leq NLSPEC$	γ_E	Stream impingement angle of Ith element specification	degrees
GAMFAN(I) $1 \leq I \leq NLSPEC$	γ_F	Fan cant angle of Ith element specification (Type 3 only)	degrees
GAMMA(I) $1 \leq I \leq NLSPEC$	γ	Counterclockwise angle of element rotation about its x-axis (see page 73)	degrees
GASFL(J) $1 \leq J \leq NST$	$\dot{w}_g$	Total stream tube gas flow at z=ZSTART	lbm/sec
GDIAD1(I,J) $1 \leq I \leq NGT$ $1 \leq J \leq NST$	D_d^n	Propellant spray size group droplet diameter	in
GVELD1(I,J) $1 \leq I \leq NGT$ $1 \leq J \leq NST$	u_d^n	Propellant spray size group velocity	$\frac{ft}{sec}$

TABLE 2 (CONT'D)

1-1-74

Fortran Variable Name	Math Symbol	Definition	Units
GWSPR(I,J) $1 \leq I \leq NGT$ $1 \leq I \leq NST$	$\dot{w}_d^{\,n}$	Propellant spray size group flowrate	$\dfrac{lb_m}{sec}$
ICRC		Number of corrector cycles calculated at each Δz interval (normally = 1)	–
IDBAR		Integer indicator of source of $\bar{D}$ data: $\neq 0$, $\bar{D}$ must be read in for all elements; $=0$, $\bar{D}$ calculated internally for NTYPE≤ 5	–
ILISP		Control integer; non-zero value causes subroutine LISP to be called	–
IPRMST, IPRSST		Number of Δz intervals between multiple, single stream tube printouts	–
IPUN		Control integer; a non-zero value causes subroutine STAPE to punch-out stream tube initialization data; a value of 1 causes LISP to punch cards for 3DC	–
IPIN3D		Control integer; if equal to 1, STC reads punched cards from 3DC	–
IPUNL,R		Obsolete control integers; enter zeroes	–
IRCRT(I) $1 \leq I \leq NCRT$		Indices of circumferential (r=const) mesh lines along which CRT plots of mass flux are to be generated	–
IST		Redundant control integer, set = 1	
ISTC		Control integer; non-zero value causes subroutine STC to be called	–
ITDK		Obsolete control integer; leave blank	–
ITRANS		Control integer, non-zero value causes subprogram TRANS to be called	–
JSYM		Integer controlling folding-in of flux from exterior radial mesh lines; = 0, no folding; = 1, mirror image folding; = 2, repeating image folding	–
KFCRT, KØCRT, KTCRT and KFFCRT		CRT contour-plot controls, for fuel, oxidizer, total propellant and modified fuel fraction, respectively. A zero value suppresses the plot. A positive value denotes the number of contour lines. A negative value causes subroutine SCALE to select approximately that number of contours at rounded-off intervals. (Maximum values = 35)	–

TABLE 2 (CONT'D)   1-1-74

Fortran Variable Name	Math Symbol	Definition	Units
LSPEC(I) $1 \leq I \leq NEL$		Element specification callout which describes Ith element $(1 \leq LSPEC \leq NLSPEC)$	-
NAP		Number of points defining chamber geometry (≤ 12)	-
NASEG		Integer multiplier for initial stream tube areas and flowrates, normally = 1	
NCSTR		Number of mixture ratios at which combustion gas properties are tabulated in LISP input data (≤ 18)	
NCRT		Number of CRT plots of mass flux along r = constant mesh circles	
ND		Number of size groups used to define an element's spray droplet size distribution (≤ 12)	-
NEL		Number of injection elements (≤ 60)	-
NGF		Number of fuel spray size groups in each stream tube $(1 \leq NGF \leq NGT)$	-
NGT		Total number of propellant (fuel+oxidizer) spray size groups in each stream tube (≤ 12)	-
NLSPEC		Number of different element specifications (i.e.. distinct type and design) (≤ 10)	-
NMR		Same definition as NCSTR, but for STC input data	-
NMSTI		Maximum number of complete or partial passes (i.e.. marching from ZSTART or ZPVSR through the ~oat) in multiple stream tube analy: (Usually ≤ 3)	-
NØZØN		Number of annular zones for subdividing mesh points, excluding those at the wall, to be combined into stream tubes $(< NRWALL$, but usually not larger than 4) OPTION: If=1, one ST per radial mesh arc	-
NP		Total number of z-planes between z=ZSTART and nozzle throat. inclusive (≤ 300)	-
NPROP1,2(I) $1 \leq I \leq NLSPEC$		The index (= 1 for fuel, = 2 for oxidizer) of the propellant flowing through orifice no. 1,2 of Ith element specification	-

TABLE 2 (CONT'D)   1-1-74

Fortran Variable Name	Math Symbol	Definition	Units
NRBAFL,R		Number of circumferential mesh lines inter-sected by a radial baffle along the left, right boundary mesh line	-
NRML		Number of circumferential (constant radius) mesh lines (≤ 20)[1]	-
NRWALL		Number of circumferential mesh lines to the chamber wall	-
NSSTI		Maximum number of complete passes, marching from z = ZSTART to throat, in single tube analysis	-
NST		Number of stream tubes(≤ 40, but usually 10-20)[2]	-
NSTPZ		Number of stream tubes per zone(Usually 4 to 10)	-
NTHML		Number of radial (constant θ)mesh lines(≤ 20)[1]	-
NTHL,R		Index of the radial mesh line which forms the left, right boundary of chamber slice analyzed	-
NTK		Number of temperatures at which propellant vapor specific heats and film thermal conductivity are tabulated (≤ 20)	-
NTYPE(I) $1 \leq I \leq NLSPEC$		Element type index for Ith element specification ($1 \leq NTYPE \leq 10$)as numbered on page 70.	-
NUG		Index specifying uniform (NUG>0) or non-uniform (NUG = 0) gas velocities for gas/liquid elements	-
PCI	$P_c(z_0)$	Static pressure at z = ZSTART	psia
PCTBL		Percent (or decimal) of total propellant flow rate to be assigned to a chamber wall boundary layer stream tube	-
PRPR1,2		Propellant properties grouping $(\mu\sigma/\rho)$ for propellant 1 (fuel), 2 (oxidizer)	$\dfrac{ft\text{-}lb_f}{sec}$
PSTREC(I) $1 \leq I \leq NLSPEC$		Liquid injection post recess for a coaxial jet (Type 6) element	in
RADE(I) $1 \leq I \leq NEL$	r_E	Radial coordinate of the Ith element's origin (e.g., impingement point)	in

1. These limits are imposed by the plotting routine PLØTC.
2. The relationship NST = 1+NØZON*NSTPZ holds if stream tubes are initialized via option (2), page 9.

TABLE 2 (CONT'D)

Fortran Variable Name	Math Symbol	Definition	Units
RHØG	ρ_g	Combustion gas density at nominal chamber conditions	$\dfrac{lb_m}{ft^3}$
RHØLF,Ø		Liquid fuel, oxidizer density at saturation temperature corresponding to P_c	$\dfrac{lb_m}{in^3}$
RHØNBF,Ø		Liquid fuel, oxidizer density at normal (1atm.) boiling point	$\dfrac{lb_m}{in^3}$
RMNØM		Nominal bulk mixture ratio	--
SA1,2(I) 1≤I≤NLSPEC		Spray coefficient a for orifice no. 1,2 for Ith element specification	--
SB1,2(I),		Ibid, coefficient b	--
SC11,2(I) to SC61,2(I) 1≤I≤NLSPEC		Ibid, coefficient C_1 to Ibid, coefficient C_6	--
SMRG(J) 1≤J≤NST		Stream tube gas mixture ratio at z = ZSTART	--
SNN(J) 1≤J≤NST		Number of LISP mesh points combined to form a stream tube	--
SPEL(I) 1≤I≤NLSPEC	X	Spacing between doublets of a like-doublet-pair (Type 3) element	in
SPFAN(I) 1≤I≤NLSPEC	Y	Spacing between doublet's spray fans of a like-doublet-pair (Type 3) element	in
SR(J) 1≤J≤NST		Stream tube mean radius	in
STH(J) 1≤J≤NST		Stream tube angular position (set = 0 in subroutine SCRMBL)	radians
TBF,Ø		Fuel oxidizer droplet saturation temperature at P_c	°R
TCØNVF,Ø(I) 1≤I≤NTK	k	Thermal conductivity of vapor/gas film surrounding fuel, oxidizer droplets	$\dfrac{BTU}{ft\text{-}sec\text{-}°R}$
TCRITF,Ø	T_{cr}	Fuel, oxidizer critical temperature	°R
TGAM(I) 1≤I≤NMR	γ	Combustion gas specific heat ratio (frozen) at TMR(I)	--
TGM(I) 1≤I≤NCSTR	γ	Same as TGAM	

TABLE 2 (CONT'D)

Fortran Variable Name	Math Symbol	Definition	Units
TGØ(I) $1 \leq I \leq NCSTR$	T_o	Combustion gas stagnation temperature at TMR(I)	°R
THETAE(I) $1 \leq I \leq NEL$	θ_E	Angular coordinate of Ith element's origin (e.g., impingement point)	degree
THETAR		Angular coordinate of farthest-right radial mesh line	degree
TMR(I) $1 \leq I \leq NCSTR$, or $1 \leq I \leq NMR$	c	Combustion gas mixture ratio array, defined as the flowrate ratio of propellant 2 (oxidizer) to propellant 1 (fuel)	–
TMU(I) $1 \leq I \leq NCSTR$	μ	Combustion gas viscosity at TMR(I)	$\frac{lb_m}{ft\text{-}sec}$
TMW(I) $1 \leq I \leq NCSTR$ or $1 \leq I \leq NMR$	M_w	Combustion gas molecular weight at TMR(I)	$\frac{lb_m}{mole}$
TNBF,Ø		Fuel, oxidizer normal boiling temperature	°R
TPRØP1,2		Liquid fuel, oxidizer temperature used for initial vaporization calculation. Same as TBF,Ø is recommended but injection temperature may be used	°R
TTØ(I) $1 \leq I \leq NMR$	T_o	Same as TGØ	°R
TVF,Ø(I) $1 \leq I \leq NTK$		Temperature at which CPVF,Ø(I) and TCØNVF, Ø(I) are tabulated	°R
TVIS(I) $1 \leq I \leq NMR$		Same as TMU	$\frac{lb_m}{ft\text{-}sec}$
WØT1(1. $1 \leq I \leq NLSPEC$		Propellant 1 (gaseous fuel) flowrate for gas/liquid element specifications (Type 6 or 10) or flow for orifice 1 of a Type 9 element	$\frac{lb_m}{sec}$
WØT2(I) $1 \leq I \leq NLS$		Propellant 2 (liquid oxidizer) flowrate for gas/liquid element specification (Type 6 or 10)	$\frac{lb_m}{sec}$
WTMLLF,Ø		Molecular weight of liquid fuel, oxidizer	$\frac{lb_m}{mole}$
WTMLVF,Ø		Molecular weight of vapor fuel, oxidizer	$\frac{lb_m}{mole}$

TABLE 2(CONT'D)

Fortran Variable Name	Math Symbol	Definition	Units
W1F,Ø		Lower limit of fuel, oxidizer flux contours)	Program
W.2F,Ø		Upper limit of fuel, oxidizer flux contours)	finds
W1FF		Lower limit of fuel fraction contours (=0.))	limits if both
W2FF		Upper limit of fuel fraction contours (=1.))	upper and lower
W1T		Lower limit of total flux contours)	are entered
W1T		Upper limit of total flux contours)	as zeroes
ZE(I) $1 \leq I \leq$ NLSPEC	z_E	Axial coordinate of element's origin (e.g., impingement point) for Ith specification	in
ZØM	z_0	Axial coordinate of first plane in which spray mass fluxes are calculated	in
ZØM2,3		Ibid, second, third planes	in
ZSTART		Axial coordinate of stream tube initialization plane	in

1. ZSTART may or may not be equal to the last ZØM analyzed by LISP, as discussed on page 84.

In Table 1, the type of FORTRAN input (integer, floating point decimal, alpha-numeric) and the subdivision of each card's first 72 columns into fields is indicated by its noted format. For decimal data, six 12-column fields are used; for integer data, either twelve six-column fields or six 12-column fields are used.

Standard FORTRAN input formats are used. Specifically used are:

Comment Cards (A-format)   18A4

Integer Variables Beginning with Letters I thru M   6I12(or 12I6)
  (No decimal points, 12 (or 6) space field widths,
  last digit in last space of field, 6 (or 12
  consecutive values per card until READ statement
  is finished)

Decimal Variables Beginning with Letters Other Than I thru M   6E12.8
  (Use decimal point or account for implied
  decimal location, one value every 12 spaces,
  6 consecutive values per card)

The "Description" column in Table 1 gives the FORTRAN code names of input variables as they appear in the program listing. One value is to be entered for each coded variable unless it is subscripted. Array sizes for subscripted integer and decimal variables are indicated in this column below the variable names. (With A-format data, the subscript ranges are not indicated; 18 fields per comment card are implied.) For most of the data, all the values of one variable are read before proceeding to the next variable. (Exceptions are in the last group of stream tube initialization data which is part of the deck normally punched by STC's subroutine STAPE. The particular form used there permits use of very compact input/output statements and results in compact data decks, but is admittedly somewhat inconvenient for manual generation of data decks.)

3.3.1 Selection of LISP Input Data

Organization of the LISP input data in Table 1 is, roughly, as six blocks of cards. The first block consists of two comment cards, in alphanumeric(A) format, which permit the user to document the case with such information as

63

injector name and/or drawing number, propellant combination, nominal chamber pressure and mixture ratio, date of the computer run, etc. The information provided on these cards is printed out with certain data printout headings and appears on computer plotted graphs.

The second block of cards contains the description of the geometry of the system (number of elements, mesh lines to be set up, location of wall and baffles, if any, etc.), liquid propellant properties, and controls for the desired calculations and for CRT graphical output.

The third block contains descriptions of common characteristics of groups of elements which make up the injector. Each of these groups is assigned an element specification index LSPEC which ranges from unity to NLSPEC. Data included are the element type (doublet, triplet, etc.), orifice diameters and discharge coefficients, propellants issuing from the orifices, and geometric specifications, such as the axial coordinate of the impingement point and orifice alignment angles.

The fourth block of data, which contains the information specific to indi- vidual elements, consists of NEL cards where NEL is the number of elements considered in the analysis. Each of the cards lists the specification group LSPEC which describes the given element, the radial and angular coordinates of the element's origin and an element orientation angle, ALFA, explained later.

The fifth block of data, input if contour plots are called for, is concerned with the upper and lower limits of the contour lines.

The sixth block of cards contains tables of the equilibrium combustion gas properties for the propellant combination, In the order of their read-in, they consist of the mixture ratio, the gas stagnation temperature ($^\circ$R), the gas viscosity (lb_m/ft-sec), the ratio of specific heats, the mean molecular weight (lb_m/mole), and the characteristic exhaust velocity (ft/sec). A minimum of two and a maximum of eighteen mixture ratios may be put.

A functional, rather than a card-by-card approach is taken in the following discussions. Reasonably self-explanatory data are not commented upon.

3.3.1.1 Geometric Considerations. Cards 2030 through 2060 contain a number of control integers and other data pertaining to the portion of chamber cross-section to be analyzed using an appropriate system of mesh points. LISP is structured to take advantage of injector symmetry, so that the data selection consists, briefly, of choosing a pie-shaped sector of the injector bounded by either planes of symmetry or baffles, such as shown in Figures 6 and 7, and 8 overlaying this sector with a grid of equally spaced radial and circumferential mesh lines. In both Figures 6 and 7, the area bounded by AOC becomes a satisfactory segment for analysis. LISP is dimensioned for up to 400 mesh _points_ which permit the choice of combinations of as many as 20 x 20 or 25 x 16 mesh _lines_ if desirable. Similarly, LISP is also dimensioned to describe up to 60 injector elements.

It is often appropriate in LISP to employ additional mesh lines and injector elements _outside_ the chamber slice under analysis. The spray flux within the slice to be analyzed includes spray from nearby injector elements when the sides of the slice are planes of symmetry; spray from interior elements will collect on the baffles when they constitute the side walls; and some spray will impinge on the walls of the thrust chamber ahead of the plane being analyzed and run down the walls to add to the spray directly impinging on wall mesh points in the plane. To treat these conditions, a number of options are included in LISP. The options can be described by reference to the sample mesh line system depicted in Fig. 8, which represents an overlay of a mesh line system on the injector segment of Fig. 7. In this illustration:

1. NRML* is the number of circumferential mesh lines.
2. NRWALL is the number of circumferential mesh lines to the chamber wall. When (as in Fig. 8) NRWALL is less than NRML, the spray mass to the mesh points beyond NRWALL is folded into the wall at the corresponding θ-location.

---

*The nomenclature for FØRTRAN variables is given in Table 2

65
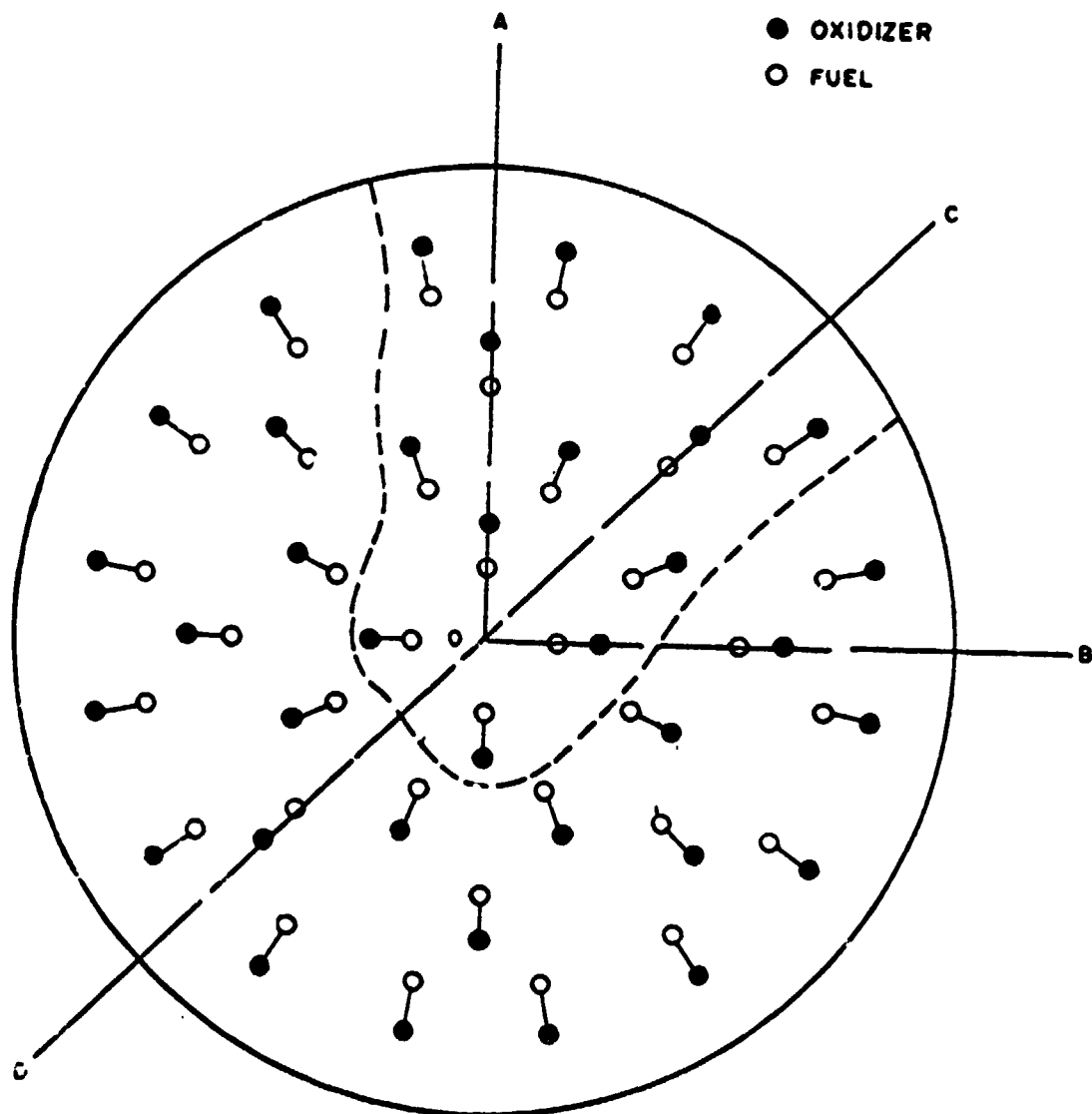
Figure 6. Se. ction of Segment of Injector for Analysis
of Spray and Combustion Gas Flow Fields –
Segment Bounded by Planes of Symmetry
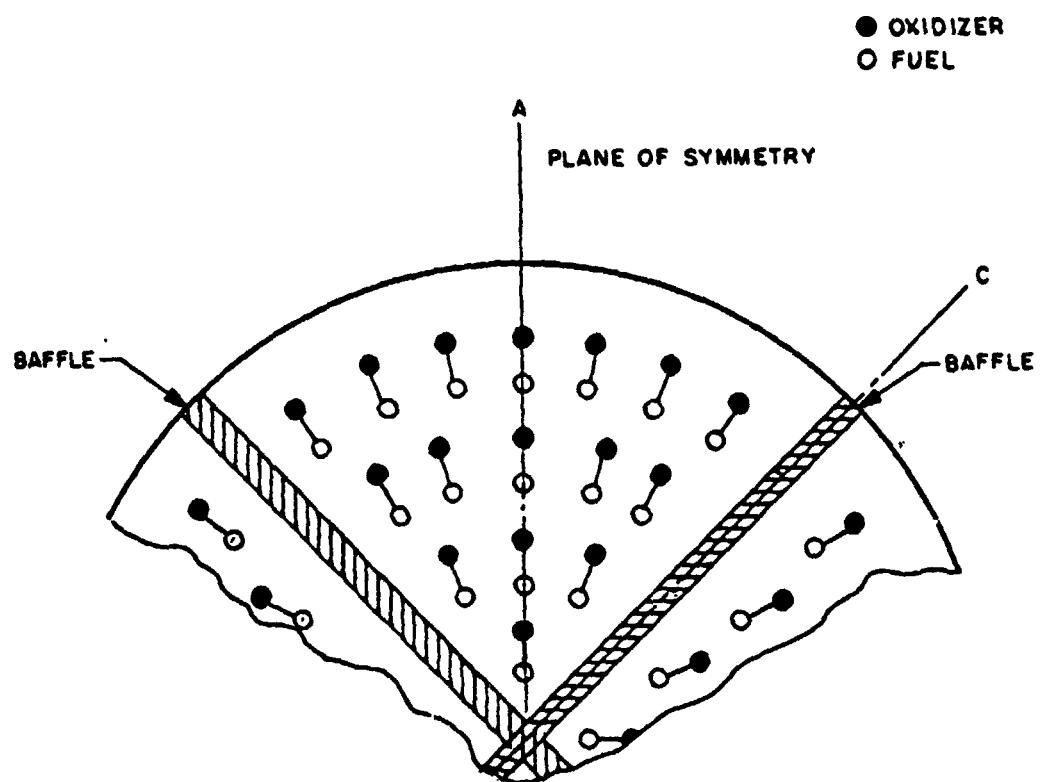
Figure 7. Selection of Segment of Injector for Analysis of
Spray and combustion Gas Flow Fields – Segment Bounded
by Combination of Planes of Symmetry and Baffles
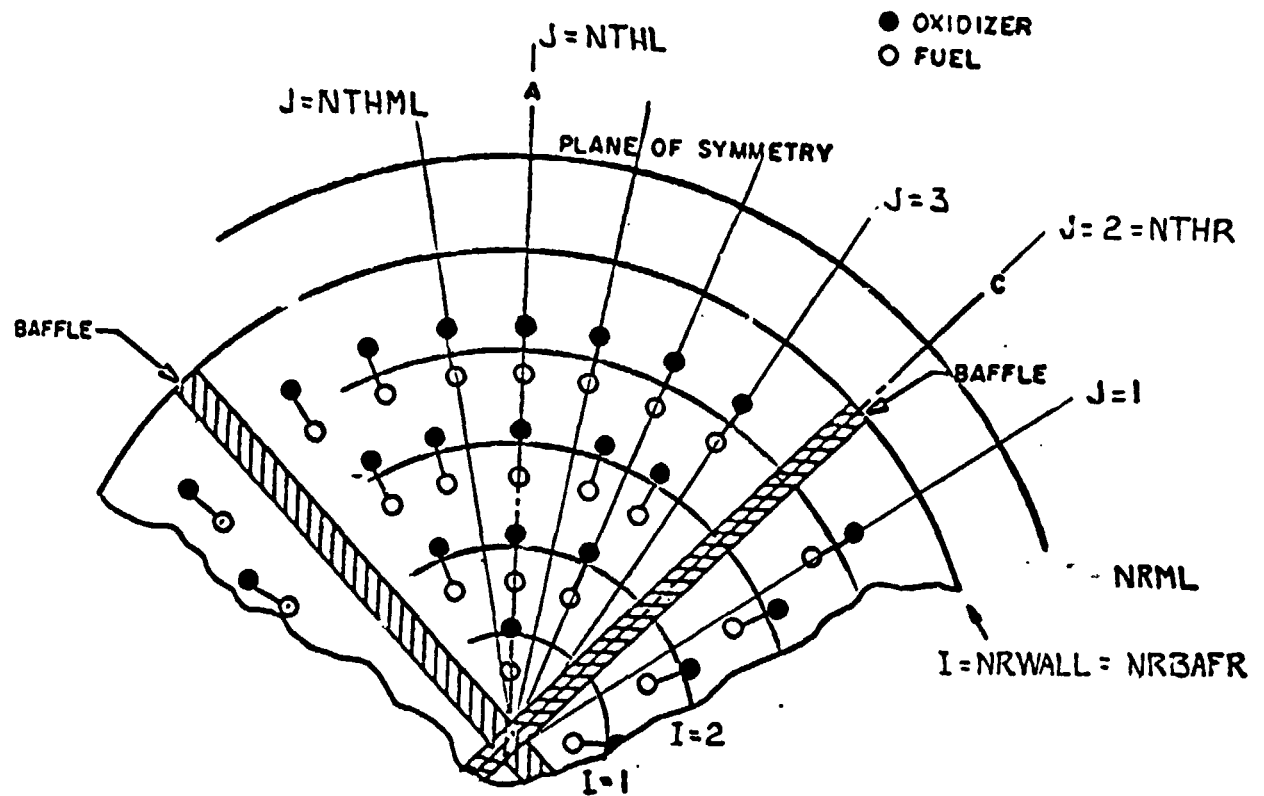
Figure 8. Overlay of System of Mesh Lines Upon Injector
          Slice to Permit Calculation of Spray Flux
          Distribution by LISP

3. NTHML is the total number of radial mesh lines. The radial mesh lines are indexed J=1 to J=NTHML counterclockwise (i.e., from right to left).

4. NTHR and NTHL are the radial mesh lines which define right and left boundaries of the chamber slice to be analyzed. It should be noted that additional radial lines are defined (in this case) to the right and left of NTHR and NTHL, respectively.

5. Because a baffle corresponds to the right boundary of the slice, a code number NRBAFR is input to LISP, which corresponds to the number of circumferential mesh lines (starting from the center) over which the baffle extends. (Since the baffle extends to the wall in Fig.8 , NRBAFR is equal to NRWALL in this example.) The spray flux to mesh points to the right of NTHR will be folded into NTHR at the corresponding radial locations. Similar results can be produced for baffles on the left boundary of a slice by the code number NRBAFL. The baffles must begin at the center of the chamber but they need not extend to the wall, i.e., NRBAFR and NRBAFL may be less than NRWALL.

6. The best a · .   .ierred way to account for the fact that the spray inside a slice may include contributions from nearby elements outside the slice is simply to include these elements in the input specifications. For example, in Fig. 6, specifications for all elements within the dotted lines would be made in the analysis of the slice AOC.

7. The procedure described in (6) above is the recommended method accounting for spray travel across slice boundaries defined by planes of symmetry; however, alternate methods are provided in LISP by the FØRTRAN code variable JSYM. When JSYM=0, the procedure in (6) above is employed. When JSYM is defined as either 1 or 2, only injector elements within the area encompassed by radial mesh lines NTHR and NTHL are input as data, but the spray from the elements to mesh points outside the area are folded back into the corresponding mesh points inside the area. When JSYM=1, this folding is done on the basis that the elements to the right and left of both the boundaries NTHR and NTHL are mirror images of each other. In this case, the spray flux to mesh line (NTHR-1) is folded back into mesh line (NTHR+1); similar relations hold around boundary NTHL. If the elements in and around slice AOC are related as repeating

sets, JSYM=2 is defined and the folding of flux from exterior to interior mesh lines is done acco*ding to (NTHR-1) folds into (NTHL-1), etc.

In analyses of several injector designs at Rocketdyne, it has been experienced that choosing JSYM=0 offers fewer "traps" in setting up a problem, particularly where the geometric distribution of injector elements may require assumptions of effective planes of symmetry which are not rigorously so. Although many planes of symmetry may be defined for an injector, analysis of the gas flow within a given chamber slice will give meaningful results only if the injection elements are specified such that the correct geometric proportions of total propellant flows enter the slice; otherwise, erroneous mass flux levels and chamber pressures will be calculated. For JSYM=0, simply defining enough elements outside the chamber slice takes care of this. For JSYM=1or 2, elements that fall on (or very near) the boundaries should be examined carefully and their hole sizes or flow rates may have to be adjusted artificially to ensure proper flows.

3.3.1.2 Injection Element Considerations. The current version of LISP permits consideration of the following ten types of injector elements:

1) Unlike doublet
2) Like-doublet
3) Like-doublet-pair
4) Triplet
5) Four-on-one
6) Gas/liquid coaxial jet or concentric tube
7) Showerhead*
3) Special callout by general spray flux equation
9) Gaseous transpiration cooled injector face
10) Gas/liquid triplet

---

*Distribution coefficients have not yet been evaluated for the showerhead element, which must currently be treated as a Type 8 element.

Element Specifications. Card 2030 carried a value of NLSPEC, the number of element specifications which may range from 1 to 10. If all elements are alike, differing only in their position and orientation on the injector face, a single specification is needed (NLSPEC=1). Any element or group of elements that differs from the others in element type, element design (e.g., orifice sizes, discharge coefficients, impingement angle and distance), propellant assignment to the orifices or element angular alignment (other than a simple rotation about its z-axis) requires a separate element specification.

Element spe ification data are entered in cards 2x10 through 2x95 where the x denotes eac' specification number. Mass flux distributions calculated by means of Eq. (1) and (2) make use of correlation coefficients built into function subprograms in terms of such parameters as the element types, orifice sizes, stream impingement angles and momentum ratios, etc., for Type 1 through 6 elements. The correlation coefficients must be supplied by the user for Types 8 and 10 elements. For the first five element types listed above, the element flow rates of each propellant are computed within the LISP subprogram EFLOW in terms of the element orifice sizes, discharge coefficients, and the injection Δp's of the separate propellants. For type 6 and 10 gas/liquid elements, propellant flowrates are supplied as input data. Mean drop sizes for propellant sprays may be supplied directly as input data for all elemental types or, in the case of the first five element types, may alternatively be calculated within the LISP subprogram DSIZE.

If an element or group of elements is defined as being type 8, the mean droplet size is read in as input data and the total flow from the element is calculated as if the element were a doublet with equivalent orifice diameters. The mass flux distribution is calculated by means of Eq. (2) using spray distribution correlation coefficients supplied directly as input data rather than being supplied by subprograms within LISP. This feature permits a designer to have a cold-flow characterization made of the single element (or elements) to be incorporated into a prospective injector and then to employ the correlated spray coefficients from the cold flow experiment in LISP. Such a procedure is useful .ot only for the situation where spray coefficients have not been determined previously for th? intended elements, but it also permits the designer to acco , :or factors such as short L/D orifices and manifold cross-flows in his

71

single element cold-flow experiments. A Type 8 callout may also be used to lump together into a single effective source a number of elements near the center of a large injector.

In the case of a transpiration-cooled injector face, the Type 9 element flow rate is read-in and is divided by the cross-sectional area of the chamber slice to define a uniform mass flux at all r, θ mesh points.

Single Element Geometry. LISP converts the (r, θ, z) coordinate system of the thrust chamber into individual rectangular (x, y, z) coordinate systems for each element based upon particular origins of the elements. Element coordinate systems are illustrated in Fig. 9. For the liquid/liquid doublet and triplet elements, the x axis corresponds to the long axis of the spray fan formed by the impinging streams while the y axis lies in the plane of the impinging streams. For the 4-on-1 and like doublet pair elements, the geometries shown in Fig. 9a were made as consistent with the doublet and triplet as was conveniently possible. The z-axis origins for liquid/liquid elements are at the element impingement points and, so far as input data are concerned, are always directed along the bisector of the stream impingement angle. (For unlike doublets, the z-axis is changed internally by LISP to lie along the resultant spray momentum vector.) The definitions cf x and y axes for the type 8 element are equivalent to those specified for the liquid-liquid unlike doublet but the z-axis is not altered to follow the momentum vector. From symmetry, the definition of x and y axes for showerhead elements is immaterial unless they are canted with respect to the chamber axis.

The coordinates for gas/liquid triplet and coaxial jet elements are illustrated in Fig. 9b. The triplet definitions parallel those for the liquid/liquid triplet, with an origin at the geometric impingement point. Coaxial jet elements are axisymmetric so that, like the showerhead, definition of x and y axes matters only if an element is canted with respect to the chamber axis. The origin for the coaxial jet lies on the element centerline and at the discharge end of the liquid (oxidizer) injection tube (post). If there is a "post-recess," the variable z_E will be negative.
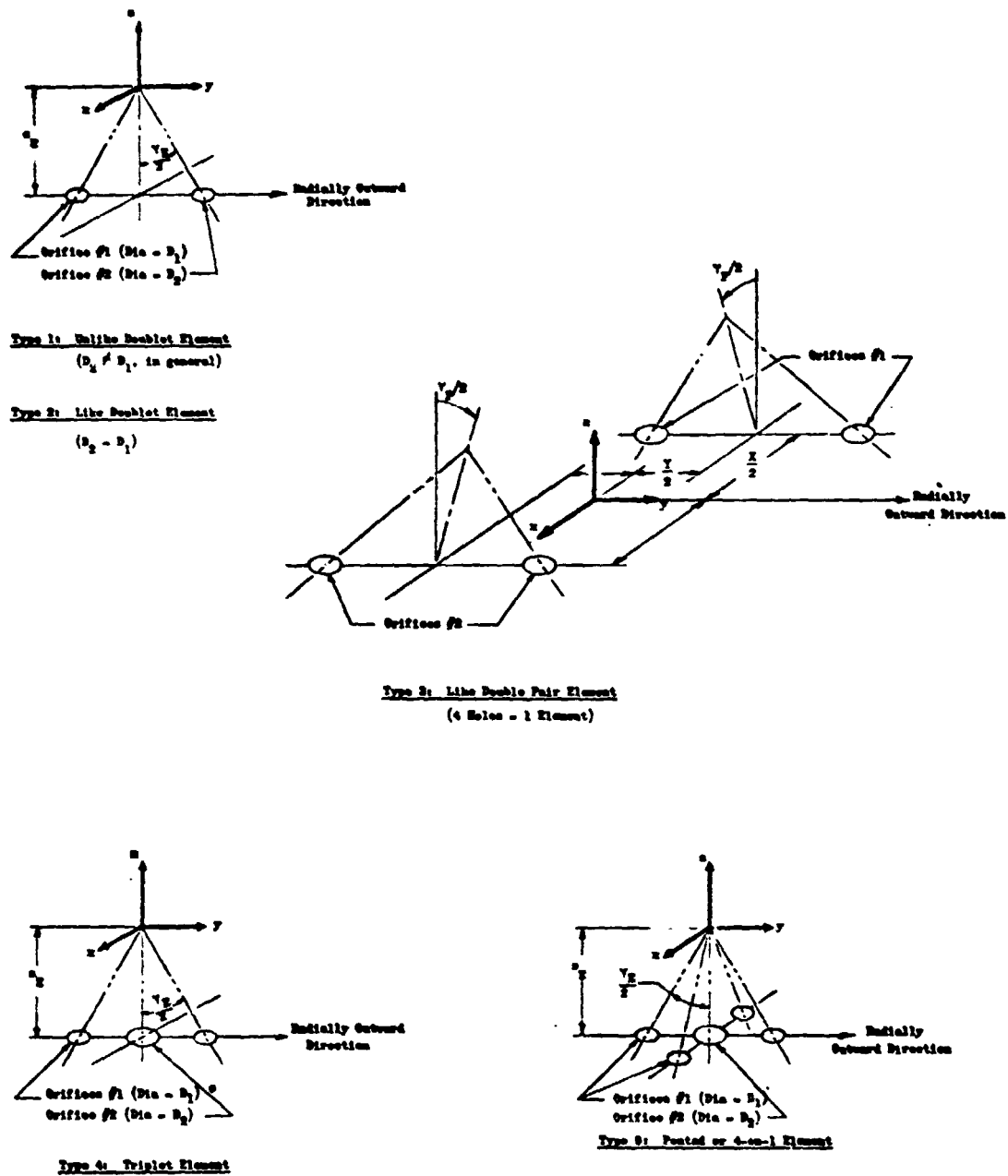
Figure 9a. Liquid/Liquid Injection Element Designations
and Coordinate Systems

Type 6: Coaxial Jet Element

Note: D_2 = Central Tube I.D.

A_1 = Annular Passage Area

Type 10: Triplet (Note Reversal of
Orifice Numbers from Liquid/Liquid Triplet)

Figure 9b.   Gas/Liquid Injection Element Designations
and Coordinate Systems

LISP considers both the liquid and gas injected with a gas/liquid element to be distributed as if they flowed as rays from point sources. Rather than lying at the element's coordinate origin, as with the liquid/liquid elements, the point sources are located at distances of $-\delta_L$ and $-\delta_G$ upstream of the origin for the liquid and gas. respectively. These parameters are derived during correlation of elemental cold-flow distribution data and values must be read-in for each gas/liquid element.

The orientation of the x. y, z coordinate system of an individual element to the chamber coordinate system is defined in terms of three rotation angles whose respective Fortran names are ALFA, BETA, and GAMMA. If:

1. The element is oriented on the injector such that its y axis (as defined by Fig. 9) is coincident with the chamber radius through the element impingement point, with positive y pointing radially outward, and

2. The z axis of the individual element (as defined in Fig. 9) is parallel to the thrust chamber axis,

then the angles ALFA, BETA, and GAMMA have zero values. If the element is not oriented in this "basic" or reference alignment, then

1. ALFA is the counterclockwise angle of rotation around the basic z axis of the element's y axis from its original alignment with the chamber radius,

2. BETA represents counterclockwise rotation around the y axis (in its new position after the first rotation); finally,

3. GAMMA represents counterclockwise rotation around the x axis (in its transformed position after the first two rotations).

For each rotation, "counterclockwise" implies the rotation direction that would be seen by an observer looking along the positive axis toward the origin of the element. In most applications, no more than one rotation will be applied to the geometry of any single element.

Each element, except for Types 7 and 9, is considered to consist of two equivalent orifices designated as 1 and 2. The physical orifices which correspond to these number designations are listed in Table 3 for all element types.

73

The LISP program also labels the injected propellant as either 1 or 2. The fuel must be chosen as Propellant 1 and the oxidizer ..s Propellant 2 because of the combustion gas properties vs mixture ratio tables and because they are expected to bear these designations when data are transferred from LISP to STC. Also, for a gas/liquid propellant combination, the gaseous propellant is always assumed to be the fuel.

Propellant Flowrates. Except for injector elements defined as Types 6, 9 or 10, the weight flow lb_m/sec) of each propellant from each orifice of the element is calculated in LISP by means of the standard orifice equation. The LISP user must input the injection density (lb_m/ft^3) and the injection pressure drop (psi) for each propellant as well as the diameter and discharge coefficient for each orifice of the element. If the injector element is specified as Type 8, then the user must specify the diameter of the circle of equivalent area as the orifice diameter because LISP treats the Type 8 element as an unlike doublet.

The Type 6 gas/liquid coaxial jet element flowrates are defined as follows. Liquid oxidizer flow is calculated from element specification data on post diameter, DIA2, discharge coefficient, CDDIA2, and injection pressure drop, DPINJ2. The gaseous fuel flowrate, WØT1, is read-in along with the actual cross-sectional area, AØ1, of its annular injection passage.

Table 3. Physical Orifices Corresponding to Equivalent
Orifice Designations One and Two for the
Element Types Defined in LISP Program

Element Type	Equivalent Orifice 1	Equivalent Orifice 2
1 (Unlike Doublet)	The orifice lying on the negative y axis of Fig. 9 which injects toward the positive y axis. It is the fuel orifice of Fig. 6 (where the r coordinate corresponds to the y axis).	The orifice lying on the positive y axis of Fig. 9 which injects toward the negative y axis. It is the oxidizer orifice of Fig. 6.
2 (Like Doublet)	Same as for the Type 1 element; however, the 1 and 2 designations are not important for the Type 2 element.	Same as for the Type 1 element.
3 (Like-Doublet-Pair)	The pair of ports to the left of the y-axis of Fig. 9.	The pair of ports to the right of the y-axis of Fig. 9.
4 (Triplet)	The two outer impinging orifices of Fig. 9.	The inner orifice of Fig. 9.
5 (4-on-1)	The four outer impinging orifices of Fig. 9	The inner impinging orifice of Fig. 9.
6 (Gas/Liquid Coaxial)	The annual orifice through which gas flows.	The inner post orifice through which liquid flows.
7 (Showerhead)	The single port.	Not defined.
8 (Special Callout)	Treated as unlike doublet.	Treated as unlike doublet.
9 (Transpiration Face)	The entire cross-sectional area.	Not defined.
10 (Gas/Liquid Triplet)	Same as Type 4	

75

Discharge coefficients of 0.84 are recommended for liquids flowing through short L/D orifices under motor firing conditions and 0.80 under cold flow condition. However, the user is free to use whatever coefficients he deems appropriate. He may, for example, use different coefficients within an element to account for effect of manifold cross flows in the injector, or use different coefficients from element to element to approximate injector maldistribution effects.

Spray Drop Sizes. If the injector element is specified as being Type 1 through Type 5, LISP will calculate a mass median drop diameter for the propellant of each orifice of the element. These calculations are based upon the correlations of Dickerson et. al. (Ref. 12) derived from hot wax experiments. Constants in the correlations have been modified to give characteristic diameters which make calculated c* efficiencies compatible with measured results for three injectors tested, analyzed and reported in Ref. 5. Alternatively, by assigning a value greater than zero to the flag IDBAR, the LISP user may assign his own estimation of drop diameter to the flow from each orifice of a given element. For elements defined as Types 6 through 10, the user always supplies his estimation of a characteristic drop size for each orifice of each element. The appropriate mean droplet diameter is the mass median diameter.

With elements of Types 2 and 3, the mean drop diameters are based on the empirical correlation of Falk, etc. (Ref. 13), modified to make c* efficiencies calculated by the STC computer program correlate with experimental data reported in Ref. 13. Additional input data are required for these elements: combustion gas density (lb_m/ft^3) at injection-end chamber pressure, chamber contraction ratio and, for each propellant, the liquid physical properties grouping $\mu\sigma/\rho$ entered as the parameters PRPR1 and 2 on card 2080. Here ρ is density (lb_m/ft^3), σ is surface tension (lb_f/ft) and μ is viscosity ($lb_m/ft\text{-}sec$), all evaluated at injection conditions.

Propellant and Combustion Gas Properties. The densities of both propellants at their injection temperatures and at their saturation temperatures corresponding

to the expected chamber pressure are required* (Cards 2070 and 2080). The former are used to calculate injection flow rates and momenta; the change in density to the latter are used in defining the output droplet diameters.

Propellant temperatures (Card 2090) are used in estimating an appropriate value of the evaporation coefficient, $C_{k'}$. Normally, these should be supplied as the saturation temperatures corresponding to the expected chamber pressure.

The required combustion gas properties are supplied as tables which relate the given gas properties to mixture ratio at assumed chemical equilibrium conditions. The tabular property data supplied are lists of the stagnation temperature,** viscosity, molecular weight, specific heat ratio (γ), and characteristic exhaust velocity (c*) corresponding respectively to a list of mixture ratios. These data are given on Cards 4110 through 4630. These combustion gas property tables have been set up so that essentially the same set of punch cards may be used again as STC input data.

The number of entries in the gas property tables are defined by the control variable NCSTR. A maximum of eighteen mixture ratios may be used in the tables which should range from essentially all fuel vapors (zero or near zero mixture ratio) to essentially all oxidizer vapors (a mixture ratio of at least 100) to prevent possible extrapolation errors when local regions of extreme propellant maldistribution car encountered in LISP analyses. If the LISP user is interested in calculating only the spray mass distribution for a given injector configuration, then only two entries (one punch card) are required for each of the six gas property tables listed above (NCSTR=2).

---

*In the case of a gas/liquid propellant combination, the density of the gaseous propellant is supplied at its injection temperature and the expected chamber pressure. An error in the estimation of chamber pressure will effect only initial vaporization and dropsize and will not effect the definition of propellant flow rate.

**The required properties may be obtained, for example, from runs of the JANNAF ∅DE (one-dimensional kinetic) computer program (Ref. 14) or an equivalent program (e.g., Ref. 15)

Partial propellant evaporation between its injection and the LISP collection plane is approximated by Eq. (3). The user needs to concern himself with a value of $C_{k'}$ for each propellant. One method is for the user to input values of $C_{k'}$ for the variables CKP1 and 2 on Card 2070; the obvious implication is that he knows appropriate values. If that is not the case, another method has been built into LISP: values of $C_{k'}$ are calculated internally if the user inputs values of latent heat of vaporization for CKP1 and 2. The program distinguishes between these methods by examining the magnitudes of the input values: if both are larger than unity, the values are interpreted as heats of vaporization. Conversely, if either is less than unity both are retained as the modified evaporation coefficients.

If CKP1 and 2 are heats of vaporization, $C_{k'}$'s are calculated in subroutine CCKP. An empirical scale factor has been applied to the theoretical values to force agreement between LISP's calculated percentages burned at $z_0=2$ and those observed in an NTO/N_2H_4-UDMH (50-50) engine experiment (Ref. 5).

Numerical values of $C_{k'}$ will ordinarily range from approximately 2×10^{-4} to 2×10^{-3} in^2/sec. Typically, they are on the order of 1/5 to 1/3 the values computed later, in STC, for k_s' for the same propellants. This is appropriate because $C_{k'}$ is being applied, in LISP, to a spray that is incompletely atomized and whose droplets are not yet heated to their saturation temperatures over much of the travel distance considered.

CRT-Plotting Options. Input data concerning computer generation of data plots are scattered throughout the LISP input data. A non-zero value of NCRT on Card 2030 will result in a cross-sectional plot of the LISP mesh system, with all element origin locations denoted. If NCRT is larger than zero, that number of mass flux profile plots will be produced, each showing the fuel and oxidizer fluxes vs angular position, θ, at a fixed value of chamber radius. The radial coordinates for these plots are prescribed through the variable array IRCRT on Card 2050 (et seq.). Each value of ICRT denotes the number of a circumferential mesh line, counting from the center of the chamber out.

Contour plotting is controlled by the values of the variables KFCRT through KFFCRT on Card 2040. A non-zero value for any of these variables results in one

contour plot. A positive value denotes the number of contour lines which are to be plotted; values between 10 and 15 ordinarily produce pleasing graphs. A negative value denotes only approximately the number of contour lines; the actual number is determined internally by selecting a rounded-off contour interval. Use of a negative value is recommended.

The maximum and minimum ranges of contour plotted variables can be controlled by input data on Cards 4010 and 4020. (These cards are needed if any of KFCRT through KFFCRT are non-zero.) If limit values are input as zero, contour scaling is done internally. Ordinarily, one would not know, a priori, appropriate ranges for mass fluxes, so zeros would be entered for W1F, $\emptyset$ and T and for W2F, $\emptyset$ and T. On a later run, it might be desired to examine some contour interval in more detail by excluding some higher or lower flux values. Ordinarily, the modified fuel fraction is restricted to the range W1FF=0, W2FF=1.

Subscripts. The variables which appear in the LISP calculations are subscripted according to either injector element index (1-60), combustion zone mesh point index (1-400), propellant index (1-2), orifice index within a specific element (1-2), or element specification set (1-10). Of these indices, the element index, the mesh point index and the element specification indices become the regular Fortran integer indices I, J, etc., for use in D$\emptyset$ loops and in READ and WRITE statements. The remaining subscripts are incorporated directly into the Fortran variable names. For example, the weight flow of Propellant 1 from Orifice 2 of injector element 3 would appear in the Fortran code as W$\emptyset$T21(3) and spray flux of Propellant 1 from all elements to mesh point 22 would appear as the Fortran variable STW1(22). Interpretation of the program listing in Volume 2 and the input variable designations included in Tables 1 and 2 should be made in this context.

Multiple Collection Plane Analyses. By specifying non-zero values for Z$\emptyset$M2 (and Z$\emptyset$M3), spray mass distributions will be calculated by LISP at a second (and third) collection-plane location. Data transferred via scratch data unit 2 to STC are from the last collection plane analyzed.

79

3.3.2 Selection of STC Input Data

Input data for the STC subprogram block are read by subroutine CINPUT. Card
sequence numbers begin with 5010, which provides values for some stream tube
initialization control parameters. If ILISP=0, stream tube initial data are
read-in via punched card so the parameters NSTPZ and NUG have no meaning.
In that case, the value of NØZØN determines whether the stream tubes are
analyzed assuming axisymmetry (NØZØN>0) or not (NØZØN=0).

If ILISP≠0, stream tubes are to be initialized from LISP-generated mesh point
flow data. Then NØZØN=0 is forbidden and will cause job termination. NØZØN=1
results in all LISP mesh points on each circle of mesh points (i.e., those
at equal chamber radii) being combined into one stream tube (option 1, page 8)
and NST=NRWALL. If NØZØN>1, its value denotes the number of circular or
annular chamber zones (option 2, page 9) of mesh points and NSTPZ gives the
number of stream tubes per zone. Then, recognizing that the wall mesh points,
at least, are combined into a wall-bounding stream tube, the total number of
stream tubes is NST=NØZØN*NSTPZ + 1.

With ILISP≠0, a non-zero value of NUG causes all of LISP's gasified propellant
flow to be distributed uniformly among the mesh points (i.e., constant gas
flux).

The option to analyze stream tubes by suppressing data concerning geometric
mesh point locations and forming stream tubes entirely according to a mixture
ratio ranking is given by ILISP = 0 and NØZØN = 0. Because LISP will be
bypassed in this case, the portion of the data deck which initializes stream
tube data must be provided (e.g., from a previous run in which LISP was run)

If a non-axisymmetric multiple stream tube analysis is desired, zero values
should be input ILISP and ITRANS. Also, it should be recognized that only one
multiple stream tube analysis will be performed in this case, i.e., there will
be no iterative analysis to converge on a solution satisfying a throat boundary
condition. This does not prevent, however, the punchout of throat-plane data
in NAMELIST form for subsequent input to the improved TDK computer program.

80

<u>Card No. 5020</u>. The variable NP designates the number of z-planes between z_0 and the nozzle throat (inclusive) at which program calculations are to be made. The preferred method is to select a desired magnitude for Δz and calculate NP by:

$$NP = \frac{z_T - z_0}{\Delta z} + 1$$

It is recommended that values of Δz be selected in the range $0.02 \leq \Delta z \leq 0.10$-in. such that there are some convenient integral number per inch for improving printout readability. The other variables on this card are self-explanatory.

<u>Combustion Chamber Geometry (Card No. 5030, et seq.)</u>. The geometry of the combustor is described through the doubly subscripted array APRØF(J,L). This array is entered in matched pairs (L=1,2) for each value of J. The values APRØF (J,1) denote axial distances (in) from the injector and the values APRØF (J,2) denote the corresponding chamber diameters (in) at these positions. It is required that distance increase with increasing J and that the array progress from the injector plane to the nozzle throat. That is, APRØF (1,1) is assumed to be zero and APRØF (NAP,1) is the injector to throat distance. The intermediate values with 1<J<NAP are used to describe the wall profile of an axisymmetric chamber. For z-planes lying between any APRØF (J,1) and APRØF (J+1,1), linear interpolation on diameter is used.

A recommended option is provided for cases employing the common design of a conical nozzle convergent section which is tangent to a radius of curvature through the throat section. This option is invoked by entering the value of the throat radius ratio (wall curvature/throat opening) as APRØF (1,1). Later, in subroutine AVAR, it is reset equal to zero for subsequent interpretation as the injector face plane. Then, the upstream end of the nozzle cone is specified by the next to the last point, APRØF (NAP-1,L), and the nozzle throat by last point APRØF (NAP,L). The nozzle profile between those two points is computed within subroutine AVAR.

There are other options available in subroutine AVAR which are not expected to be useful for analyzing axisymmetric cylindrical chambers so their influences on APRØF (J,L) input are not discussed here.

Multiple stream tube STC analysis is continued past the throat for several z-planes. With the throat plane denoted by Z(NP), mirror-image symmetry is assumed such that areas at Z(NP+1) and Z(NP-1) are equal, etc.

Combustion Gas Properties (Cards No. 5130, 5210-5630). Combustion gas properties, tabulated as functions of gas mixture ratio, are obtained from prior, peripheral computation using a thermodynamic equilibrium performance computer program. Rocketdyne's free-energy-minimization program (Ref. 15) has been used here, but any comparable program is adequate. Data entered in this table are properties for equilibrium combustion products at stagnation conditions corresponding to the mean expected chamber pressure. The parameters are assumed not to vary appreciably with pressure, but are taken to be functions only of mixture ratio.

Propellant Vapor Properties (Cards No. 5710-6230). Tables of fuel and oxidizer vapor specific heat and thermal conductivity as functions of temperature are required, spanning the range of temperatures across the vapor/combustion gas films around spray droplets. At the lower temperatures, specific heats at constant pressure may be conveniently obtained directly from propellant enthalpy tables or charts. At higher temperatures, dissociation is important and it is ap opriate to blend the low temperature, undissociated data into equilibrium dissociation data.

The thermal conductivity needed is not simply that of the vapor, but that of the combustion gas-vapor mixture between a droplet's surface and a surrounding flame-front. Again, a blending between undissociated propellant, dissociated propellant and propellant-rich combustion gases is appropriate at low temperatures with a gradual shift to the conductivity of the combustion gases alone at high temperatures.

These properties are used in subroutine KPRIME to calculate propellant droplet evaporation coefficients, Eq. (23). More detailed discussions of these data concerning selection and the following miscellaneous properties) appear in Reference 16 together with some example data.

82

Miscellaneous Propellant Properties (Cards No. 6510 to 6530). The various temperatures, densities, and molecular weights are explained adequately in Table 2, as are the liquid molecular weights. The vapor molecular weights will normally be lower than those of their liquids; values corresponding to equilibrium dissociation at the mean film temperature are recommended here. Such dissociation is considered, however, to take place far enough from the droplet surface that it has little effect on the thermal gradients there. Hence, it is no longer deemed appropriate to add a heat of dissociation to the heat of vaporization. Similarly, approximating a droplet warm-up effect by adding a liquid sensible heat to the heat of vaporization is no longer recommended, i.e., the value entered for latent heat of vaporization should be just that.

The heats of vaporization are used in calculating evaporation coefficients, Eq. (23). The liquid saturation densities are used in droplet size and behavior calculations and the rest of these properties are used in obtaining mean film viscosities in subroutine EVAPS.

Program Control Variables (Cards No. 6540 and 6550). An averaged, single stream tube analysis of the multiple stream tube input is forced by internal assignment of IST=1, regardless of the value read-in. The main purpose of this is to ensure that the initial-plane chamber pressure (PCI) and total propellant flowrates are compatible with the nozzle throat boundary condition of maximum gas flux. The boundary condition is approximated by requiring that the single stream tube's gas velocity at the throat be equal to calculated sound speed for the combustion gases at the calculated throat mixture ratio. A required flow area is computed from a gas continuity equation; the ratio of this area divided by geometric throat area may differ from unity by CRTØL or less; otherwise, the value of the initial pressure is adjusted and the entire single stream tube analysis is repeated. Convergence of the area ratio to (1.00 ± CRTØL) is required within NSSTI iterations before proceeding to multiple stream tube analysis. Values of CRTØL = 0.005 and NSSTI = 5 are recommended.

Single stream tube data are written out after each of the first four Δz
increments. Thereafter, the data are written out at intervals of IPRSST Δz's
until the neighborhood of the throat is reached. Data are then written out
at every increment. Print-out control in multiple stream tube analysis is
identical except that the intermediate print interval is IPRMST.

Improved computational accuracy may result from repeating the sequential spray
phase/gas phase solutions using updated values of flow parameters, in a
"corrector" cycle. Repetitive corrector cycles may be obtained by means of
the variable ICRC which denotes the number of corrector cycle calculations
to be made in each Δz increment. The recommended value is ICRC = 1.

The FORTRAN variable ARTØLD denotes the decimal tolerance ε_{A_t} discussed on
page 18. A value of ARTØLD = 0.0025 has been used successfully in checkout
analyses.

    Other Data (Cards 6560-6610). The ZSTART plane is used as the initial
plane for vaporization calculations in STC. Its location is not necessarily
the same as ZØM, the LISP spray flux collection plane. The pattern calculated
for the last value of ZØM used in LISP is carried over into STC. Then, if the
value of ZSTART is in the range 0<ZSTART<ZØM, it may differ from ZØM. However,
if ZSTART = 0 or ZSTART>ZØM is entered, it is reassigned as being equal to ZØM.

The variable PCTBL indicates the minimum percent (or, equivalently, decimal
fraction) of the total propellant flow rate to be assigned to the stream tube
adjacent to the chamber wall (if ILISP≠0). A zero value is recommended unless
the STC computed data are going to be used in a wall boundary layer analysis.

The two arrays FRACUM and DØDBAR allow the user to control the distributions
of spray droplet sizes about the values of $\bar{D}$ obtained from LISP. If these
variables are not provided, the empirical distribution observed for like-
doublet elements (Ref. 12) is used by default:

Cumulative Mass Fraction	0.05	0.15	0.25	0.35	0.45	0.55	0.65	0.75	0.85	0.95
$D/\bar{D}$	0.375	0.57	0.69	0.8	0.93	1.07	1.22	1.39	1.6	2.1

Stream Tube Initialization Data (Card No. 7010, et seq.) Stream tube
gases are completely defined by input of pressure, flow area, flow rate and
mixture ratio at a starting plane. All other desired parameters are calculated
from these using the combustion gas properties tables. The spray flow rate,
velocity, and droplet diameter for each spray group define the liquid input.
Of the NGT spray groups, the first NGF are fuel spray and the remainder are
oxidizer spray.

Card 7010 carries the area, gas flow rate and mixture ratio for the first
stream tube. Additionally, it has fields for the number of mesh points that
went into the stream tube and the mean r and θ coordinates of the stream tube.
These last three variables are holdovers from an earlier program: they are
written out but are not used in STC analyses. It is thus immaterial what
their values are.

3.4   PROGRAM OPERATION

In addition to understanding program input and output, a program user needs
to know operational information, such as required core size, typical execution
times, reasonable line count and CRT limits and required auxiliary data
units. Experience in these regards on Rocketdyne's IBM System 360, Mod 50/65
will be related briefly to provide preliminary indications, at least.

With the overlay structure shown in Fig. 5, the program length is 42,450
words. Total storage used during execution was about 47,200 words.

It is difficult to be quantitative about execution times for several reasons.
Foremost among these is the 360's ability to process several jobs simul-
taneously so that "time" becomes rather nebulous. Neither clock time nor
CPU time are valid indicators of what it costs to run cases. For that
reason, figures used here are Rocketdyne "billing units", which are calcu-
lated by a complicated formula but may be thought of roughly as "minutes of
execution time".

LISP runs typically cost about 1-1/2 to 3 BU (billing units) for analyzing
one collection plane. Higher costs go along with many injection elements
and many mesh points.

35

STC runs typically cost 3 to 8 BU. Initialization and single stream tube
analysis probably accounts for about 1/5 to 1/3 of that cost depending upon
the number of stream tubes. The cost is roughly proportional to the chamber
length analyzed (or the number of Δz's), and to the number of stream tubes. Other
influential variables are the numbers of spray group sizes, the number of
corrector cycles utilized, the number of CRT plots and the iteration
convergence tolerances.

Thus, a complete DER analysis may vary in cos' om about 4 to 11 or more
billing units. The example case given in Volume II cost 4.00 BU.

Printout linecount depends upon the numbers of Δz's, stream tubes, spray
groups, iterations and print intervals. It is not unusual for one case to
print 10,000 lines. Concerning CRT's, rarely does one ask for more than
10-12 graphs.

The DER program requires two data set units other than the standard input/
output units. These units, which may be either magnetic tapes or disks, are
used to transfer data between subroutines, both between and within major sub-
program blocks. Usage is shown in Table 4.

Table 4. Special Data Set Usage

Data Set		Usage
No.	Var. Name	
2	M	(1) Transfer Data from LISP to STC (STAPE)
2		(2) Save Data in STC for Plotting in STCRT
3	--	(1) Save and Retrieve Data for Iterating in STC (ITER8)

3.5 PROGRAM OUTPUT

Each of the major subprogram blocks has its own distinct output and each is
discussed separately below. Print-out is the principal output and there are
also some graphical and punched card outputs.

3.5.1 LISP Output

The output of the LISP computer program is provided both as the usual tabular printout, as computer-plotted CRT graphs and as a scratch tape record of data to be read and used by the STC computer program.

A sample of LISP tabular output is presented in Volume II. First, there is a tabulation of all input data, which permits both a full documentation of the computer run conditions for later analysis and a convenient method to check input for errors if unusual results are calculated. The input data table is followed by a one page table of specific spray distribution coefficient values used for each element specification. As in the example case, this table of coefficients may be preceded by a warning message that some unlike doublet design or flow parameter falls outside the correlation range in the LISP library of coefficients along with values of pertinent parameters. None of the other coefficient subroutines provides such a message.

Additional diagnostic date are printed out on the page following the distribution coefficient values. These are: the sums of the elemental injection rates, SWE1 and 2; the integrated sums of propellant flowrates calculated at the collection plane, TMCS1 and 2; and, inverse collection efficiency factors, RCØNT1 and 2, obtained by dividing SWE1 by TMCS1 , for example. Here, the 1 and 2 refer to the fuel and oxidizer, respectively. Overall continuity is subsequently enforced by multiplying all mesh point flows and fluxes by RCØNT1 or 2, as appropriate.

The input data is followed by a second table which cross references (by injector element) the calculated flow rates and drop sizes before evaporation to the read-in element coordinates

The element reference table is followed by two tables referenced to the combustion zone mesh points. For the usual liquid/liquid injector, the third table lists the coordinates of the mesh points in the chamber slice at the collection plane z_n. together with the weight flux $(lb_m/inch^2 sec)$, the total collected mass (lb_m/sec), the three mean droplet velocity components (ft/sec). and the mean

drop diameters (inch) of each propellant at the mesh points. The total
collected mass at a mesh point is defined as the weight flux times the
associated area at the mesh point. The values in this third table are based
upon cold flow conditions, i.e., no vaporization is assumed between elements
and mesh point. The mesh points are listed in ascending order according to
radial and angular coordinate. The last column of this table lists the sum of
the collected mass of each propellant at all the mesh-points of constant radial
coordinate, i.e., lists the radial distribution of the spray mass flux.

If the injector uses gas/liquid elements, the definition of "Propellant 1"
at the chamber mesh points is changed to mean the combustion gas resulting
from the mixing of the injected gaseous fuel and the evaporated liquid
oxidizer. Under these circumstances, the third table again lists the mesh
point coordinates and local weight fluxes and collected mass of each propel-
lant (Propellant 1 having its altered meaning), but the three spray velocity
components and mean drop diameter are defined only for Propellant 2, liquid
oxidizer. Listed in the place of the mean droplet diameter for Propellant 1

is the weight flux of evaporated oxidizer which arrives at the mesh point as part of the combustion gas. No listing of the collected propellant mass by radial coordinate is made with gas liquid injectors.

The fourth table again lists the coordinates of the mesh points in the chamber slice in the plane z_0, together with the reduced weight fluxes and droplet diameters of the collected spray after evaporation. A mass-weighted average evaporation of the original spray flux to each mesh point is also listed in this table. At the bottom of the fourth table are listed the Rupe mixing factor, E_m, the mixing limited c* efficiency, the overall percent vaporization of each propellant, and, finally, the average mixture ratio, terperature, velocity, and density of the (uniform) combustion gas at plane z_0.

The fifth table, printed only for gas/liquid injection systems, again lists the mesh point coordinates, together with the associated liquid spray flux and drop diameter after evaporation, the fraction of the total oxidizer arriving at the mesh point which is in the vapor phase, the local gas mixture ratio, temperature and axial velocity, and the axial liquid spray velocity. A number of entries in this table represent repetitions of data in other tables which are grouped here with local gas variables for convenience in interpretation.

The LISP graphical output is exemplified in Fig. 10 through 14. Fig. 10 shows the mesh system for the chamber slice analyzed and the element origin locations for all injection elements considered to contribute flux to that slice. Fig. 11 is an example of the fuel and oxidizer mass flux profiles around the chamber slice at one fixed chamber radius. Figures 12 and 13 are the contour plots of fuel and oxidizer mass flux for the entire chamber cross-section. A similar plot for total mass flux is not shown. Figure 14 is a contour plot of a modified fuel fraction function. The expression plotted is given at the top of the figure; it was chosen because it is bounded between zero and unity and has a value of 0.5 at the injection mixture ratio.
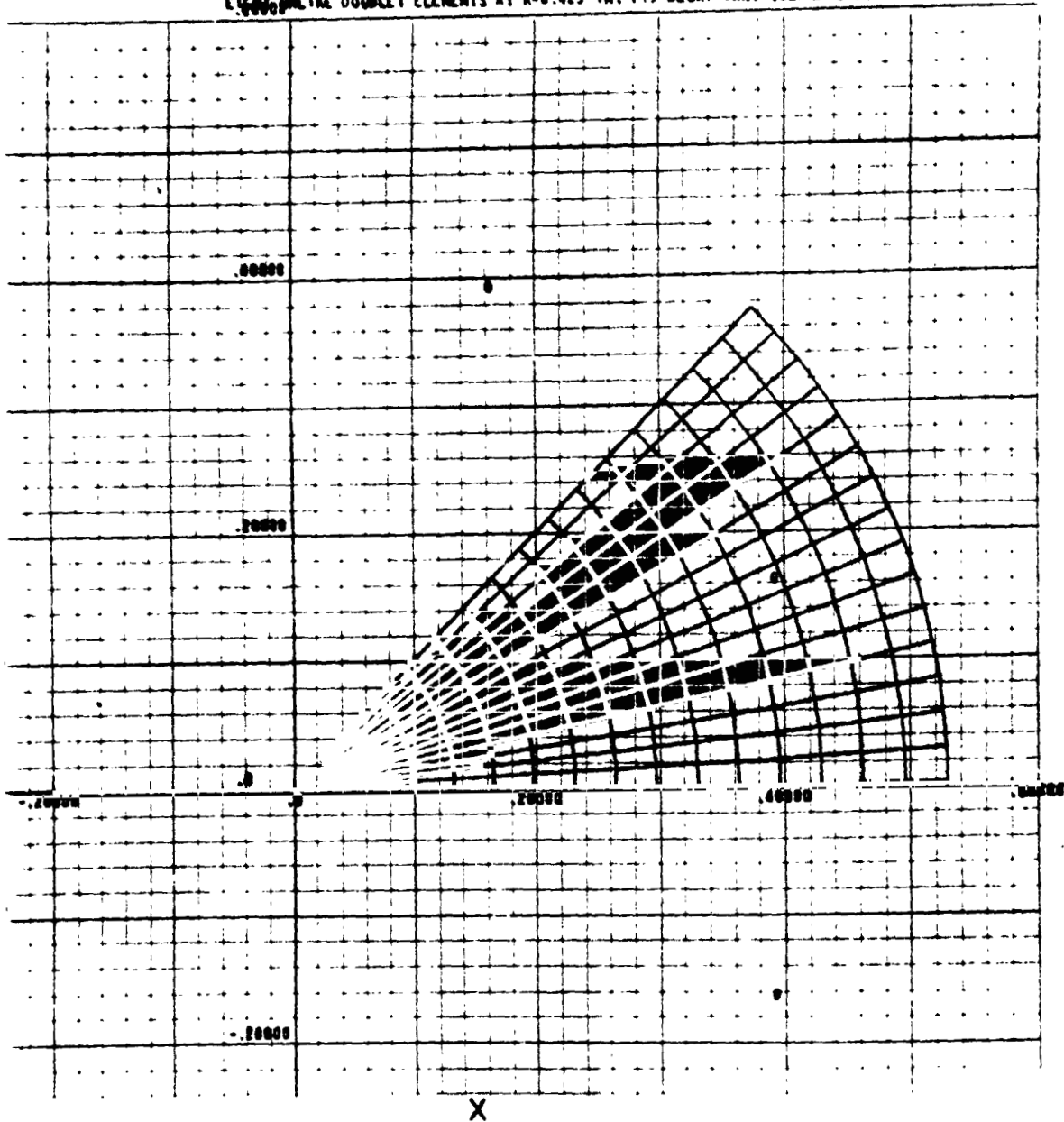
Figure 10. Segment of Injector Analyzed by LISP

89

FUEL AND OXIDIZER SPRAY FLUXES AT CONSTANT RADIUS SECTION
EXAMPLE CASE FOR D.E.R. PROGRAM USER'S GUIDE
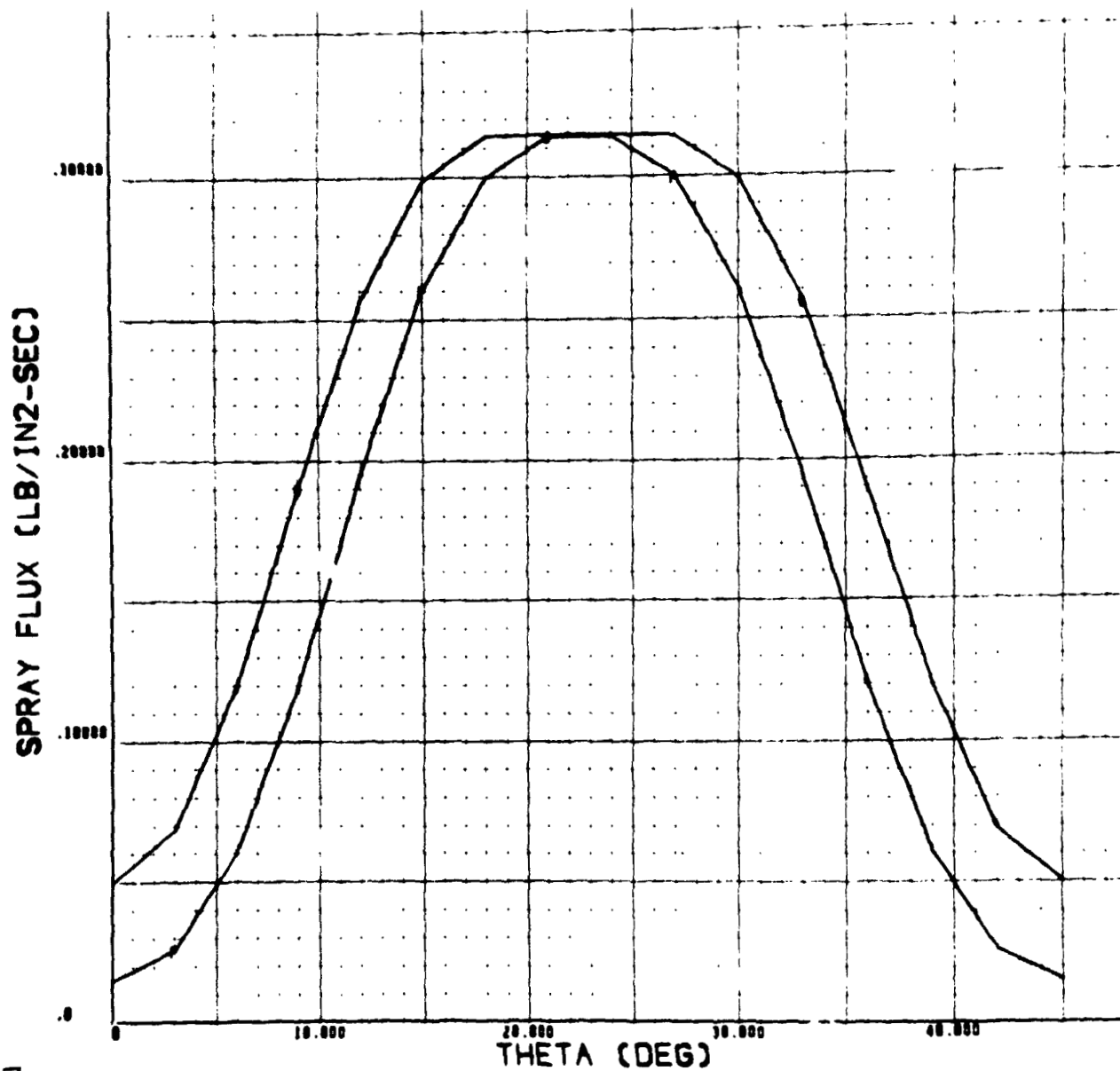EIGHT UNLIKE DOUBLET ELEMENTS AT R=0.425-IN. 7.5-DEGR CANT TOWARD WALL

Figure 11. Fuel and Oxidizer Mass Flux Profiles Computed
by LISP at a Given Chamber Radius

FUEL FLUX CONTOUR PLOT
EXAMPLE CASE FOR D E R PROGRAM USER'S GUIDE
EIGHT UNLIKE DOUBLET ELEMENTS AT R=0.425-IN. 7.5-DEGR. CANT TOWARD WALL

CONTOUR LEVELS

1	0.0200
2	0.0600
3	0.1000
4	0.1400
5	0.1800
6	0.2200
7	0.2600
8	0.3000
9	0.3400
A	0.4200
B	0.4600
	0.5000
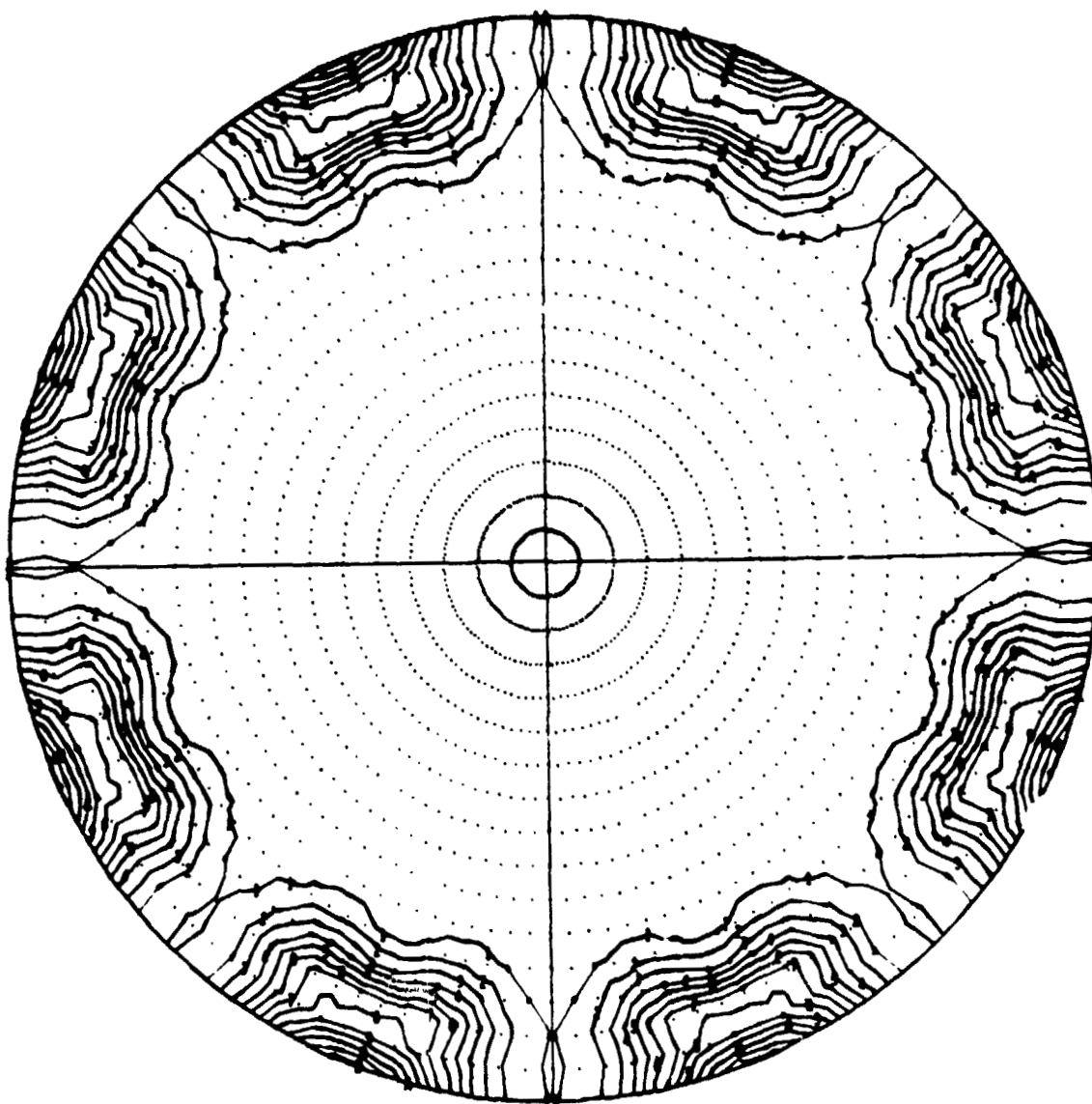
Figure 12. Contour Plot of Fuel Mass Flux Computed by LISP

91
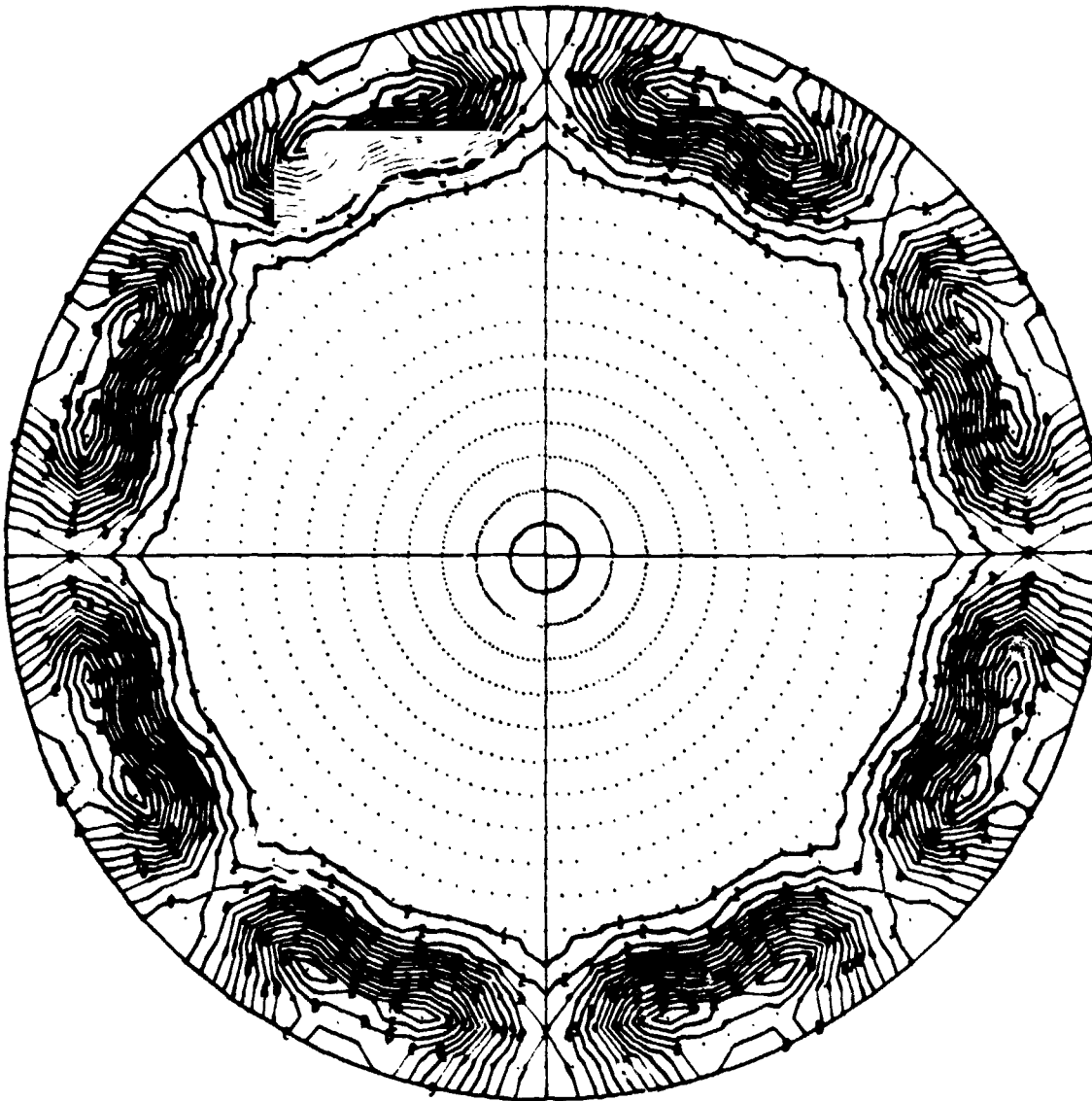
Figure 13. Contour Plot of Oxidizer Mass Flux Computed by LISP

EXAMPLE CASE FOR D E R PROGRAM USER'S GUIDE
EIGHT UNLIKE DOUBLET ELEMENTS AT R=0.425-IN. 7.5-DEGR. CANT TOWARD WALL

CONTOUR LEVELS

1	0.6350
2	0.1050
3	0.1750
4	0.2450
5	0.3150
6	0.3050
7	0.4550
8	0.5250
9	0.5950
0	0.6050
B	0.7350
	0.8050
D	0.8750
E	0.9050
F	1.0150
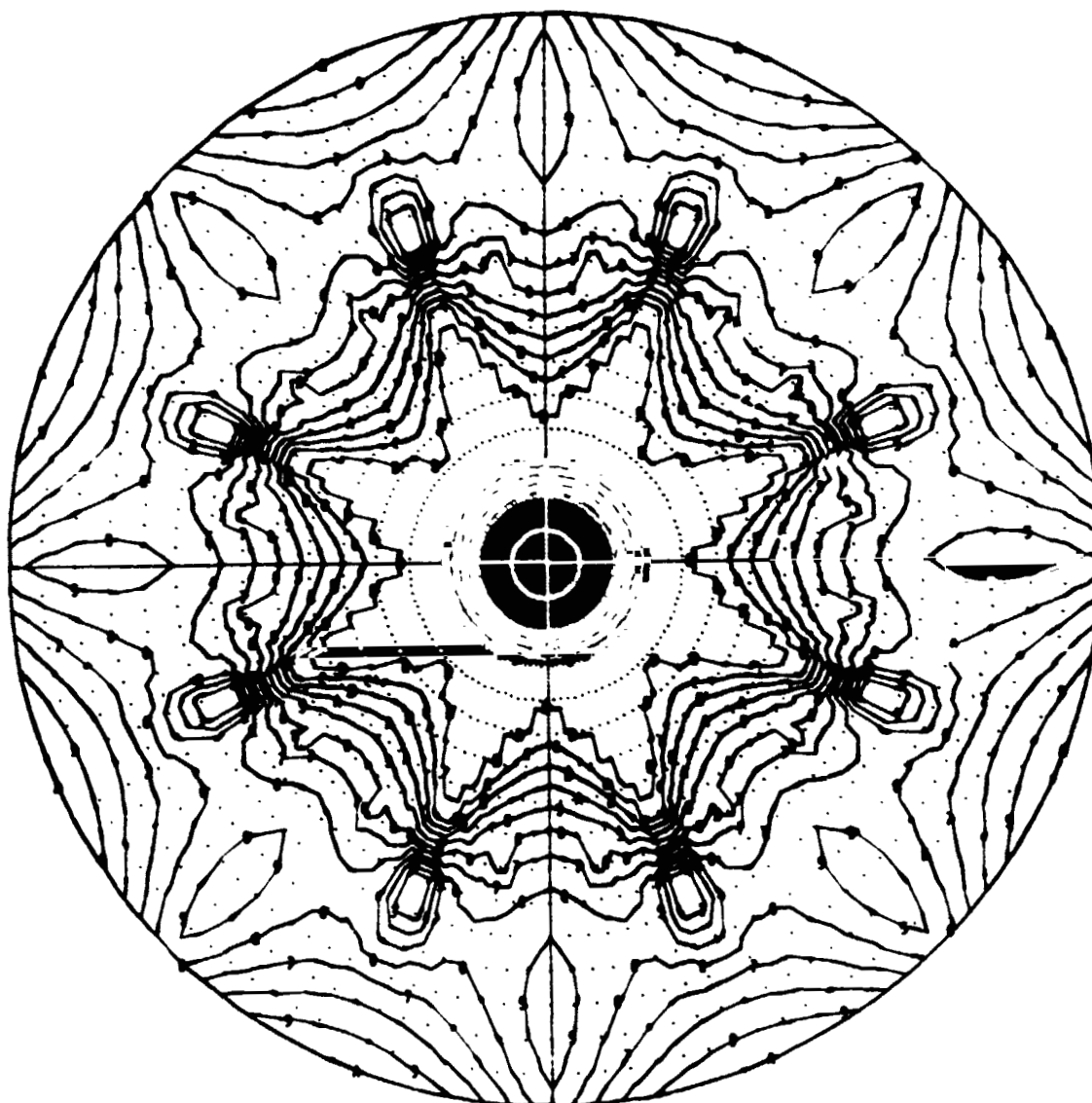
Figure 14. Contour Plot of Modified Fuel Fraction Computed by LISP

93

3.5.2 STC Output

A sample case of STC computer program printout is included in Volume II. Input data are written out immediately as they are read in. This documents the data used for the particular case as well as showing whether or not the data were read-in properly. The input section should be examined for each case run to be sure that the intended input data were actually used.

Input data trasnferred from LISP are not printed out, but a table of diagnostic data from subroutines STAPE and SCRMBL is printed. Parameters appearing in that table are, with flowrates in lb/sec:

SUMW1, 2   - Total fuel, oxidizer spray flowrates summed over all mesh points in STAPE

WGF, $\emptyset$ - Gaseous fuel, oxidizer flowrates

TIF1, 2   - Total fuel, oxidizer flowrates transferred from LISP

FF, F$\emptyset$ - The ratio TIF1/(SUMW1 + WGF), etc.

K   - The number of circular rings of mesh points from LISP

SMBL   - Total Flowrate assigned to the wall boundary layer stream tube

PCTT   - The ratio SMBL/(TIF1 + TIF2)

$\emptyset$TT   - The product PCTBL*(TIF1 + TIF2)

N1,N2   - The indices of circular rings of mesh points in a given geometric zone

SUM   - The cumulative total flowrate in a geometric zone and those set up prior to it

The stream tube initialization data are tabulated and simultaneously punched out in cards.

Based on the stream tube initialization data, a table is printed out of stream tube total flowrates and overall mixture ratios. This table is followed by values of the Rupe mixing efficiency factor, E_m, and a mixing c* efficiency for the stream tube flows. The latter represents an upper limit for multiple stream tube c* efficiency, since it corresponds to complete evaporation and burning of sprays within all stream tubes.

Subroutine AVAR sets up the array of chamber areas and writes out a table of chamber geometry information. Similarly, subroutine KPRIME computes and writes out tables of evaporation coefficients.

Single stream tube analysis is preceded by writing out a one-page table of input total flows and averaged spray and gas parameters. During single stream tube analysis, data are written out as they are generated. At each z-plane to be printed, complete gas and propellant spray group data are given. Additionally, the percentages of propellants evaporated and burned are listed and volume number mean propellant droplet diameters, D_{30}, are computed.

Calculated values of local flow Mach number and stagnation pressure (obtained from static pressure, Mach number and frozen gamma via the isentropic relationship) are printed in the right hand margin. In the printout of Volume II, these have been torn off.

Two values each of flow area and contraction ratio may be given. Where the gas flow is subsonic, these should agree with each other precisely, whereupon the second set is not printed; if they do not there is some discrepancy between the read-in APROF(J,L) and AREA1(NT) arrays. At or near the nozzle throat plane, the two sets may disagree because the gas velocity has been set equal to sound speed and the local nozzle area adjusted to satisfy mass flow continuity. The contraction ratio calculated from continuity at the throat

plane is used as a multiplier (if it differs from unity by more than CRT∅L) to adjust initial plane chamber pressure for a next iteration of single stream tube analysis. After the throat plane data are writte: out, a calculated c^* efficiency is listed.

When the foregoing analysis has converged on its solution, the input value of nozzle radius ratio and calculated value of mean nozzle expansion coefficient, $\bar{\gamma}$, are used by TRANS to generate transonic flow region isobars. The reduced coordinates and flow directions for each of 20 points along each isobar are written out, ber'nning with the furthest downstream isobar and progressing upstream. Additionally, for the $\alpha=0$ isobar, the absolute coordinates are written out for 40 points. Finally, a value is printed out for the nozzle discharge coefficient, C_{ND}.

TRANS also generates a CRT plot of the isobars' coordinates. Examples were shown earlier in Fig. 3.

Multiple stream tube analysis follows the foregoing single stream tube and TRANS analyses. Stream tube input data are re-initialized and some additional data are written out to more completely define the initial-plane conditions. Initial-plane pressure is taken as the product of its value left in storage from the last preceding single stream tube iteration multiplied by the mixing c^* efficiency and divided by TRANS's nozzle discharge coefficient.

At each prescribed z-plane for printing multiple stream tube results, complete definitive data for combustion gases and propellant sprays are written out. Local chamber area and contraction ratio are given; additionally, overall percentages of the propellants evaporated and burned are listed. Calculated stream tube stagnation pressures are printed in the right hand margin, which has been torn off in the printout of Volume II.

At the throat position and intermittently downstream, diagnostic-type printouts containing data concerning dividing streamline intersections with the fifth, $\alpha=0$, isobar are inserted between the regular z-plane printouts. A summary table of these data is given near the end of the multiple stream tube printout. Finally, a long summary table is given of the stream tubes' outer radii at each z-plane. This is terminated with the minimum value of the sum of stream

tube areas and the ratio of that value to throat area ratio. The magnitude
of this latter value's deviation from unity determines whether or not all,
or a portion*, of STC's multiple stream tube analysis will be repeated. If
so, it is readily apparent in the printout.

One (or more) computer-plotted graphs accompany the printout discussed above,
showing the stream tubes' outer radii along the entire chamber length,
illustrated in Fig. 15. One of these graphs is plotted for each iteration
through the multiple stream tube analysis.

Following the last pass through the multiple stream tube analysis, a table
is print d which lists the NAMELIST data punched by STC for subsequent use
in running the improved TDK computer program. These are in the form of a TDK
data deck, with all appropriate title cards, a problem definition card and
NAMELIST data file cards. Only a part of the required data can be provided by
STC, however, so comment cards are included to indicate the additional input
data required before TDK can be run. STC provides values for the following
parameters calculated at the throat plane:

> NZ∅NES-------The number of stream tubes
> P(1)---------Stream-tube-area-weighted mean stagnation pressure
> XM(J)--------Mass fraction of the total gasified propellant flow
>   carried by each stream tube
> ∅FSKED(J)----The gasified propellant mixture ratio of each stream
>   tube

Finally, a table is printed out of additional data in anticipation of their
being needed and used in a future revision of TDK. Included are each stream
tube's cross-sectional area, ASUBS(J), static pressure, PSUBS(J) and stream gas
velocity, VSUBS(J) at the position Z=ZPVSR where the TRANS pressure distribution
is first invoked. Additional nozzle throat plane data are each stream tube's
m--s fraction (gas and spray) of the total injected propellant flowrate, XMI(J),
overall mixture ratio, ∅FI(J), stagnation pressure, PNS(J), mean evaporation
efficiency, ETAVAP(J), and mean fuel and oxidizer spray velocities,VBARF(J) and
VBAR∅(J).

---

*Subroutine ITER8 writes all pertinent data on scratch data unit 3 when the
 multiple stream tube analysis reaches the plane where absolute pressures
 are imposed on the nozzle. If only the subsequent portion of the analysis is
 to be iterated upon, ITER8 is recalled to read that record and reinitialize
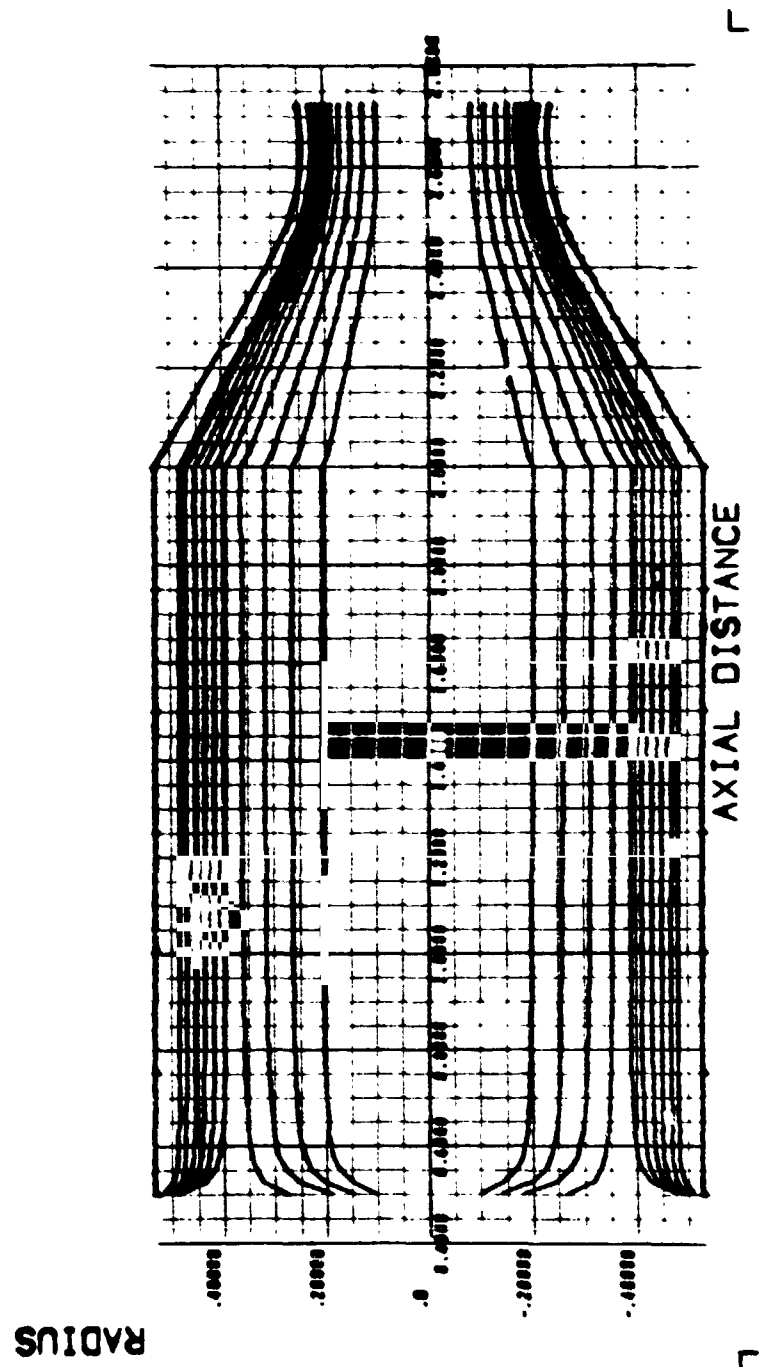 the analysis in that plane.

Figure 15. Example of Computer-Plotted Dividing Stream Lines from STC Program Block

97

4. REFERENCES

1. Combs, L. P., Chadwick, W. D. and Campbell, D.T.: "Liquid Rocket Performance Computer Model with Distributed Energy Release," Interim Final Report, NASA-CR-111000, (R-8298), Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, September 1970.

2. Combs, L. P. and Chadwick, W. D.: "Liquid Rocket Performance Computer Model with Distributed Energy Release - Computer Program Documentation," Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, September 1970.

3. Frey, H. M. and Nickerson, G. R.: "TDK, Two-Dimensional Kinetic Reference Program, Improved Version", Dynamic Science, a Division of Marshall Industries, Irvine, California, December 1970.

4. Hines, W. S., Combs, L. P., Ford, W. M., and VanWyk, R., "Development of Injector Chamber Compatibility Analysis," Final Report, AFRPL-TR-70-12, Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, March 1970.

5. Hines, W. S., Schuman, M. D., Ford, W. M. and Fertig, K. W.: "Extension of a Thrust Chamber Compatibility Model," Final Report, AFRPL-TR-72-19, Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, March 1972.

6. Combs, L. P., et al.: "Catalog of Injector Spray Correlations," Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, Contract NAS7-746, June 1972.

7. Rupe, J. H. and Jaivin, G. H., "The Effects of Injection Mass Flux Distributions and Resonant Combustion on Local Heat Transfer in a Liquid Propellant Rocket Engine," Prog. Rpt. 32-648, Jet Propulsion Laboratory, Pasadena, California, October 1964.

8. Kliegel, J. R., et al, "ICRPG Two-Dimensional Kinetic Reference Program," Dynamic Science, a Division of Marshall Industries, Irvine, California, July 1968.

9. Nickerson, G. R., "Instructions for Replacing the Transonic Analysis of the TDK Computer Program," Dynamic Science Letter Report, received April 1970.

10. Kliegel, J. R. and Levine, J. N., "Transonic Flow in Small Throat Radius of Curvature Nozzles," AIAA Journal, Vol. 7, No. 7, July 1969.

11. Chadwick, W. D. and Combs, L. P.: "Pulse Mode Performance Model," Final Report, Contract F04611-70-C-0074, Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, AFRPL-TR-72-16, November 1972.

12. Dickerson, R. A., Tate, K. and Barᶜic, N., "Correlation of Spray Injector Parameters with Rocket Engine Performance," Final Report, AFRPL-TR-68-147, Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, June 1968.

13. Falk, A. Y., Clapp, S. D. and Nagai, C. K., "Space Storable Propellant Performance Study," Final Report, NASA CR-72487, Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, November 1968.

14. Gordon, S. and McBride, B. J.: "Computer Program for Calculation of Complex Chemical Equilibrium Composition, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jonguet Detonations," NASA-SP-273, National Aeronautics and Space Administration, Washington, D.C., 1971.

15. Thompson, R. J., Jr., The Chemistry of Propellants, Pergamon Press, New York, 25-120 (1960).

16. Sutton, R. D. and Combs, L. P.: "Propellant Physical Property Data and Computer Input Cards - Documentation Report for N_2H_4/F_2 and MMH/FLOX Combustion Systems," Final Report on JPL Purch. Order No. DD-422166, Rocketdyne, a Division of North American Rockwell Corporation, Canoga Park, California, 15 December 1971.