

APOLLO

GUIDANCE, NAVIGATION AND CONTROL

Approved: Gerald M. Levine Date: 17 June 71
G.M. LEVINE, DIRECTOR, GUIDANCE ANALYSIS
APOLLO GUIDANCE AND NAVIGATION PROGRAM

Approved: R. H. Battin Date: 17 June 71
R.H. BATTIN, DIRECTOR, MISSION DEVELOPMENT
APOLLO GUIDANCE AND NAVIGATION PROGRAM

Approved: D. G. Hoag Date: 18 June 71
D.G. HOAG, DIRECTOR
APOLLO GUIDANCE AND NAVIGATION PROGRAM

Approved: R. R. Ragan Date: 18 June 71
R.R. RAGAN, DEPUTY DIRECTOR
CHARLES STARK DRAPER LABORATORY

R-695

APOLLO LUNAR-DESCENT GUIDANCE

by
Allan R. Klumpp

JUNE 1971

MIT

CAMBRIDGE, MASSACHUSETTS, 02139

**CHARLES STARK DRAPER
LABORATORY**

ACKNOWLEDGMENTS

This report was prepared under DSR Project 55-23890, sponsored by the Manned Spacecraft Center of the National Aeronautics and Space Administration through Contract NAS9-4065.

The P66 vertical channel was developed by Craig W. Schulenberg. The analytical design and gain setting of the P66 horizontal channels was done by Nicholas J. Pippenger using concepts suggested by Jerrold H. Suddath. The concept of analytically extrapolating to yield the predictive guidance equation for P63 and P64 was conceived by William S. Widnall. The existence of an analytical solution for the guidance frame orientation to yield zero crossrange target jerk was recognized by Thomas E. Moore. The thrust direction filter configuration for eliminating thrust-pointing errors due to attitude bias within the digital autopilot deadband was conceived by William S. Widnall and Donald W. Keene.

The publication of this report does not constitute approval by the National Aeronautics and Space Administration of the findings or the conclusions contained therein. It is published only for the exchange and stimulation of ideas.

© Copyright by the Massachusetts Institute of Technology
Published by the Charles Stark Draper Laboratory of the
Massachusetts Institute of Technology.
Printed in Cambridge, Massachusetts, U. S. A. , 1971

APOLLO LUNAR-DESCENT GUIDANCE

ABSTRACT

This report records the technology associated with Apollo lunar-descent guidance. It contains an introduction plus five major sections:

1. Braking-phase and approach-phase guidance. Braking-phase guidance begins in lunar orbit prior to engine ignition and transfers the lunar module (LM) to a terminus typically 7 800-m slant-range before the landing site. Approach-phase guidance begins at braking-phase terminus and transfers the LM to a terminus typically 30-m above the landing site. The braking-phase transfer is near-optimal, whereas the approach-phase transfer sacrifices propellant-utilization efficiency to provide landing-site visibility and landing-site redesignation capability.
2. Terminal-descent-phase guidance. Initiated automatically at approach-phase terminus, or manually any time during the approach phase, terminal-descent-phase guidance automatically nulls horizontal velocity and controls altitude rate to a reference value. The reference value is incremented or decremented by astronaut manipulation of a rate-of-descent control switch.
3. Powered-flight Attitude-maneuver Routine. The routine connects all powered-flight guidance programs — descent and ascent — to the digital autopilot. A departure from traditional approaches, the routine transfers the LM from any current attitude to any commanded attitude while avoiding gimbal lock by inherent characteristics of the maneuver algorithm. No gimbal-lock-avoidance strategy is required.
4. Throttle Routine. The routine connects several guidance programs to the descent propulsion system (DPS). DPS hardware limitations require operation either at a maximum-thrust point or within a separate permitted-thrust region. The routine alters thrust commands from the guidance programs when necessary to meet DPS constraints and issues corrected thrust increment commands.

5. Braking-phase and Approach-phase Targeting Program. This ground-based program is used at the Draper Laboratory and at the NASA Manned Spacecraft Center. The program supplies descent targets which are loaded into guidance computer memory shortly before launch.

by
Allan R. Klumpp
June 1971

TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
INTRODUCTION	1
Descent Phases	1
Navigation, Guidance, and Control Configuration	4
Navigation	4
Guidance	4
Powered-flight Attitude-maneuver Routine	6
Throttle Routine	6
Braking-phase and Approach-phase Targeting Program	7
Digital Autopilot	7
DEFINITIONS OF LUNAR DESCENT COORDINATE FRAMES, ATTITUDE ANGLES, AND GIMBAL ANGLES	9
BRAKING-PHASE AND APPROACH-PHASE GUIDANCE	13
Guidance Equation Derivation	13
Braking-Phase Targeting Objectives	18
Approach-Phase Targeting Objectives	19
Landing-Site Redesignation Procedure	19
P63, P64 Guidance Algorithm	21
P63 Ignition Algorithm	27
TERMINAL-DESCENT-PHASE GUIDANCE	31
P66 Horizontal Guidance Algorithm	31
P66 Vertical (ROD) Guidance Algorithm	32
POWERED-FLIGHT ATTITUDE-MANEUVER ROUTINE	37
THROTTLE ROUTINE	45

TABLE OF CONTENTS (Cont.)

<u>Section</u>	<u>Page</u>
BRAKING-PHASE AND APPROACH-PHASE TARGETING PROGRAM . . .	51
Constraints	53
Approach-phase Targeting	54
Braking-phase Targeting	56
REFERENCES	61

NOMENCLATURE

Symbols are normally defined where first introduced. Therefore it is necessary to define here only those symbols used in more than one section of this report.

Most symbols are self-defining by being constructed of standard identifiers as follows:

1. Type of variable. Position and its derivatives velocity, acceleration, jerk, and snap are denoted R, V, A, J, S . Thrust is denoted F , thrust-acceleration AF , unit vectors $\underline{U}N$, clock-times by lower case t , and target-referenced times (times with respect to the target point of a particular mission phase) by upper case T .
2. Mission phase. The braking phase (P63) is denoted BR , the approach phase (P64) AP .
3. Applicable point in phase. Inception is denoted I , terminus F , and target point T .
4. Coordinate frame of reference. Platform coordinates are denoted P , guidance G , and LM body B .
5. Achieved (as opposed to nominal) values are denoted A .

Thus by construction, $\underline{R}BRFGA$ is the position vector expressed in guidance coordinates achieved at the braking phase terminus. Without a phase identifier, $\underline{R}TG$ to $\underline{S}TG$ represent the braking or approach phase targets $\underline{R}BRTG$ to $\underline{S}BRTG$ or $\underline{R}APTG$ to $\underline{S}APTG$, whichever phase is current. Vector elements are denoted by subscript, e.g., V_Z . Vector magnitudes are implied whenever a symbol reserved for a vector lacks the underscore. Row vectors of 3×3 matrices are denoted $\underline{C}_X, \underline{C}_Y, \underline{C}_Z$, and matrix elements are identified by row and column, e.g., C_{YZ} is the Z component of the row vector $\underline{C}_Y$.

<u>Symbol</u>	<u>Definition</u>
$\underline{A}FCP$	Thrust-acceleration command
$\underline{A}FP$	Thrust-acceleration measurement (computed by dividing the inertially sensed velocity increment by the guidance sample interval)
AGS	Abort Guidance System
ATT	Powered-flight Attitude-maneuver Routine
CBP	Transformation to body from platform coordinates
CGP	Transformation to guidance from platform coordinates

DAP	Digital autopilot
DEC A	Descent engine control assembly. A digital to analog interface between the LGC and the DPS throttle
DPS	Descent propulsion system
$\delta F A$	Thrust correction increment which must be added to the thrust measurement averaged over the sample interval to obtain the sample-instant thrust.
GM	Lunar gravity (acceleration of positive sign valid at the lunar surface)
<u>G</u> P	Current gravity valid at the current position <u>R</u> P
IMU	Inertial measurement unit consisting of a three-gimballed stable member and three accelerometers
LEADTIME	A time interval (typically 2.2 seconds) added to the target-referenced time T in P63 and P64 to account for the effective transport lag due to computation and system response times
LGC	LM guidance computer
LIMIT	A function of two arguments yielding the first argument limited in magnitude to the value of the second argument
LM	Lunar module
<u>L</u> P	Landing site
LPD	Landing point designator consisting of two reticles, one on the inside window panel, and one on the outside
MAXIMUM	A function of n arguments yielding the algebraically highest argument
MINIMUM	A function of n arguments yielding the algebraically lowest argument
NASA	National Aeronautics and Space Administration
Pitch Yaw Roll	LM attitude angles. See "Definitions of Lunar Descent Coordinate Frames, Attitude Angles, and Gimbal Angles"
P63	Braking phase or braking-phase guidance program
P64	Approach phase or approach-phase guidance program
P66	Terminal-descent phase or terminal-descent-phase guidance program
<u>R</u> APTG, <u>R</u> BRTG, <u>R</u> TG, <u>V</u> APTG, <u>V</u> BRTG, <u>V</u> TG, <u>A</u> APTG, <u>A</u> BRTG, <u>A</u> TG, <u>J</u> APTG, <u>J</u> BRTG, <u>J</u> TG, <u>S</u> APTG, <u>S</u> BRTG, <u>S</u> TG	Target position, velocity, acceleration, jerk, and snap of the approach phase, braking phase, or current phase
<u>R</u> G, <u>R</u> P	Current position
ROD	Rate of descent
t	clock-time
T	Target-referenced time (time with respect to the target point of the current mission phase)

THROT	Throttle Routine
<u>UNFCP</u>	Unit thrust command
<u>UNWCP</u>	Unit window command
<u>VG, VP</u>	Current velocity

INTRODUCTION

This report describes how the Apollo lunar-descent guidance works, why it was designed this way, and, in several cases, how it might have been designed differently. The concepts described can be applied to landing on any planetary body, with or without atmosphere, should man resolve to continue this adventure. The solutions presented offer ample opportunity for checking the theory. Such checks have been made, and all algorithms are known to work as conceived.

Lunar-descent guidance begins with the lunar module (LM) at about 15-km altitude in a slightly elliptical coasting lunar orbit, and ends with the LM on the lunar surface. The guidance is performed by the onboard LM guidance computer (LGC), which takes input data and commands directly from the LM crew and via the uplink from NASA's Real Time Computation Center in Houston, Texas. The crew consists of a commander and a LM pilot. (See Figure 1.) Standing on the left, the commander monitors and controls the descent using visual cues and various hand controllers and switches. Standing on the right, the LM pilot monitors the computer display, vocally relays pertinent data to the commander, and enters any necessary data into the computer via the keyboard.

The primary guidance mode for the lunar descent is automatic; the LGC controls both attitude and thrust. The commander can, temporarily or permanently, select nonautomatic guidance modes if he wishes to control, manually, attitude or thrust or both. The nonautomatic modes, not described further in this report, provide attitude and thrust references for the commander to follow if he chooses to fly the LM manually along the automatic guidance profile.

Descent Phases

The lunar descent is a nominally-planar trajectory consisting of three phases illustrated in Figure 2 and described as follows:

1. The braking phase (Program 63, or P63) is initiated by keyboard entry about 10 minutes before nominal ignition time. P63 first computes the precise time and attitude for ignition. Next, at typically 492-km slant-range from the landing site, P63 ignites the DPS. Finally, P63 transfers the LM to the terminal state required as initial conditions for the succeeding approach phase. The transfer takes typically 514 seconds and is near-optimal.

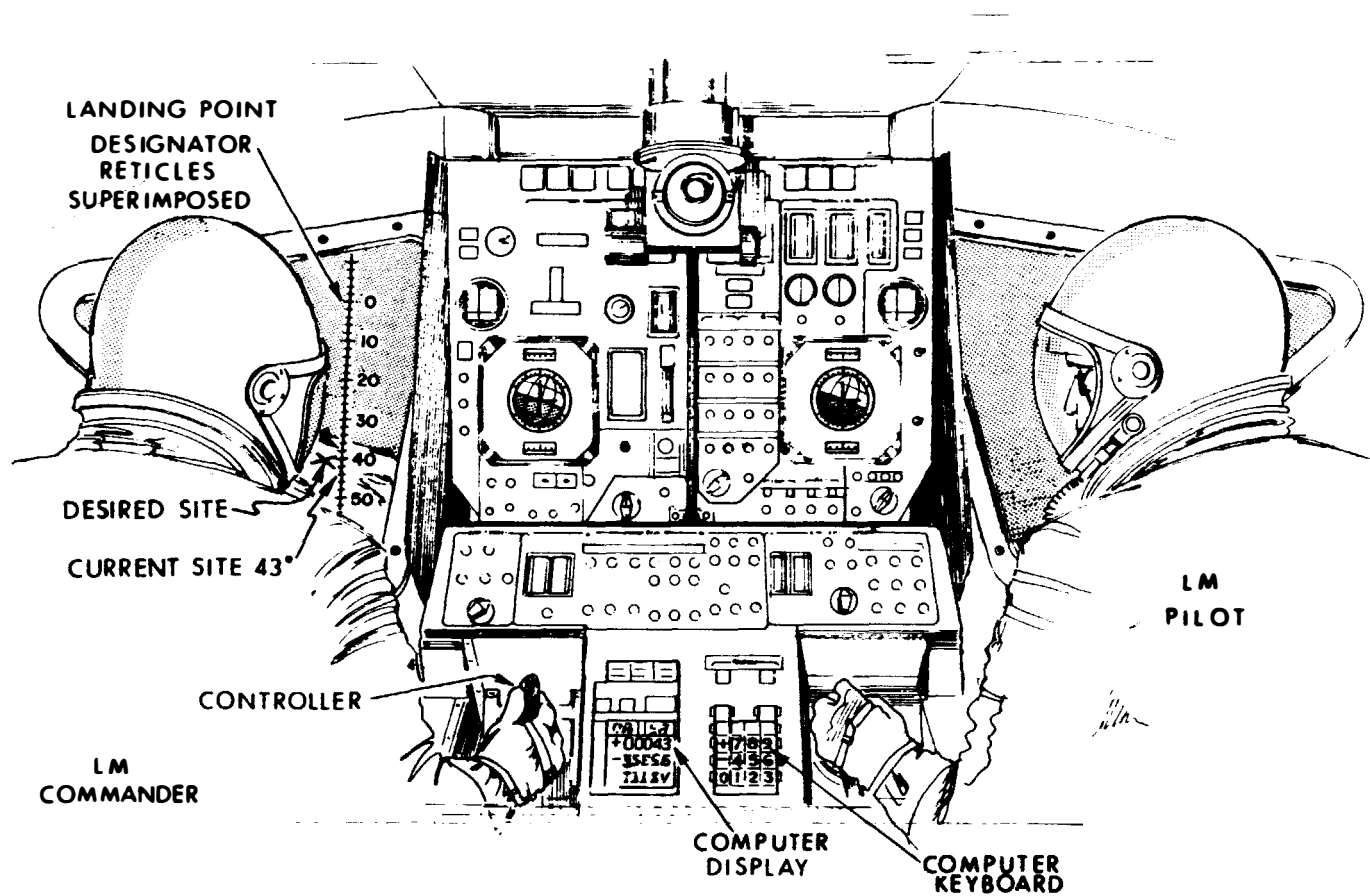


Figure 1 Components of the Lunar-descent Guidance System

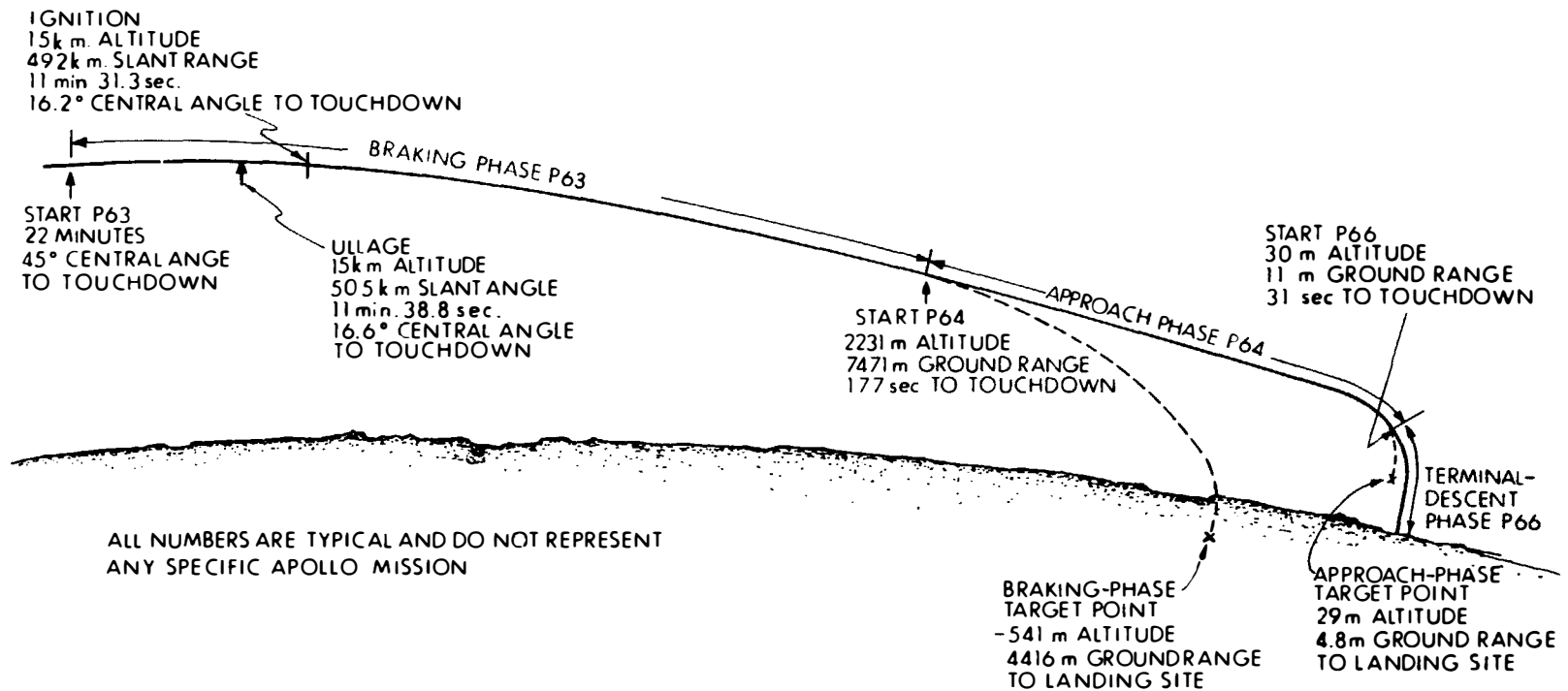


Figure 2 Lunar-descent-guidance Phases and Targets

2. Approach-phase (P64) guidance begins with initial conditions consisting of, typically a) 2.2-km altitude and 7.5-km ground range and b) -44-m/sec vertical velocity and 129-m/sec forward velocity. In typically 146 seconds, P64 transfers the LM to a point almost directly above the landing site. P64 provides continuous visibility of the lunar surface and, specifically, of the landing site until around 5 seconds before terminus. During P64 the commander can direct the LGC to land at any visually chosen point on the lunar surface by a landing-site redesignation procedure which can be continued until initiation of the terminal-descent phase.

3. The terminal-descent phase (P66) begins automatically at typically 30-m altitude and 11-m ground range from the landing site, or it may be initiated by the commander any time during P64. The P66 guidance algorithm controls velocity only; there is no position control. P66 nulls the forward and lateral velocity components to produce a vertical approach to the lunar surface, an objective which cannot be achieved from visual cues when the surface is obscured by a sheet of radially moving dust. P66 controls altitude rate to a reference value that is incremented or decremented by 0.3-m/sec each time the commander raises or lowers a three-position rate-of-descent (ROD) control switch located near his left hand.

Navigation, Guidance, and Control Configuration

The Navigation, Guidance, and Control configuration illustrated in Figure 3 applies to all LM powered-flight guidance maneuvers. This report describes only the solid portions of Figure 3. All routines are processed once every two seconds, except the vertical channel of the P66 guidance algorithm is processed once per second, and the digital autopilot is processed 10 times per second.

Navigation

Navigation (see Kriegsman¹) provides an estimate of the current state vector based on data from an inertial measurement unit (IMU) and a landing radar. IMU data are used throughout all thrusting maneuvers, but, to avoid accumulation of inertial errors, IMU data are not used during coasting flight except for a minimum period immediately preceding and following each thrusting maneuver. The landing radar provides altitude data below typically 10-km altitude, and velocity data below 610-m/sec.

Guidance

Guidance transfers the LM from the current state to the terminus of the current phase. In addition to the current state estimate from Navigation, Guidance is based

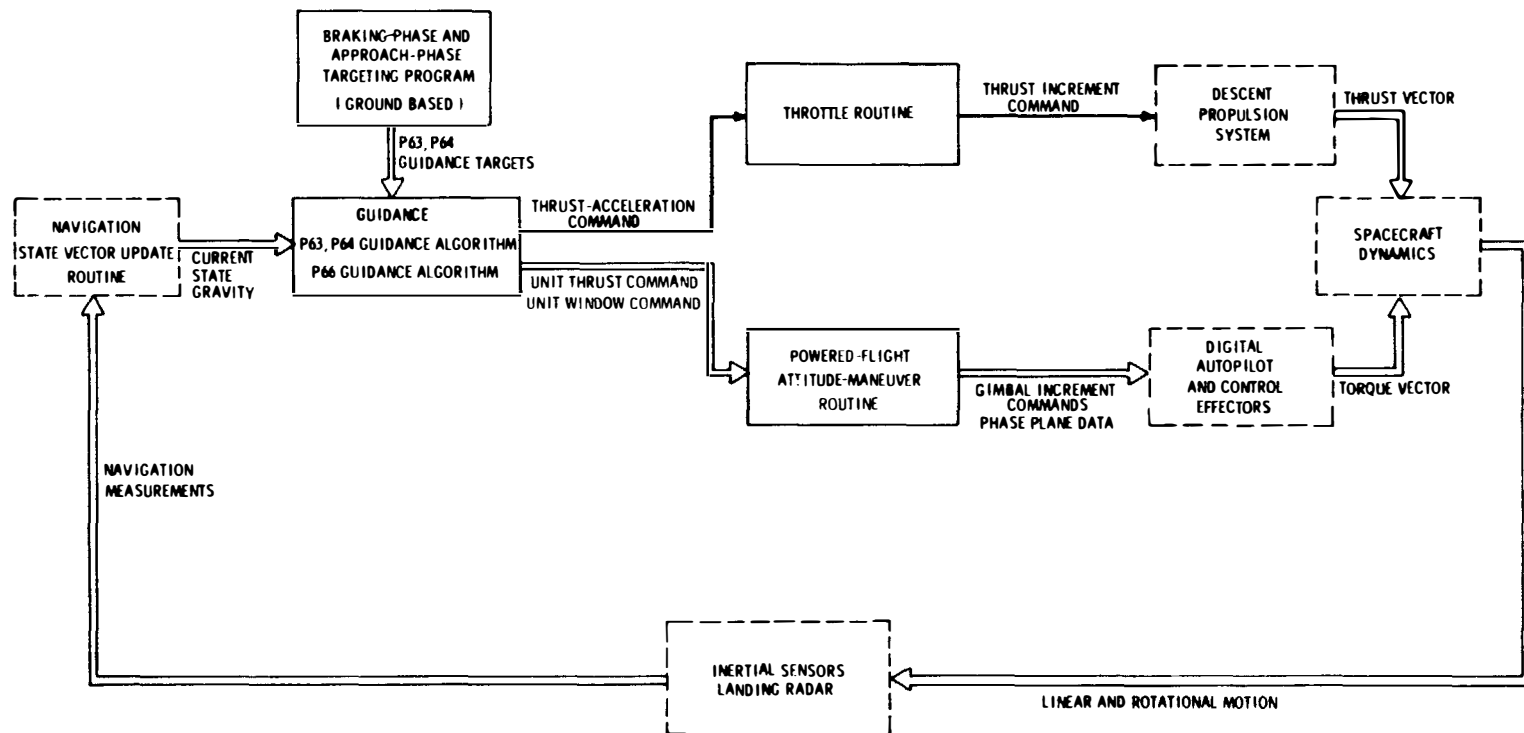


Figure 3 Navigation, Guidance, and Control Configuration

on precomputed targets from the ground-based targeting program. The outputs from the Guidance algorithm are a unit thrust command and a unit window command issued to the Powered-flight Attitude-maneuver Routine, and a thrust-acceleration command issued to the Throttle Routine. Through these routines, Guidance controls the thrust vector magnitude and direction with respect to inertial space.

Powered-flight Attitude-maneuver Routine

The Powered-flight Attitude-maneuver Routine (ATT) connects all guidance programs, descent and ascent, to the digital autopilot (DAP). ATT inputs are two command vectors; a unit thrust command and a unit window command. ATT estimates a unit thrust vector from accelerometer measurements, and issues incremental commands to the DAP. These commands cause the DAP to drive the estimated unit thrust vector into coincidence with the unit thrust command and the symmetry plane of the LM into coincidence with the unit window command.

During P64, as long as the landing site would be visible, the unit window command issued to ATT by Guidance is the line-of-sight vector to the current landing site. By rotating the LM symmetry plane into coincidence with the line-of-sight vector, ATT superimposes the landing-point designator reticles of Figure 1 on the current landing site.

Throttle Routine

The Throttle Routine (THROT) connects several powered-flight guidance programs to the DPS. The DPS is used by all descent guidance programs, one of the two abort programs, and one guidance program whose purpose is to provide a velocity-vector increment computed by the Real Time Computation Center in Houston and transmitted to the LM.

The DPS must be operated either at the maximum-thrust point (about 92% of the rated thrust of 46 706 newtons) or within a permitted-thrust region (11 to 65% of rated thrust). The intervening region (65 to 93%) is forbidden because in this region oxidizer flow and fuel flow make independent transitions from cavitating to noncavitating regimes. The independent transitions cause gross deviations from the required mixture ratio and produce excessive erosion of the DPS nozzle.

Using a computed mass estimate, a thrust-acceleration measurement, and the thrust-acceleration command from the guidance equations, THROT computes the current and commanded thrusts and issues thrust increment commands to the

DPS. These commands either 1) drive the computed thrust into coincidence with the commanded thrust whenever the commanded thrust lies within the permitted-thrust region, 2) produce maximum thrust whenever the commanded thrust lies above the permitted-thrust region, or 3) produce minimum thrust whenever the commanded thrust lies below the permitted-thrust region.

Braking-phase and Approach-phase Targeting Program

The targeting program provides targets for P63 and P64. The targets define braking- and approach-phase reference trajectories as independent vector polynomials centered at individual target points as illustrated in Figure 2. Although only P63 and P64 are targeted, the targets are designed to achieve all the guidance objectives of P63, P64, and P66.

Digital Autopilot

The DAP (see Widnall²) controls the attitude of the LM during powered flight by means of control effectors consisting of a reaction control system and a trim gimbal system. As the name implies, the trim gimbal system is a slow system used primarily for trimming the DPS thrust vector through the LM center of mass.

DEFINITIONS OF LUNAR DESCENT COORDINATE FRAMES, ATTITUDE ANGLES, AND GIMBAL ANGLES

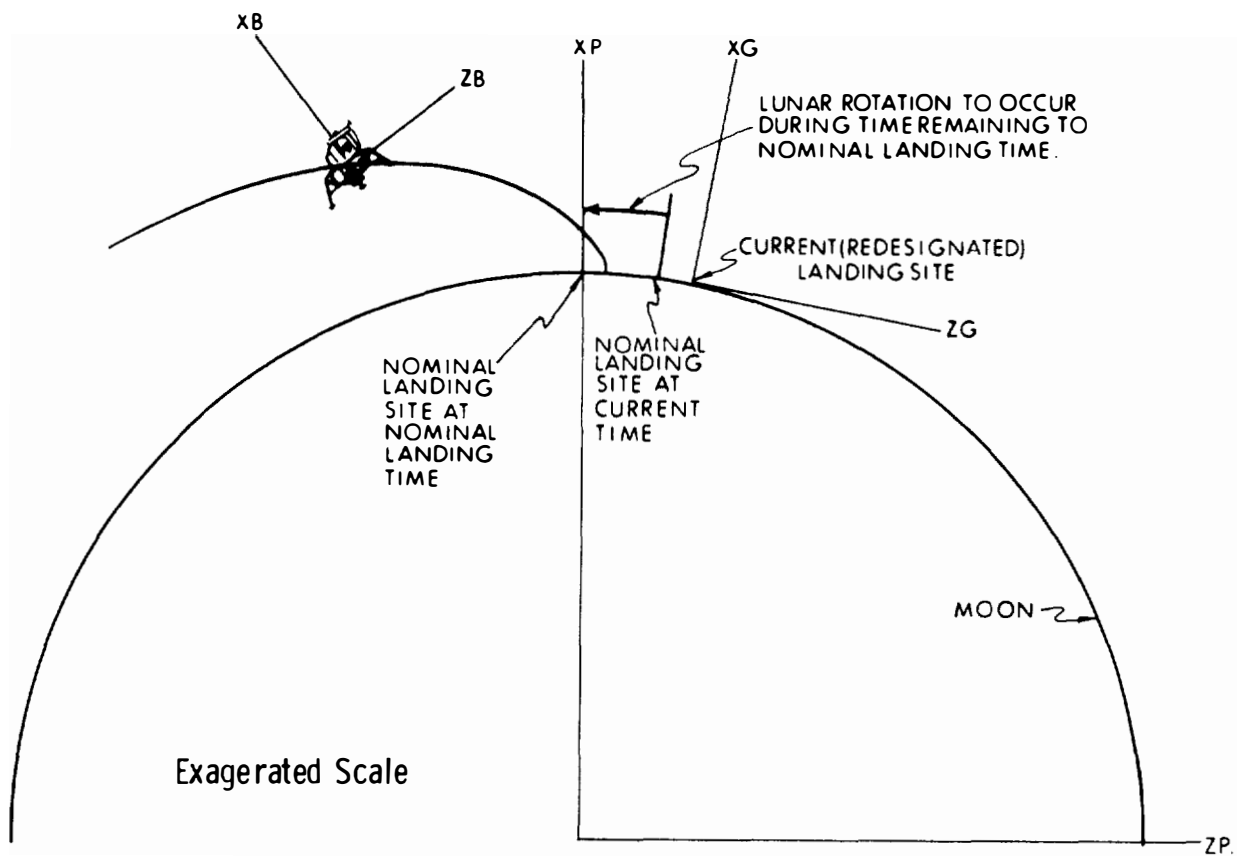
Three coordinate frames are required for lunar descent guidance because 1) all inertial measurements are with respect to the stable platform of the IMU, 2) P63 and P64 guidance is with respect to a landing site which rotates with, and can be redesignated along, the lunar surface, and 3) thrust-vector determination and landing-site redesignations are with respect to LM body axes. These coordinate systems are illustrated in Figure 4 and defined as follows:

1. Platform coordinates. Variables in platform coordinates are tagged P. The origin is at the center of the moon, the XP-axis pierces the nominal (unredesignated) landing site at the nominal landing time, the ZP-axis is parallel to the orbital plane of the Command Module* and points forward, and the YP-axis completes the right-hand triad. Platform coordinates are nonrotating.
2. Guidance coordinates. Variables in guidance coordinates are tagged G. The origin coincides continuously with the current landing site (the frame rotates with the moon); the XG-axis is vertical; the ZG-axis lies in, or near, the plane containing the LM and the landing site and points forward; and the YG-axis completes the right-hand triad. Thus, the origin and orientation of the guidance frame are altered each time the landing site is redesignated. Guidance-frame unit vectors expressed in platform coordinates are the row vectors $\underline{CGP}_X$, $\underline{CGP}_Y$, $\underline{CGP}_Z$ of the matrix CGP.
3. Body coordinates. Variables in body coordinates are tagged B. These are the generally accepted LM coordinates. The XB-axis is in the direction of the nominal thrust vector, the ZB-axis is directed forward, and the YB-axis completes the right-hand triad. Body-frame unit vectors expressed in platform coordinates are the row vectors $\underline{CBP}_X$, $\underline{CBP}_Y$, $\underline{CBP}_Z$ of the matrix CBP.

From these definitions it is noted that if the LM lands at the nominal site at the nominal time in a nominal erect attitude, the three frames will be parallel at the instant of touchdown.

The following conventions are defined for orthogonal matrices:

* The LGC transfers state vectors in platform coordinates to an abort guidance system (AGS). The AGS requires the state to be expressed in a frame whose Z axis parallels the orbital plane of the command module.



X_P, Y_P, Z_P → Platform (Inertial Measurement Unit) Coordinates

X_G, Y_G, Z_G → Guidance Coordinates

X_B, Y_B, Z_B → Body (LM Spacecraft) Coordinates

Figure 4 Lunar-descent Coordinate Frames

- 1) A matrix element is denoted by two subscripts which indicate the row and column respectively of the element. Thus CBP_{XY} denotes the Y-component of the row vector $\underline{CBP}_X$.
- 2) A matrix transpose (inverse) is denoted by interchanging tags.
- 3) From the definitions, it follows that matrix products are obtained by deleting internal tags.

By conventions 2 and 3, a vector $\underline{V}$ is transformed to body from guidance coordinates by

$$\underline{V}_B = CBP \ CGP^{-1} \ \underline{V}_G = CBP \ CPG \ \underline{V}_G = \underline{CBG} \ \underline{V}_G.$$

LM attitude angles are a set of Euler angles defined as clockwise rotations about the XB-axis (yaw), the displaced YB-axis (pitch), and the displaced ZB-axis (roll).

LM gimbal angles are a set of Euler angles defined as clockwise rotations about the YB-axis (inner), the displaced ZB-axis (middle), and the displaced XB-axis (outer).

BRAKING-PHASE AND APPROACH-PHASE GUIDANCE

The guidance programs for P63 and P64 are almost identical. The two phases use the same guidance algorithm, the same Throttle Routine, and the same Powered-flight Attitude-maneuver Routine. The differences are 1) the guidance equation selects different sets of targets, 2) the erection of the guidance coordinate frame is slightly different, and 3) landing site redesignation capability is available only in P64.

Guidance Equation Derivation

To guide a spacecraft from any initial or current state to a specified target state can be viewed either as an explicit guidance problem or as an implicit guidance problem. Explicitly, we can repetitively determine, as the mission progresses, a vector polynomial function of time that intersects the current and target states. Guidance then commands the corresponding profile of acceleration vs time. Implicitly, we can define, in advance of the mission, a reference trajectory as a vector polynomial function of time that evolves backward from the target state but cannot be expected to intersect a dispersed initial (or dispersed current) state. Onboard guidance then commands an acceleration vector profile composed of three terms, namely the acceleration along the reference trajectory minus two feedback terms proportional to the deviations in velocity and position of the actual trajectory with respect to the reference trajectory. In either the explicit or the implicit case, repetitively solving the guidance problem produces convergence upon the specified target state even though the target point may be redesignated in flight and the commanded acceleration is not precisely achieved because of control errors.

The implicit guidance equation derived here is categorically superior to the explicit guidance equation because the explicit equation is merely a special case, as will be shown. Besides being intellectually more satisfying, the implicit equation has demonstrated in simulations significantly faster reduction of deviations from the reference trajectory. Deviations come from navigation errors and from displacing the reference trajectory to intersect a redesignated landing site. Rapid reduction of deviations restores a nominal approach to the redesignated landing site. Unfortunately, the implicit equation had not been developed when the program for Apollo 11 was coded, and the advantages were insufficient to recode the guidance program for later missions.

Cherry⁽³⁾ derived the explicit guidance equation. Klumpp^(4,5) simplified it for LGC coding and generalized it to nth order. Moore et al⁽⁶⁾ derived an implicit equation which did not generalize the explicit equation. The general implicit guidance equation is now derived and specialized to the explicit case.

It is convenient to think of the reference trajectory as evolving backwards in time from the target point, with the time variable T reaching zero at the target point and negative prior to that point. Thus target-referenced time (T) is to be distinguished from clock-time (t). Because guidance gains would become unbounded, the target point is never reached. Instead, a guided phase is terminated at a negative time T and the succeeding phase is started. Both the terminus and the target point lie on the reference trajectory, but the target point lies beyond the portion that is actually flown, similar to a suggestion of McSwain and Moore.⁽⁷⁾

In terms of a vector polynomial function of target-referenced time, we wish to define a reference trajectory that satisfies a two-point boundary value problem with a total of five degrees of freedom for each of the three components. This number of degrees of freedom is required in order to constrain terminal thrust in P63 and to shape the trajectory design in P64, as is discussed in connection with the targeting program.

A quartic polynomial is the minimum order with which five constraints on the reference trajectory can be satisfied. With the reference trajectory evolving backwards in time from the target point, it can be defined as

$$\underline{RRG} = \underline{RTG} + \underline{VTG} T + \underline{ATG} T^2/2 + \underline{JTG} T^3/6 + \underline{STG} T^4/24, \quad (1)$$

where $\underline{RRG}$ is the position vector on the reference trajectory in guidance coordinates at the negative time T , and $\underline{RTG}$, $\underline{VTG}$, $\underline{ATG}$, $\underline{JTG}$, and $\underline{STG}$ are the target position, velocity, acceleration, jerk, and snap.

The acceleration to be commanded at any point in space consists of three terms: the acceleration of the reference trajectory at the particular time T , minus two feedback terms proportional to velocity and position deviations from the reference trajectory. Taking derivatives of Eq. (1) as the velocity and acceleration on the reference trajectory yields the three-term guidance equation

$$\begin{aligned} \underline{ACG} = & \underline{ATG} + \underline{JTG} T + \underline{STG} T^2/2 \\ & - (\underline{VG} - \underline{VTG} - \underline{ATG} T - \underline{JTG} T^2/2 - \underline{STG} T^3/6) K_V/T \\ & - (\underline{RG} - \underline{RTG} - \underline{VTG} T - \underline{ATG} T^2/2 - \underline{JTG} T^3/6 - \underline{STG} T^4/24) K_R/T^2, \end{aligned} \quad (2)$$

where $\underline{ACG}$ is the commanded acceleration, $\underline{VG}$ and $\underline{RG}$ are the current velocity and position, and K_V and K_R are the nondimensional feedback gains.

Combining like terms in Eq. (2) yields

$$\begin{aligned}
\underline{A}CG = & - \underline{R}G K_R / T^2 - \underline{V}G K_V / T \\
& + \underline{R}TG K_R / T^2 \\
& + \underline{V}TG (K_V + K_R) / T \\
& + \underline{A}TG (1 + K_V + K_R / 2) \\
& + \underline{J}TG (1 + K_V / 2 + K_R / 6) T \\
& + \underline{S}TG (1 / 2 + K_V / 6 + K_R / 24) T^2.
\end{aligned} \tag{3}$$

Equation (3) is the implicit guidance equation. Although the reference trajectory is quartic, the trajectory generated by the implicit guidance equation is obviously not. The implicit equation can, however, be specialized to the explicit equation — which does generate a quartic — by a specific choice of the feedback gains K_V and K_R . First we note that Eq. (2) may be identified with the linear second-order differential equation

$$\ddot{X} + 2 \zeta \omega_n \dot{X} + \omega_n^2 X = 0$$

by the associations (noting T is negative)

$$K_V / T = -2 \zeta \omega_n, K_R / T^2 = \omega_n^2, \tag{4}$$

where ω_n is the undamped natural frequency and ζ is the damping ratio. Of course the system is time varying. However, this association does afford some intuition on gain setting. Solving Eqs. (4) yields

$$K_R = (2\pi T/P)^2, \tag{5}$$

where P is the undamped period $2\pi/\omega_n$, and

$$K_V = -2 \sqrt{K_R} \zeta. \tag{6}$$

Equation (5) provides a means of controlling the system response time in terms of the nondimensional ratio P/T , and Eq. (6) provides a means of setting the damping ratio.

An interesting set of values to choose for response and damping is

$$P/T = -\pi/\sqrt{3}, \quad \zeta = \sqrt{3}/2.$$

This choice yields

$$K_R = 12, K_V = -6.$$

When these values are substituted into Eq. (3), the result is the explicit guidance equation derived in references 3 to 5,

$$\underline{ACG} = 12 (\underline{RTG} - \underline{RG})/T^2 + 6 (\underline{VTG} + \underline{VG})/T + \underline{ATG}. \quad (7)$$

The discussion of implicit vs explicit guidance is concluded by introducing the concept of a space containing all permissible combinations of guidance parameters. Implicit guidance-parameter space is one quadrant of the $\zeta, P/T$ plane or, equivalently, one quadrant of the K_R, K_V plane. Explicit guidance-parameter space is a single point in either plane.

Equation (7) presents the explicit guidance equation assuming negligible transport time delay. The explicit equation programed in Apollo is corrected to command an acceleration appropriate for the time at which the acceleration is predicted to be achieved. Let this predicted target-referenced time be

$$T_P = T + \text{LEADTIME}$$

where LEADTIME is the transport delay due to computation and command execution. An explicit guidance equation will now be derived that fits a quartic polynomial through the target position, velocity, and acceleration, and through the current position and velocity. The acceleration of the quartic at the predicted time T_P is the acceleration to command at the current time T in order to realize the quartic profile. It will be shown that the resulting guidance equation reduces to Eq. (7) for $T_P = T$, and therefore Eq. (7) generates a quartic profile when the transport delay is zero.

Constraining the actual trajectory to be a quartic function of time allows the current position and velocity to be expressed as

$$\begin{bmatrix} \underline{\mathbf{R}}\mathbf{G} \\ \underline{\mathbf{V}}\mathbf{G} \end{bmatrix} = \begin{bmatrix} 1 & T & T^2/2 & T^3/6 & T^4/24 \\ 0 & 1 & T & T^2/2 & T^3/6 \end{bmatrix} \begin{bmatrix} \underline{\mathbf{R}}\mathbf{T}\mathbf{G} \\ \underline{\mathbf{V}}\mathbf{T}\mathbf{G} \\ \underline{\mathbf{A}}\mathbf{T}\mathbf{G} \\ \underline{\mathbf{J}}\mathbf{T}\mathbf{G}\mathbf{A} \\ \underline{\mathbf{S}}\mathbf{T}\mathbf{G}\mathbf{A} \end{bmatrix} \quad (8)$$

where $\underline{\mathbf{J}}\mathbf{T}\mathbf{G}\mathbf{A}$ and $\underline{\mathbf{S}}\mathbf{T}\mathbf{G}\mathbf{A}$ are the jerk and snap which would be achieved at the target point, and are not targets loaded into LGC memory. Solving Eq. (8) for the jerk and snap yields

$$\begin{bmatrix} \underline{\mathbf{J}}\mathbf{T}\mathbf{G}\mathbf{A} \\ \underline{\mathbf{S}}\mathbf{T}\mathbf{G}\mathbf{A} \end{bmatrix} = \begin{bmatrix} -24/T^3 & -18/T^2 & -6/T & 24/T^3 & -6/T^2 \\ 72/T^4 & 48/T^3 & 12/T^2 & -72/T^4 & 24/T^3 \end{bmatrix} \begin{bmatrix} \underline{\mathbf{R}}\mathbf{T}\mathbf{G} \\ \underline{\mathbf{V}}\mathbf{T}\mathbf{G} \\ \underline{\mathbf{A}}\mathbf{T}\mathbf{G} \\ \underline{\mathbf{R}}\mathbf{G} \\ \underline{\mathbf{V}}\mathbf{G} \end{bmatrix} \quad (9)$$

The acceleration to be commanded at the current time T and realized at the predicted time T_P is

$$\underline{\mathbf{A}}\mathbf{C}\mathbf{G} = \underline{\mathbf{A}}\mathbf{T}\mathbf{G} + \underline{\mathbf{J}}\mathbf{T}\mathbf{G}\mathbf{A} T_P + \underline{\mathbf{S}}\mathbf{T}\mathbf{G}\mathbf{A} T_P^2/2. \quad (10)$$

Substituting Eq. (9) into Eq. (10) and simplifying yields the Apollo lunar-descent guidance equation

$$\begin{aligned} \underline{\mathbf{A}}\mathbf{C}\mathbf{G} = & \left[3 \left(\frac{T_P}{T} \right)^2 - 2 \left(\frac{T_P}{T} \right) \right] 12 (\underline{\mathbf{R}}\mathbf{T}\mathbf{G} - \underline{\mathbf{R}}\mathbf{G})/T^2 + \left[4 \left(\frac{T_P}{T} \right)^2 - 3 \left(\frac{T_P}{T} \right) \right] 6 \underline{\mathbf{V}}\mathbf{T}\mathbf{G}/T \\ & + \left[2 \left(\frac{T_P}{T} \right)^2 - \left(\frac{T_P}{T} \right) \right] 6 \underline{\mathbf{V}}\mathbf{G}/T + \left[6 \left(\frac{T_P}{T} \right)^2 - 6 \left(\frac{T_P}{T} \right) + 1 \right] \underline{\mathbf{A}}\mathbf{T}\mathbf{G}. \end{aligned} \quad (11)$$

Note that when time T is large in magnitude compared to the transport delay, T_P/T approaches unity, all bracketed coefficients in Eq. (11) approach unity, and Eq. (11) approaches Eq. (7) identically. The net effect of Eq. (11) not achieved by Eq. (7) is a gain reduction as the target point is approached. Because Eqs. (7) and (11) are identical for $T_P = T$, Eq. (7) generates a quartic profile when the transport delay is zero.

In the derivation of the guidance Eq. (7) or (11), nothing constrained the time T. At any point in a guided phase, T could be set to any arbitrary negative value, and Eq. (7) or (11) would satisfy the boundary-value problem from that point forward. Landing-site redesignation, which can arbitrarily stretch or shrink the trajectory, would produce an unnecessarily severe guidance response if T were not correspondingly adjusted. Because T is arbitrary, it can be computed to satisfy an additional boundary constraint. In Apollo, this additional constraint is imposed on the downrange (Z) component of jerk. Thus the Z-component of the jerk polynomial of Eq. (9) is solved for T by using a target Z-component of jerk JTG_Z . Separating this scalar cubic polynomial from Eq. (9) yields

$$JTG_Z T^3 + 6 ATG_Z T^2 + (18 VTG_Z + 6 VG_Z) T + 24 (RTG_Z - RG_Z) = 0. \quad (12)$$

One root of this cubic is the required time T.

An alternate criterion for computing T reduces the propellant-consumption penalty of downrange landing-site redesignations. Although extensively tested, the alternate was developed too late for incorporation in the LGC program. The alternate criterion sets the downrange position error to zero. Thus T is one root of the quartic

$$STG_Z T^4/24 + JTG_Z T^3/6 + ATG_Z T^2/2 + VTG_Z T + RTG_Z - RG_Z = 0.$$

Braking-Phase Targeting Objectives

The near-optimal transfer provided by P63 targeting must satisfy a throttling constraint that the DPS be operated within the permitted-thrust region for, nominally, the final two minutes of the phase. This throttling duration absorbs dispersions in DPS performance and errors in lunar terrain modeling. With a total propellant consumption of over 6 600-kg, Yang⁽⁸⁾ shows that the Apollo guidance and targeting are within 16-kg of an optimal trajectory satisfying the same throttling constraint.

To provide thrust within the 11 to 65% region for the last two minutes of P63, the targets are chosen to produce a constant thrust level of about 57% of rated thrust at P63 terminus. The targeting program accomplishes this by constraining the magnitude of the terminal thrust-acceleration vector to be F/M , and constraining the Z-component of terminal jerk to be (essentially) $K F \dot{M}/M^2$, where F is the required terminal thrust, M is the estimated terminal mass, $\dot{M}$ is the estimated terminal mass flow rate (negative), and K is a jerk coefficient to account for the vertical component of thrust. The targeting program achieves the two minute duration of constant thrust by adjusting the initial range.

Properly targeted, the guidance algorithm commands during most of P63 a thrust-acceleration in excess of what can be achieved. The throttle routine multiplies this thrust-acceleration command by the estimated mass to yield the guidance thrust

command (GTC), and provides maximum thrust until the GTC falls below 57%. Figure 5 illustrates the profiles of GTC and actual thrust for a properly targeted P63 phase and shows the effects of adjustments of the ignition time and of the DPS thrust level.

The targeting program computes the P63 targets by projecting computed terminal conditions forward typically 60 seconds. Although the targets are projected, they are computed to produce the required terminal conditions on a nominal trajectory. Trajectory dispersions cannot be eliminated prior to the target point, but they can be reduced sufficiently by terminus to achieve the targeting objectives. Both the ignition time and the projected targets are computed iteratively using a descent simulation in the iteration loop.

Approach-Phase Targeting Objectives

The P64 targets are computed to provide lunar-surface visibility until about 5 seconds before terminus, and continuous throttling. In addition, the P64 targets are computed to produce at terminus a matched set of values for the Z-components of acceleration, velocity, and position such that, in the nominal case, the P66 algorithm will produce no initial pitch transient and will simultaneously null the Z-components of velocity and position. Unlike P63, the P64 reference trajectory can be determined in closed form from specified trajectory constraints. Thus the projected targets are computed without numerical iteration.

Landing-Site Redesignation Procedure

To steer the LM via the automatic P64 guidance to a visually selected landing site, the commander uses an iterative procedure akin to steering an automobile. The procedure consists of 1) identifying the current landing site where the LGC would take him in the absence of intervention and 2) steering the current site into coincidence with his visually selected site by commanding incremental landing-site displacements (redesignations). Because the P64 targets are defined in the guidance coordinate frame, which is repetitively erected through the landing site, the P64 target point is displaced accordingly.

To identify the currently selected landing site to the astronauts, the LGC 1) orients the LM about the thrust axis to superimpose the landing point designator (LPD) reticles (see Figure 1) on the current site and 2) displays a number which is read by the LM pilot and vocally relayed to the commander. By sighting through the indicated point of the LPD reticles, the commander identifies the current site. He registers his eye by superimposing the two LPD reticles, one of which is painted on the inside window panel, and one on the outside window panel. The separation between reticles is 2.5 cm.

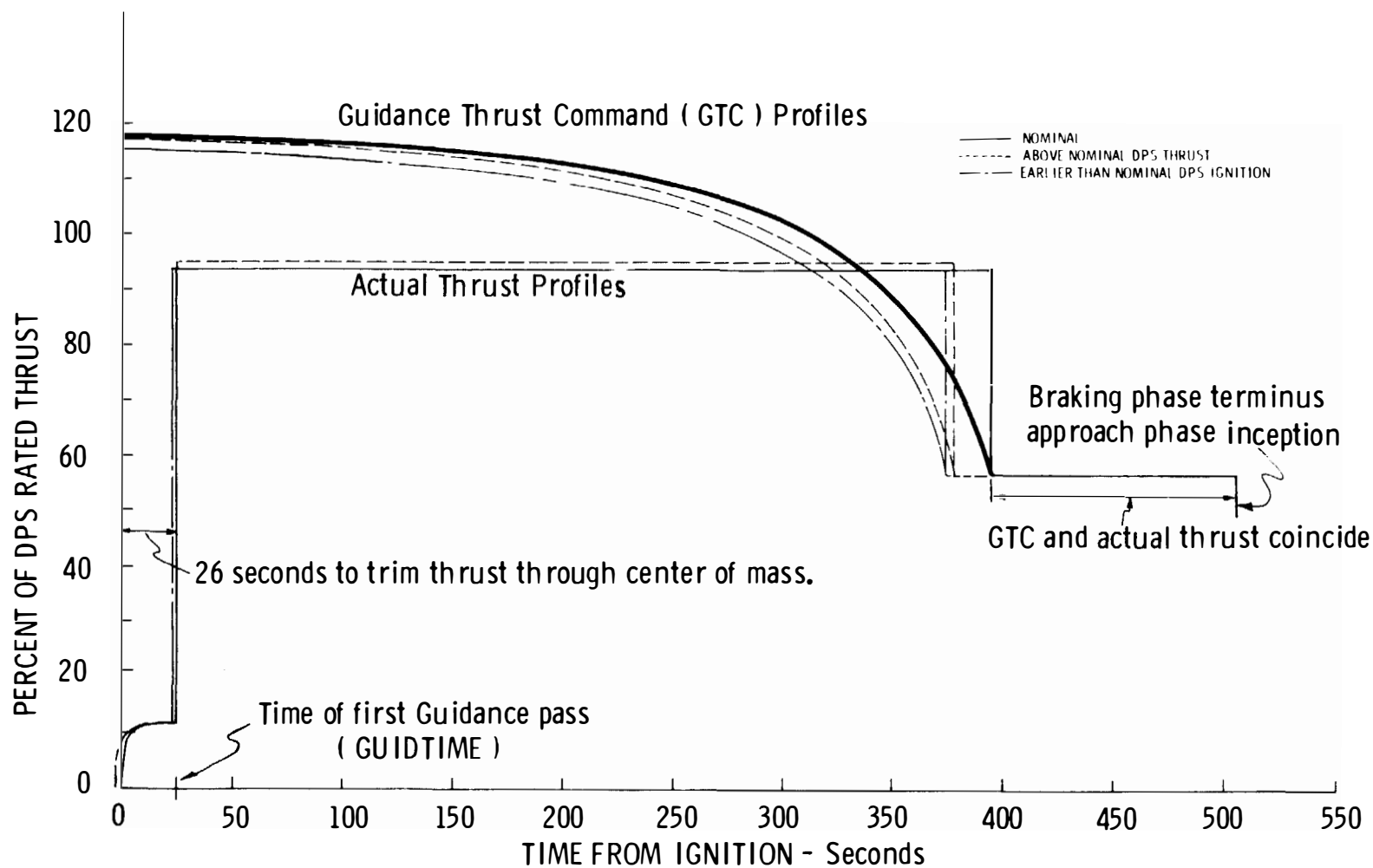


Figure 5 Actual Thrust Profiles and Guidance Thrust Command Profiles During P63

By manipulating his controller left, right, forward, or aft, the commander directs the LGC to displace the landing site (and the P64 targets) along the lunar surface by a correspondingly directed fixed angular increment (1°) with respect to the current line of sight.

The LGC redirects the thrust to guide to the redesignated (now current) site, and reorients about the thrust axis to maintain superposition of the reticles on the current site. The commander can continue this redesignation process — steering the current landing site into coincidence with his chosen site — until 10 seconds before reaching the P64 target point, at which time P66 is initiated.

P63,P64 Guidance Algorithm

Figure 6 illustrates the P63,P64 Guidance Algorithm. As shown in Figure 3, the algorithm receives guidance targets, the current state vector, and the current gravity vector as inputs and issues a thrust-acceleration command, a unit thrust command, and a unit window command as outputs.

Because the landing site moves due to lunar rotation and landing-site redesignation, the LM is guided with respect to the guidance coordinate frame, which is erected through the landing site each pass. Guidance targets are fixed in this floating frame. Other inputs and all outputs are expressed in platform coordinates.

The landing site vector $\underline{L_P}$ is updated for lunar rotation (Eq. (6.1)) using an approximate algorithm that avoids computation of trigonometric functions, yet preserves the magnitude of the lunar radius. The algorithm accounts for the lunar rotation rate $\underline{W_{MOONP}}$ and the elapsed clock-time since the preceding update ($t - t_{OLD}$).

For the landing-site redesignation algorithm (Eqs. (6.2) – (6.7)), whenever the commander manipulates the controller (Figure 1) in the automatic mode, the LGC is interrupted and the azimuth command count (NCAZ) or the elevation command count (NCEL) is incremented or decremented according to the direction of manipulation. The redesignation algorithm fetches and resets to zero the NCAZ and NCEL accumulators and rotates $\underline{L_{OSP}}$ (the unit line-of-sight vector to the current landing site) by 1° per count (Eq. (6.3)). If NCAZ and NCEL are both zero, the redesignation algorithm has no effect. Given that attitude control maintains coincidence of the ZB,XB plane and $\underline{L_{OSP}}$, the rotations of $\underline{L_{OSP}}$ are about two axes normal to $\underline{L_{OSP}}$. Elevation redesignations rotate $\underline{L_{OSP}}$ about the YB-axis, and azimuth redesignations rotate $\underline{L_{OSP}}$ about an axis normal to $\underline{L_{OSP}}$ in the ZB,XB plane. The landing-site redesignation geometry shown in Figure 7 depends upon the defined LM platform orientation, namely that the XP-axis is near vertical through the landing site. The constraint that L_{OSP_X} be at least as negative as -0.02 (Eq.

(6.5)) prevents redesignating the landing site beyond the horizon. Equation (6.7) computes the displaced point near the surface shown in Figure 7 and places the redesignated site directly beneath this point.

The displayed LPD angle (θ_{LPD} , Eq. (6.8)) is the angle between $\underline{LOSP}$ and the ZB-axis.

The computation of the state vector in guidance coordinates (Eq. (6.9)) places the origin of the guidance frame at the landing site and yields the velocity of the LM relative to the lunar surface.

Target-referenced time T is computed using Newton's method starting with a good estimate (Eqs. (6.12) – (6.13)). Note that the denominator of Eq. (6.12) is the derivative of the numerator.

The guidance equation (Eq. (6.15)) is identical to Eq. (11). The thrust-acceleration command (Eqs. (6.16) – (6.17)) is merely the total acceleration command minus current gravity $\underline{GP}$, and the unit thrust command (Eq. (6.18)) is the direction of the thrust-acceleration command. The so-called radial-acceleration guidance correction described in reference 9 is rendered unnecessary by current targeting techniques and is omitted from this report, although present in the LGC program.

The computation of the unit window command presented here is a simplification of the LGC coding which produces the same result. The object is to keep the landing site in the center of vision (superimpose the LPD reticles on the current site) whenever the geometry permits and, otherwise, to command a forward-facing attitude. Figure 8 shows why the landing site cannot always be kept in the center of vision. Figure 9 shows the geometry pertinent to computation of the unit window command $\underline{UNWCP}$. Commanding the line-of-sight vector ($\underline{UNWCP} = \underline{LOSP}$) aligns the reticles with the landing site; commanding the forward vector ($\underline{UNWCP} = \underline{FORP}$) produces a forward facing attitude. If the first alternative is chosen ($\underline{UNWCP} = \underline{LOSP}$) the LM will rotate about the XB-axis to align the YB-axis with the vector $\underline{LOSP} \times \underline{CBP}_X$. Thus the direction of $\underline{LOSP} \times \underline{CBP}_X$ indicates whether a normal forward-facing attitude or an abnormal attitude would result from the command $\underline{UNWCP} = \underline{LOSP}$. In addition, the magnitude of $\underline{LOSP} \times \underline{CBP}_X$ measures the degree of indeterminacy in the command $\underline{UNWCP} = \underline{LOSP}$. The projection (PROJ, Eq. (6.20)) of $\underline{LOSP} \times \underline{CBP}_X$ on the YG-axis detects both the magnitude and the direction of $\underline{LOSP} \times \underline{CBP}_X$. Thus PROJ is used as the criterion for mixing $\underline{LOSP}$ and $\underline{FORP}$ into $\underline{UNWCP}$. If the descent trajectory is planar, the mixing (Eq. (6.21)) yields $\underline{UNWCP} = \underline{LOSP}$ for $\theta_{LPD} \leq 65^\circ$, $\underline{UNWCP} = \underline{FORP}$ for $\theta_{LPD} \geq 75^\circ$, and $\underline{UNWCP}$ a mixture linear with $\cos \theta_{LPD}$ for $65^\circ < \theta_{LPD} < 75^\circ$. Regardless of whether the trajectory is planar or nonplanar, it is never possible to command a side-facing or a rear-facing attitude.

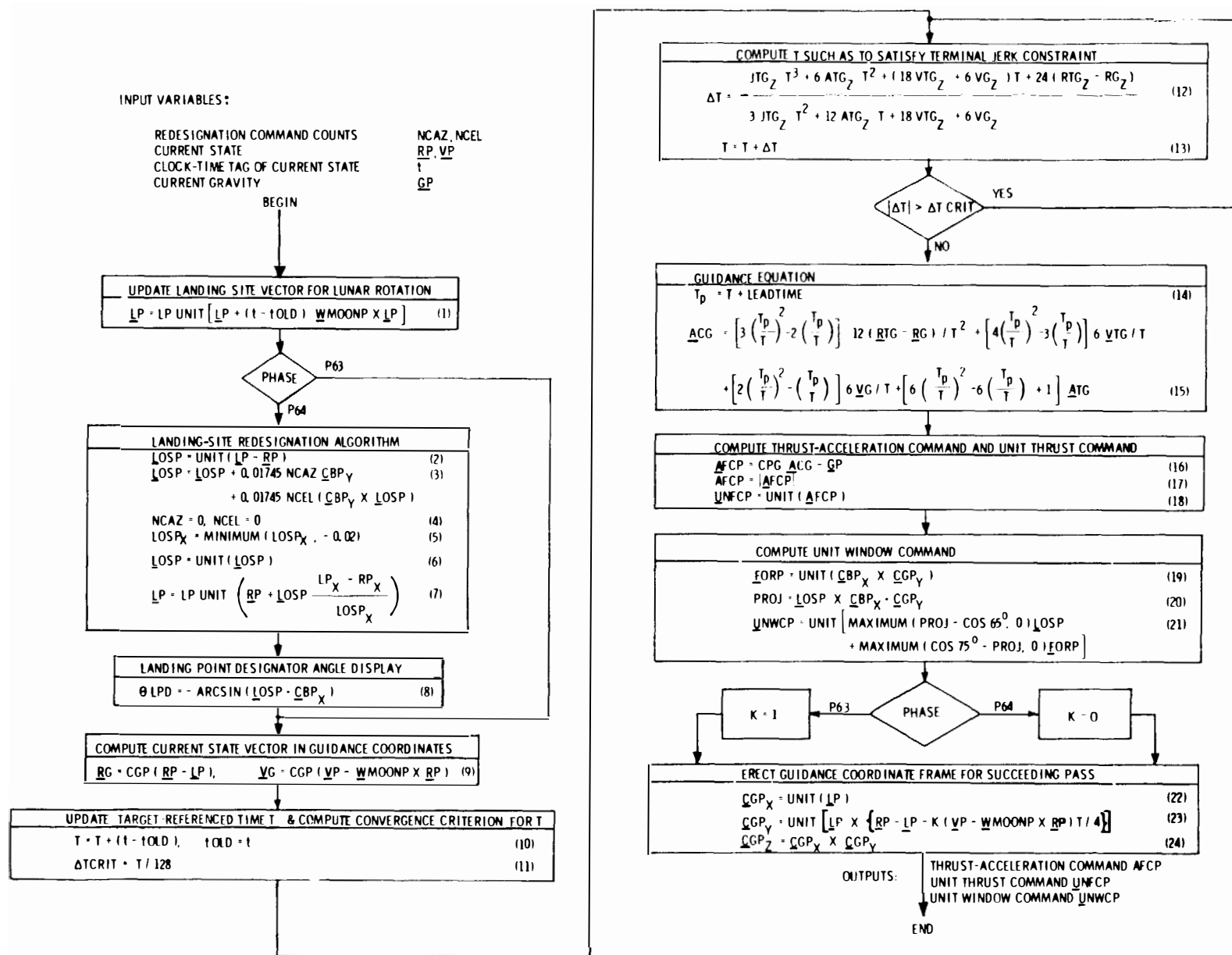


Figure 6 P63, P64 Guidance Algorithm

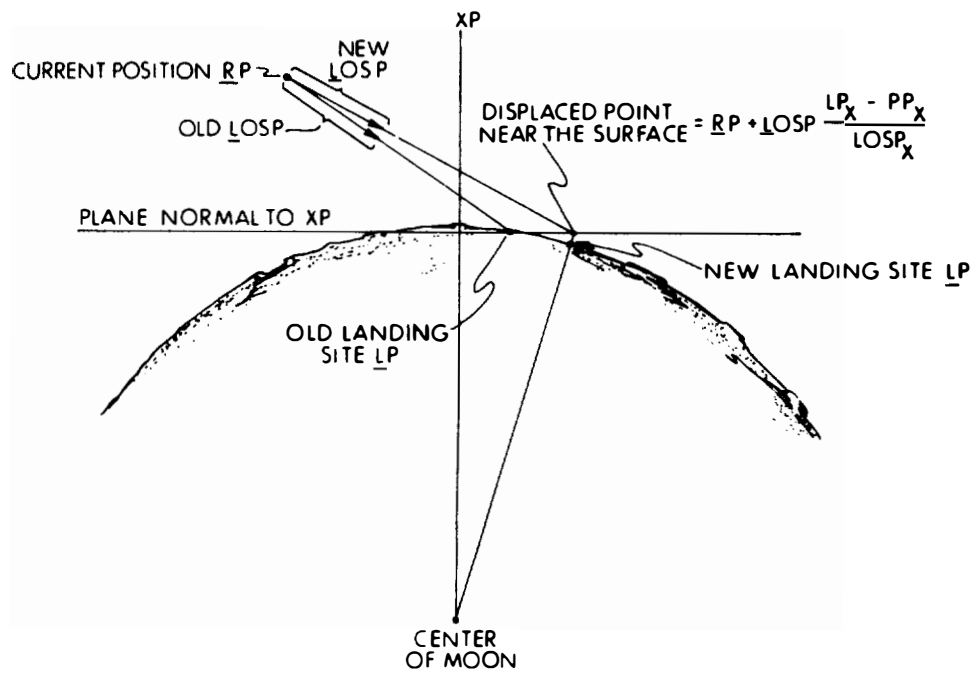


Figure 7 Landing-site Redesignation Geometry

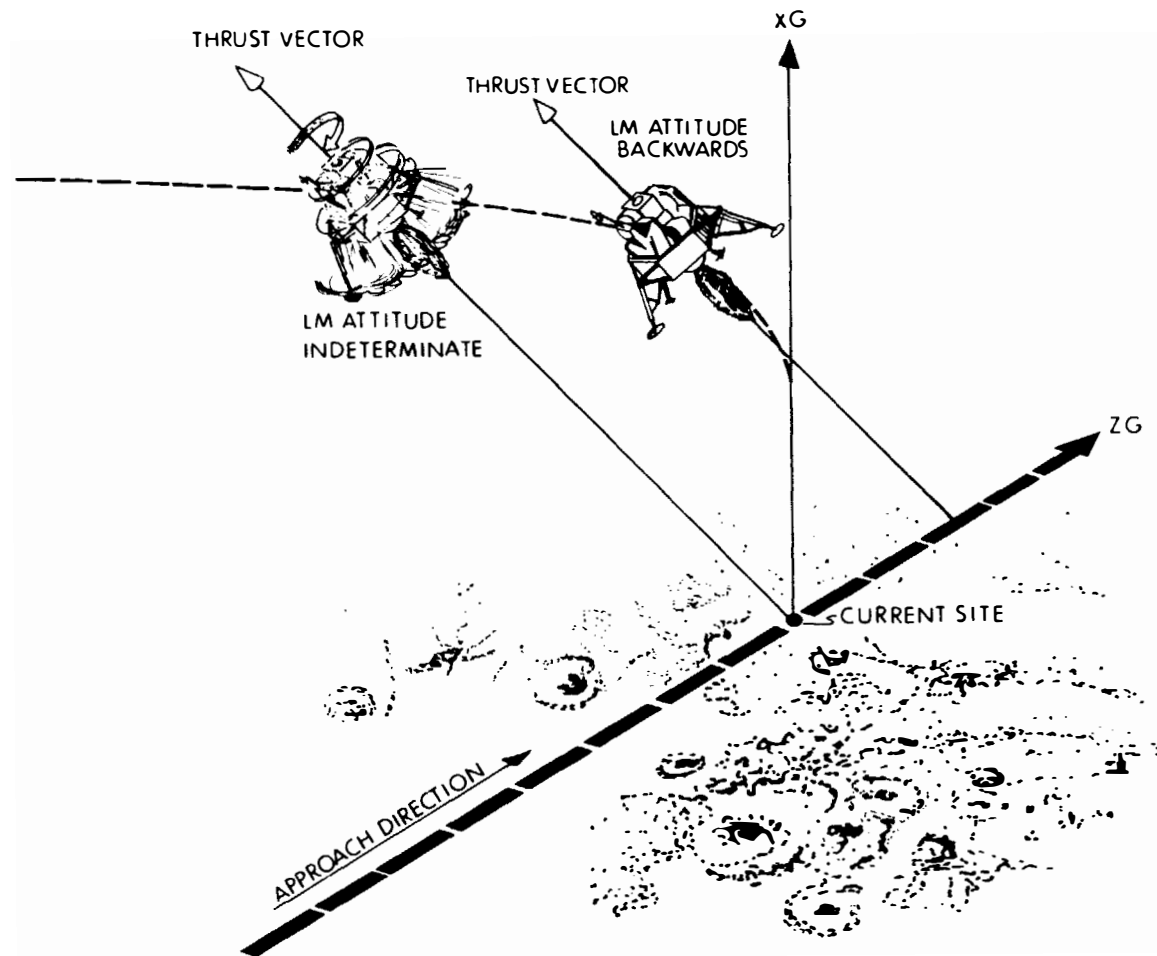


Figure 3 Why Keeping the Landing Site in the Center of Vision Cannot be the Sole Criterion for Controlling Attitude About the Thrust Axis

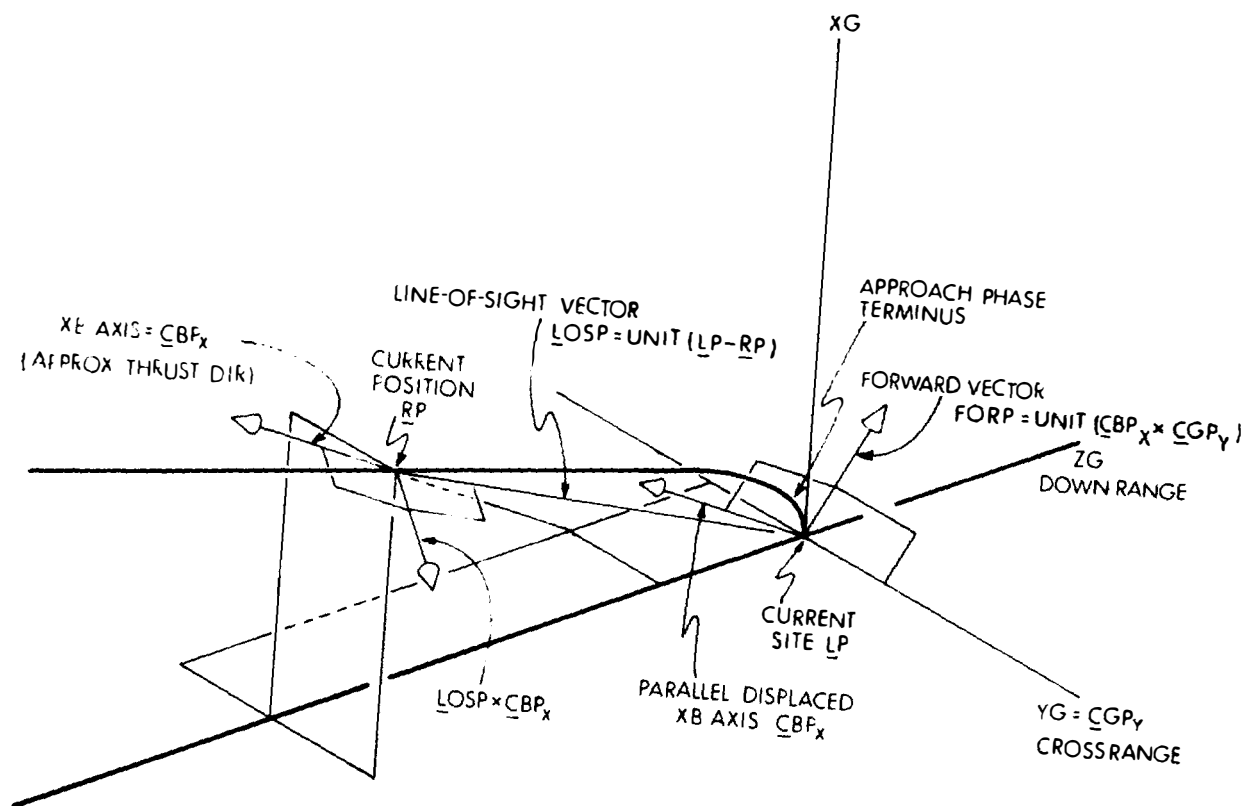


Figure 9 Geometry Pertinent to Computing the Unit Window Command

Erection of the guidance coordinate frame (Eqs. (6.22) – (6.24)) is illustrated in Figure 10. With $K = 1$ in P63, the guidance coordinate frame orientation about the vertical XG-axis is such that the YG-component of jerk would reach zero at the target point if the trajectory were flown there (see reference 5). With $K = 0$ in P64, the ZG-axis is in the vertical plane containing the line-of-sight vector. For crossrange landing-site redesignations, setting $K = 0$ in P64 was found to consume less DPS propellant than setting $K = 1$ to null the crossrange jerk at the target point.

P63 Ignition Algorithm

Trajectory dispersions preceding P63 require an accurate ignition time and attitude to be computed to 1) avoid excessive variations of the time duration of throttle control in P63 and 2) to avoid commanding an excessive attitude transient the first time the P63,P64 Guidance Algorithm is processed. The P63 ignition procedure consists of:

- 1) Computing onboard the precise ignition time and attitude about 10 minutes in advance of ignition
- 2) Orienting the LM to the ignition attitude
- 3) Initiating reaction control system ullage 7.5 seconds prior to ignition
- 4) Igniting the DPS at minimum thrust and holding constant thrust and constant attitude for 26 seconds, the maximum time required for the DAP to orient the DPS trim gimbal system to point the thrust vector through the LM center of mass
- 5) Connecting the guidance algorithm, which immediately commands maximum thrust and begins commanding an attitude profile according to the current state vector and the P63 targets.

To determine the required ignition attitude, the ignition algorithm (Figure 11) calls the guidance algorithm as a subroutine. The ignition algorithm supplies inputs consisting of an accurate extrapolation of the state vector and the corresponding gravity vector (both valid at GUIDTIME, the estimated clock-time of the first P63 guidance pass). In preparation, Eqs. (11.1) – (11.5) initialize guidance algorithm inputs. On the first iteration, the state vector extrapolation represented by Eq. (11.6) is performed by an orbital integration routine and, on subsequent iterations, by a Kepler routine. Equation (11.9) corrects the extrapolated velocity vector by the velocity increment imparted during the 26 seconds of minimum thrust preceding this point. (The errors due to not correcting the extrapolated position vector and not correcting for ullage are negligible.) The guidance algorithm produces a unit

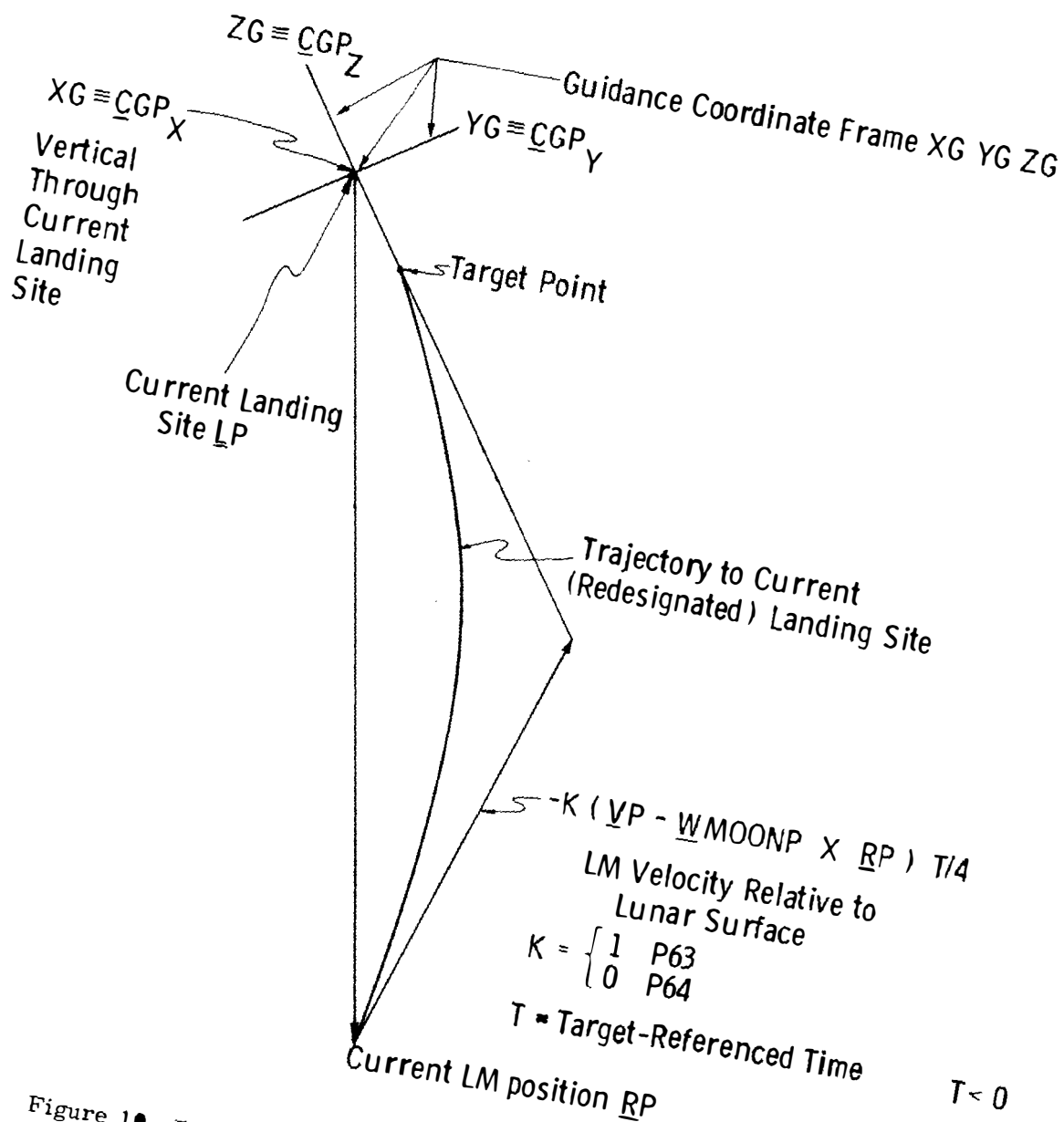


Figure 1● Plan View of Guidance-coordinate-frame Erection

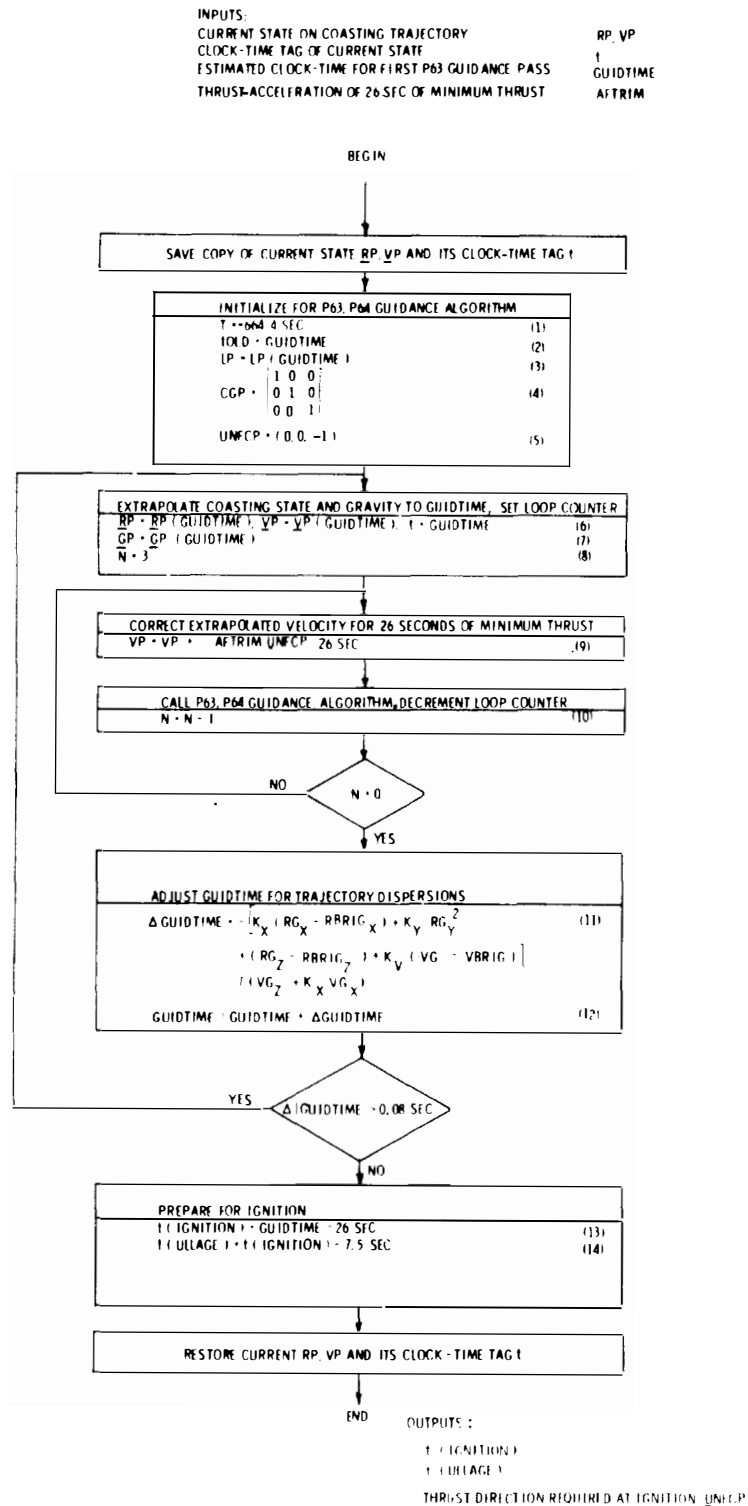


Figure 11 P63 Ignition Algorithm

thrust command $\underline{UNFCP}$, which is the direction to point the XB-axis. Because the direction of the velocity correction is unknown on the first iteration, the above procedure is iterated thrice.

An outer ignition-algorithm loop accounts for dispersions with respect to the nominal trajectory. Equations (11.11) - (11.12) adjust GUIDTIME to correct the RG_Z component of position at GUIDTIME as 1) a linear function of the dispersion in orbital speed VG and of the dispersion in the RG_X component of position (essentially altitude) and 2) as a quadratic function of the out-of-plane position RG_Y . $RBRIG_X$ and $RBRIG_Z$ are nominal initial altitude and range components of position in guidance coordinates; $VBRIG$ is the nominal initial speed; and K_X , K_Y , and K_V are correction coefficients. The nominal initial altitude, range, and speed are computed by the targeting program. The correction coefficients are computed using a manual procedure based on descent simulations.

When converged, this process yields a precise time and attitude for igniting the DPS. Trajectory dispersions result in typical variations of 2 seconds in the time duration of throttle control and typical attitude transients of 2 milliradians commanded by the guidance algorithm on the first P63 pass.

TERMINAL-DESCENT-PHASE GUIDANCE

Horizontal and vertical velocity are controlled in P66 by completely independent algorithms. P66 provides a nonautomatic attitude-hold mode in which the commander can control the LM attitude to translate or not, as he wishes, horizontally over the lunar surface. P66 issues no unit window command; yaw is controlled manually. A description of P66 including the nonautomatic modes is provided by Eyles.⁽¹⁰⁾

P66 Horizontal Guidance Algorithm

The P66 horizontal guidance algorithm (Figure 12), processed once every two seconds, nulls the horizontal components of velocity relative to the lunar surface by directing the thrust vector a small angle away from vertical in opposition to horizontal velocity. The horizontal algorithm neither measures nor commands thrust-acceleration magnitude; the algorithm is derived on the assumption that the vertical component of thrust-acceleration equals lunar gravity.

Just as velocity feedback damps a position control loop, acceleration feedback damps a velocity control loop. Because of the sampled-data character of the system, a good measure of current acceleration is the acceleration commanded the preceding pass. The P66 horizontal algorithm feeds back the velocity error (current velocity VP_Y, VP_Z minus lunar surface velocity V_{MOONP_Y}, V_{MOONP_Z}) and, to provide the required damping, feeds back a fraction of the thrust-acceleration command from the preceding pass (Eqs. (12.2) - (12.3)). On the first P66 pass, the thrust-acceleration fed back is that commanded the final P64 pass.

The direction of the thrust-acceleration command is limited to 20° from vertical (Eqs. (12.4) and (12.5)) to maintain a nearly erect LM attitude. The LIMIT function of two arguments limits the magnitude of the first argument to the value of the second argument.

The unit thrust command (Eq. (12.6)) is the direction of the limited thrust-acceleration command.

The assumption in generating horizontal commands that the vertical component of thrust-acceleration equals lunar gravity (Eq. (12.1)) is realized only if the LM is not accelerating vertically. The purpose of ignoring vertical acceleration is to eliminate coupling from ROD inputs to LM attitude. The effect of vertical acceleration, which occurs whenever the commander manipulates the ROD switch, is to modulate the gains of the horizontal channels. This gain modulation is negligible because

only limited changes in the descent rate will ever be commanded; the vertical acceleration can be significantly nonzero only for short periods of time.

P66 Vertical (ROD) Guidance Algorithm

The ROD guidance algorithm, processed once per second, controls altitude rate to the reference value by throttling the DPS. The ROD algorithm has no control over the LM attitude; the thrust-acceleration command it issues accounts for any non-vertical orientation of the thrust vector.

The object of the ROD guidance is to respond rapidly without overshoot to ROD increment commands. The algorithm provides a time constant of 1.5 seconds, even though the sample interval is 1.0 second, by capitalizing on the sampled-data character of the system. Using a computed estimate of the total acceleration at the ROD sample instant, the ROD algorithm extrapolates sample-instant measured velocity by the effective transport lag of 0.35 second and thus commands an acceleration appropriate for the velocity error at the time the acceleration command will be realized. A sampled data analysis (reference 11) shows that the compensation for effective transport lag is highly effective in stabilizing the vertical channel. The significant system dynamics reduce to a single zero and two poles in the Z plane. The zero is

$$Z_Z = - \text{LAG}/(\text{sample interval} - \text{LAG}) = -0.35/(1 - 0.35) = -0.538.$$

One pole is at the origin, and the second pole is

$$Z_P = (\text{time constant} - \text{sample interval})/\text{time constant} = (1.5 - 1)/1.5 = 1/3.$$

The poles are the same as for an ideal system containing neither a transport lag nor an extrapolation.

The ROD algorithm has been simplified for this report as follows:

1. In the LGC coding, the ROD algorithm begins each pass by reading the accelerometers and recording the time at which they are read. This time is called the ROD sample instant. ROD sample instants occur irregularly, but the interval between them, called the ROD sample interval, averages 1 second. The accelerometer readings are used to compute a) the three-component current velocity vector valid at the ROD sample instant, based on updating the velocity vector supplied by the state vector update routine (SVUR, Figure 3), and b) a thrust-acceleration

measurement which is the average over the ROD sample interval. To compute the velocity vector it supplies, the SVUR also reads the accelerometers, each pass, at the SVUR sample instant occurring at regular 2-second intervals. The irregular ROD sample instants are essentially asynchronous with the regular SVUR sample instants. Consequently the interactions between the ROD algorithm and the SVUR in updating the SVUR-supplied velocity vector are extremely intricate. In this report, the data processed by the ROD algorithm are shown as inputs. How these inputs are obtained is described in reference 9.

2. Although the vertical orientation of the XP-axis is capitalized upon by several LGC routines, including the P66 horizontal algorithm and the landing-site redesignation algorithm, the LGC ROD algorithm laboriously manipulates complete-vector state data to maintain validity for any platform alignment. Presented here is the scalar equivalent valid for the lunar-landing platform alignment.

Figure 13 shows the ROD algorithm. The inputs are all valid at the ROD sample instant. Equation (13.1) computes the sample-instant total vertical acceleration by adding, to the thrust-acceleration measurement (averaged over the ROD sample interval), current gravity and a correction for the throttle change concluding the preceding ROD pass. The thrust correction increment δF_A is supplied by the Throttle Routine. Equation (13.2) extrapolates the sample-instant measured velocity. The commanded vertical velocity (reference altitude rate) is initialized as the vertical velocity existing at the time P66 is initiated, and is incremented or decremented by Eq. (13.3) each ROD pass according to the ROD commands issued by the commander since the preceding ROD pass. Equation (13.5) first computes the total vertical acceleration required as the negative of the extrapolated velocity error divided by the ROD time constant (1.5 seconds). The equation then obtains the required vertical thrust-acceleration by subtracting current gravity. Finally, dividing by CBP_{XX} , which is the cosine of the angle between the XB-axis and the vertical XP-axis, Eq. (13.5) yields the thrust-acceleration command AFCP. To avoid an empirically discovered instability which occurs when the throttle routine or the DPS cannot comply with the thrust-acceleration command from P66, Eqs. (13.6) and (13.7) restrict AFCP to produce thrust within the permitted-thrust region.

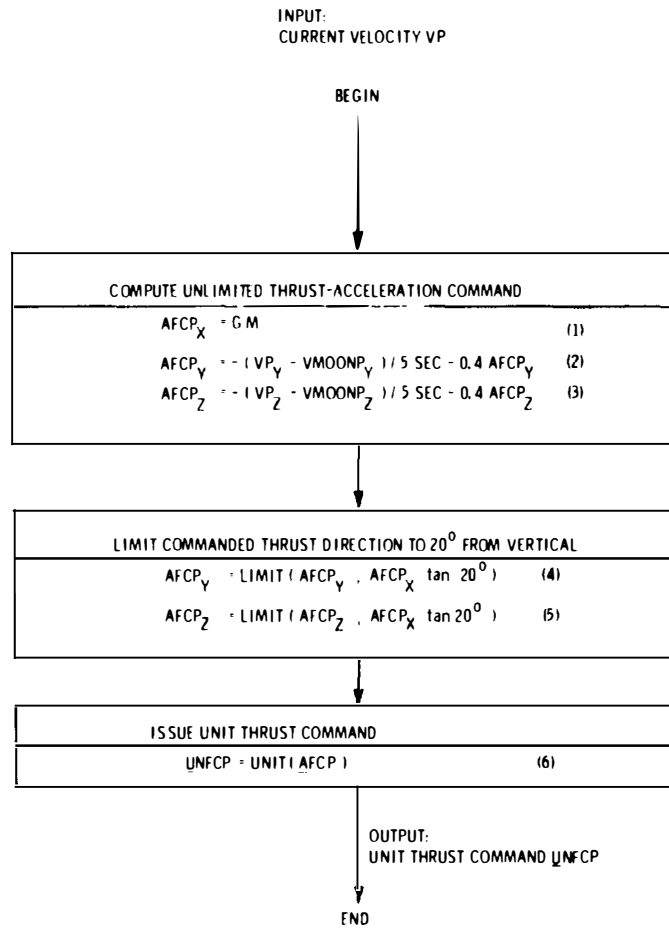


Figure 12 P66 Horizontal Guidance Algorithm

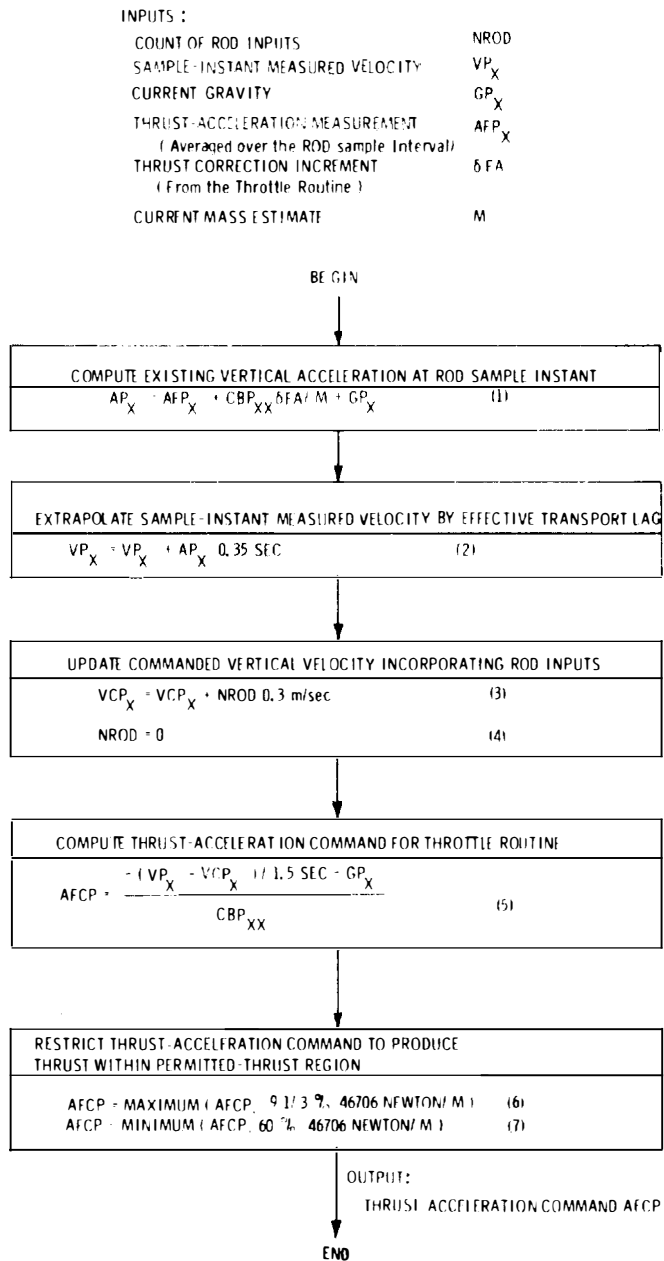


Figure 13 P66 Vertical (ROD) Guidance Algorithm

POWERED-FLIGHT ATTITUDE-MANEUVER ROUTINE

A link in the attitude control chain of command, the Powered-flight Attitude-maneuver routine (ATT) connects the various powered-flight guidance programs to the DAP. The functions of ATT are:

1. For the small attitude changes normally required each guidance cycle, ATT commands a maneuver of constant rate such as to achieve the required attitude 2 seconds later.
2. For gross attitude maneuvers which may be required at phasic interfaces or upon abort, ATT commands a rate-limited maneuver which may extend over several guidance cycles.
3. For all attitude maneuvers ATT avoids the gimbal-lock region (middle gimbal angle $> 70^\circ$ magnitude). ATT issues a gimbal-lock alarm code if and only if the commanded attitude computed from guidance inputs lies within the gimbal-lock region. ATT commands a maneuver which circumvents the gimbal-lock region and issues no gimbal-lock alarm code when the most direct path to the commanded attitude passes through the gimbal-lock region.

Switching from a descent program to an abort program may produce up to 180° change in commanded thrust direction. A break with traditional approaches, ATT makes gimbal lock during any maneuver inherently impossible by 1) computing commanded gimbal angles, 2) limiting the magnitude of the middle commanded gimbal angle, and 3) issuing to the DAP a series of incremental attitude-maneuver commands that monotonically* drive the gimbal angles from their current values to their commanded values. Provided the attitude is not currently in gimbal lock, and given that the middle commanded gimbal angle is magnitude limited at the gimbal-lock boundary, it is inherently impossible to maneuver through gimbal lock; the middle gimbal angle is confined to the range between its current and commanded values. Other attitude-maneuver schemes with appended gimbal-lock avoidance require more computation to produce similar maneuvers.

Figure 14 presents an overview of the LM powered-flight attitude control process, including some information on the procedures on the DAP side of the

* Except the outer gimbal angle profile may not be monotonic in the geometrically complex case of a large maneuver about multiple axes at substantial middle gimbal angle and with magnitude limiting of the X-axis attitude angle change on at least one pass through ATT.

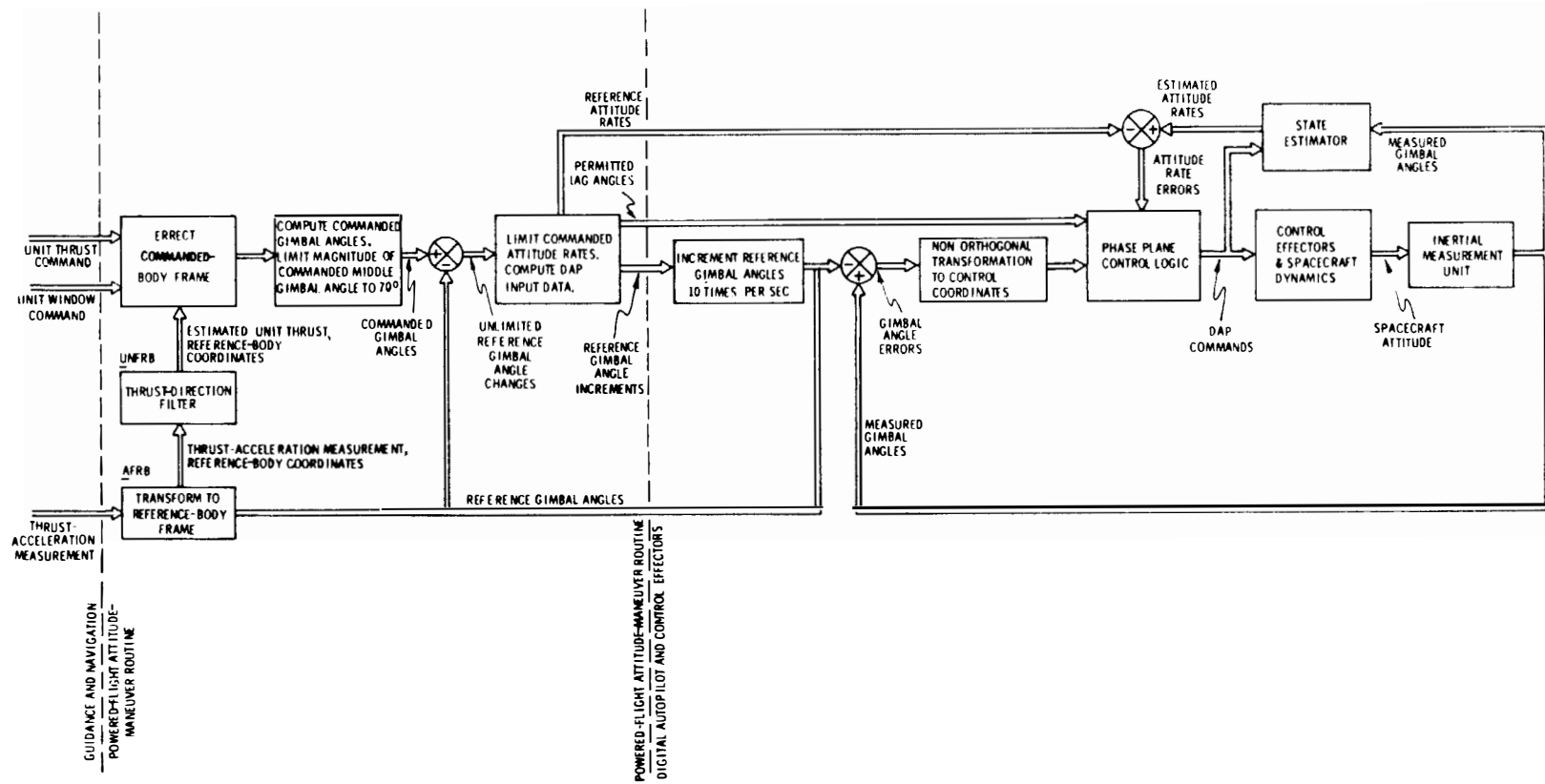


Figure 14 LM Powered-flight Attitude Control

interface. Two computational coordinate frames are introduced. From guidance and navigation inputs, ATT computes a commanded-body frame (tag CB) to represent the commanded attitude inherent in the input vectors. From ATT inputs, the DAP computes reference gimbal angles to compare with measured gimbal angles for computing the attitude errors. The reference gimbal angles define a reference-body frame (tag RB). ATT computes that the attitude errors are zero when these two computational coordinate frames coincide. Of course, there may be DAP control errors undetected by ATT, but any thrust pointing error is detected in the steady state by a thrust-direction filter, and corrected.

The guidance and navigation inputs to ATT, shown in Figure 14, consist of a unit thrust command, a unit window command, and a thrust-acceleration measurement. ATT processes the thrust-acceleration measurement in a thrust-direction filter to determine an estimated unit thrust vector with respect to the reference-body frame. Correcting for the offset of the estimated unit thrust vector with respect to the XRB-axis, ATT uses the unit thrust command and the unit window command to erect the commanded-body frame. From the commanded-body frame matrix, ATT extracts commanded gimbal angles which it compares with the reference gimbal angles to generate inputs to the DAP. Ten times per second, the DAP updates the reference attitude and generates the corresponding control commands. The dynamic response is sufficiently fast and tight that the reference attitude is a good measure of instantaneous spacecraft attitude.

A feature of this configuration is that, although ATT runs at a sample rate of 2 seconds, close to the fuel-slosh resonant frequency at certain points in the mission, it avoids exciting fuel slosh by avoiding all coupling with the actual spacecraft attitude except through the slow thrust-direction filter.

Figure 15 details the Powered-flight Attitude-maneuver routine. The thrust-direction filter computes the thrust-acceleration measurement in reference-body coordinates by constructing the required transformation from the reference gimbal angles (Eq. (15.1)). The change in thrust direction is limited on each cycle to 7-mr (Eqs. (15.3) and (15.4)), the maximum travel of the trim gimbal in 2 seconds. The total excursion of the estimated unit thrust vector is limited to 129-mr (Eqs. (15.5) and (15.6)), the mechanical excursion limit of the trim gimbal plus mechanical deflection and thrust offset with respect to the nozzle. The X-component of the estimated unit thrust vector is not needed and not computed.

If either 1) guidance provides a unit window command too closely aligned with the unit thrust command to adequately determine the attitude orientation about the

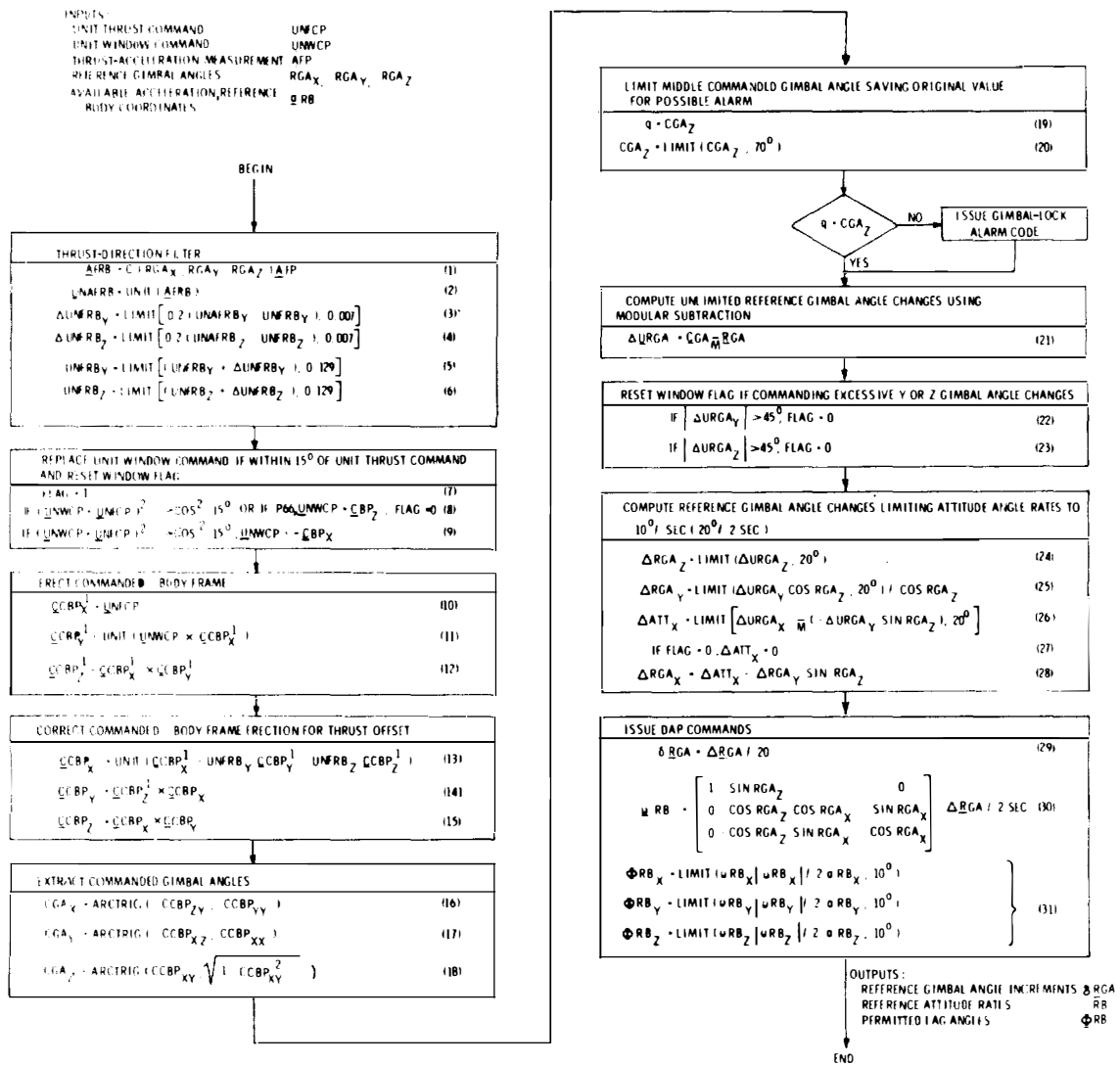


Figure 15 Powered-flight Attitude-maneuver Routine

XCB-axis, or 2) the guidance program is P66 (which provides no unit window command), then ATT provides a unit window command suitable for erection of the commanded-body frame and resets a flag to indicate that no attitude rotation is allowed about the XCB-axis. ATT first provides the current ZB-axis (Eq. (15.8)). But this choice may also be nearly collinear with the unit thrust command, so a second possibility, the current negative XB-axis, is also offered (Eq. (15.9)). Because the ZB- and XB-axes cannot both parallel the unit thrust command, no further checks need be made.

The matrix CCBP, whose row vectors are the commanded-body frame unit vectors expressed in platform coordinates, is computed to satisfy the unit thrust command, the unit window command, and the thrust offset (the angular displacement between the estimated unit thrust vector and the XRB-axis). CCBP is computed in two steps as illustrated in Figure 16. The first step (Eqs. (15.10)–(15.12)) uses the unit thrust command and the unit window command but fails to account for thrust offset. The second step (Eqs. (15.13)–(15.15)) corrects for thrust-offset components $UNFRB_Y$ and $UNFRB_Z$. Since these corrections are small, no unit need be taken in Eq. (15.14). A small window pointing error, shown in Figure 16, is introduced by the thrust-offset correction. Defined as the angle between the ZCB,XCB plane and the unit window command, the window pointing error is the product of the sine of the LPD angle and the thrust-offset angle about the ZCB-axis. Although the trim gimbal has a maximum displacement of 6° , the maximum thrust offset during descent is about 1° , which yields a maximum window pointing error of 0° at 0° LPD angle and 0.9° at 65° LPD angle, the lower edge of the LM window.

Because the matrix CCBP is the transformation from platform to commanded-body coordinates, it can be expressed in terms of the IMU gimbal angles which would place the body axes in the commanded directions. Therefore, commanded gimbal angles can be extracted from the commanded-body matrix. Expressing CCBP as the product of the three matrices that correspond to rotations about the three gimbal axes yields

$$CCBP = \begin{bmatrix} +CZ & CY & & & & & +SZ & & -CZ & SY \\ -CX & SZ & CY + SX & SY & & & +CX & CZ & +CX & SZ & SY + SX & CY \\ +SX & SZ & CY + CX & SY & & & -SX & CZ & -SX & SZ & SY + CX & CY \end{bmatrix}, \quad (13)$$

where S and C indicate sine and cosine, and X, Y, and Z indicate the commanded X, Y, and Z gimbal angles. From Eq. (13), it is apparent that the commanded gimbal angles are extracted from the elements of CCBP by Eqs. (15.16)–(15.18), with ARCTRIG defined as follows.

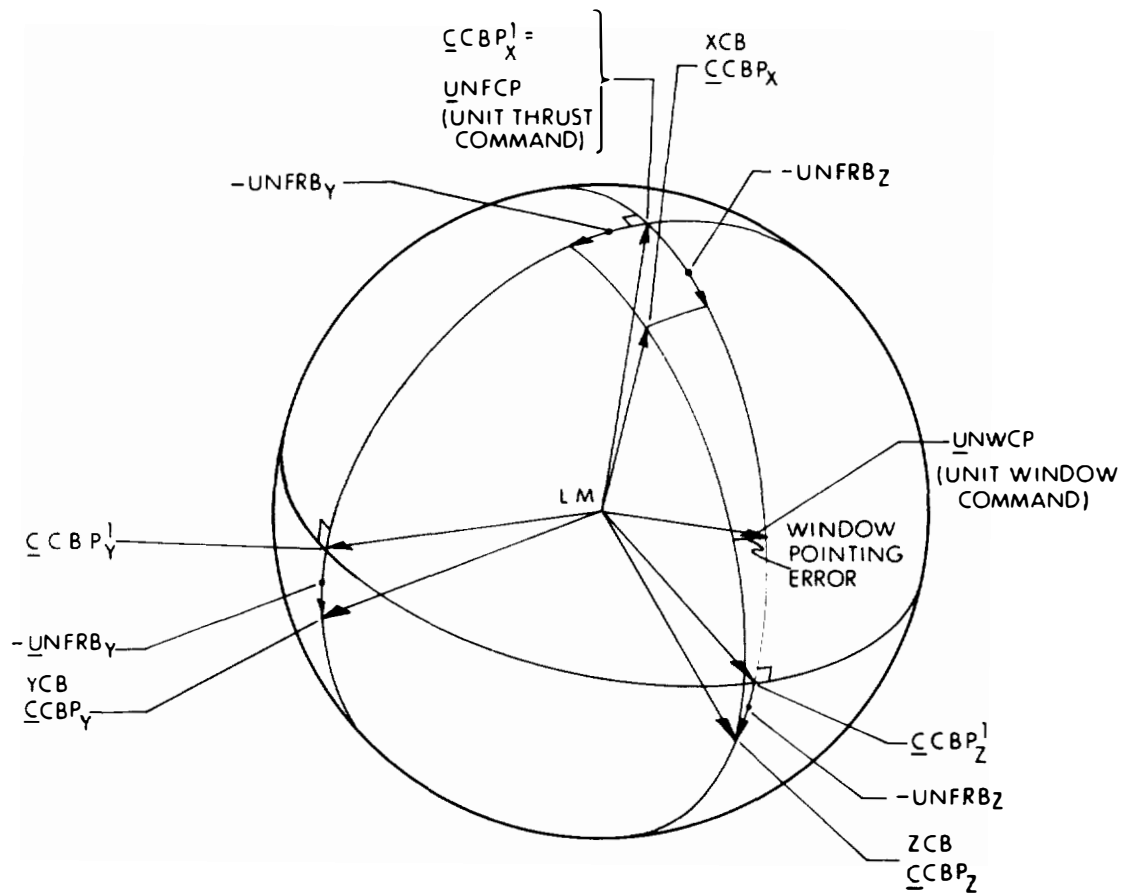


Figure 16 Geometry of Erection of Commanded-body Frame Viewed on a LM-centered Unit Sphere

The ARCTRIG function of two arguments yields the angle whose tangent is the ratio of the first and second arguments. ARCTRIG extracts the angle anywhere in the circle by using the ratio of the smaller-magnitude argument to the larger-magnitude argument as the tangent of the angle or its complement, and by using the signs of the arguments to determine the quadrant of the angle. Equations (15.16) and (15.17) yield the outer and inner commanded gimbal angles anywhere in the circle. Because the second argument is always positive, implying a positive cosine, Eq. (15.18) yields the middle commanded gimbal angle between $\pm 90^\circ$.

To preclude commanding gimbal lock, Eq. (15.20) limits the middle commanded gimbal angle to 70° magnitude. Because the unlimited value lay between $\pm 90^\circ$ and the outer and inner commanded gimbal angles were computed consistent with the middle commanded gimbal angle range, no quadrant switching of the outer or inner commands is required by gimbal-lock limiting. If limiting changes the middle command, the guidance is commanding gimbal lock, and the gimbal-lock alarm code is issued.

Unlimited reference gimbal angle changes are the changes which would be required to bring the DAP's reference gimbal angles into coincidence with the commanded gimbal angles. These are computed by subtracting, modularly, the current reference gimbal angles from the commanded gimbal angles (Eq. (15.21)). The modular subtractions yield the smaller angular differences, i.e.,

$$-170^\circ \bar{M} 170^\circ = +20^\circ, \text{ not } -340^\circ$$

If a Y or Z gimbal angle change greater than 45° is required, the flag is reset indicating no attitude rotation is allowed about the XCB-axis. This is necessary to prevent false starts about the XCB-axis as derived in the appendix of reference 12.

Equations (15.24)–(15.28) yield the reference gimbal angle changes by limiting the magnitude of the attitude changes to 20° in 2 seconds ($10^\circ/\text{sec}$) about each of three orthogonal axes; one axis is coincident with the XCB-axis and the other axes lie in the YCB,ZCB plane. This permits an angular-rate vector of length $10\sqrt{3}$ deg/sec. Note that if the flag is reset, the attitude rotation about the XCB-axis is made zero, resulting in an outer gimbal angle change to offset the inner gimbal angle change (Eq. (15.28)).

The DAP commands consist of the reference gimbal angle increments to be applied by the DAP each 1/10 second, the reference attitude rates, and the permitted

lag angles. The reference gimbal angle increments are the reference gimbal angle changes multiplied by the ratio of the DAP and ATT sample intervals (Eq. (15.29)). The reference attitude rates are computed by the nonorthogonal transformation of the reference gimbal angle changes shown in Eq. (15.30). The permitted lag angles, which account for the angles by which the attitude will lag behind a ramp angular command due to the finite accelerations available, are computed using the available acceleration $\underline{a}_{RB}$, and then individually magnitude limited (Eq. (15.31)). The DAP avoids attitude-rate overshoot by permitting lagging attitude errors equal to the permitted lag angles.

THROTTLE ROUTINE

The throttle routine connects the currently operating guidance algorithm to the DPS, as illustrated in Figure 17. For ease of understanding, all thrust levels are represented as percentages of the DPS rated thrust of 46 706 newtons. THROT generates thrust increment commands to drive the input thrust-acceleration measurement into coincidence with the input thrust-acceleration command whenever the resulting thrust would lie within the illustrated permitted-thrust region. When the resulting thrust would lie below or above the permitted-thrust region, THROT causes minimum or maximum thrust. The hysteresis-like region from 57 to 65% thrust avoids frequent alternation between the maximum-thrust point and the permitted-thrust region when the thrust command dwells at the boundary between the permitted-thrust and forbidden-thrust regions.

A digital-to-analog interface between the LGC and the DPS is provided by the descent engine control assembly (DECA). Each guidance cycle (once per two seconds, except once per second for P66) THROT generates the thrust increment command $\Delta FC\%$ which is converted to a pulse train and issued to the DECA. Each pulse causes about 12.5 newtons thrust change, and the pulse rate is 3200/second. Following issuance of a thrust increment command, the thrust therefore changes at the rate of 40,000 newtons/second (85% of rated thrust per second) until the thrust increment is achieved. With a guidance cycle as short as one second and an engine response time which may be a substantial fraction of one second, it is necessary for P66 and THROT to account for this transport delay.

As illustrated in the rightmost box of Figure 17, in the region from 11 to 93% the DPS thrust is a nearly linear function of the pulse count accumulated by the DECA. Not shown in Figure 17 is the manual throttle command, which is summed with the DECA output command by the DPS and provides the minimum 11% thrust when the DECA command is zero. The DPS contains a mechanical stop at typically 93% rated thrust. This thrust level minimizes propellant consumption on a nominal descent, considering the loss of specific impulse at higher thrust. To ensure that the DPS is driven to the mechanical stop, the DECA saturates at a substantially higher thrust level (about 99%) and the throttle routine drives the DECA into saturation whenever maximum thrust is required.

Nonlinearities in response and uncertainties in DECA and DPS scale factors are overcome by the thrust increment command concept. Nominally, THROT provides dead-beat response to step inputs, but with downstream nonlinearities and scale-factor errors THROT drives the thrust-acceleration error to zero in the steady state.

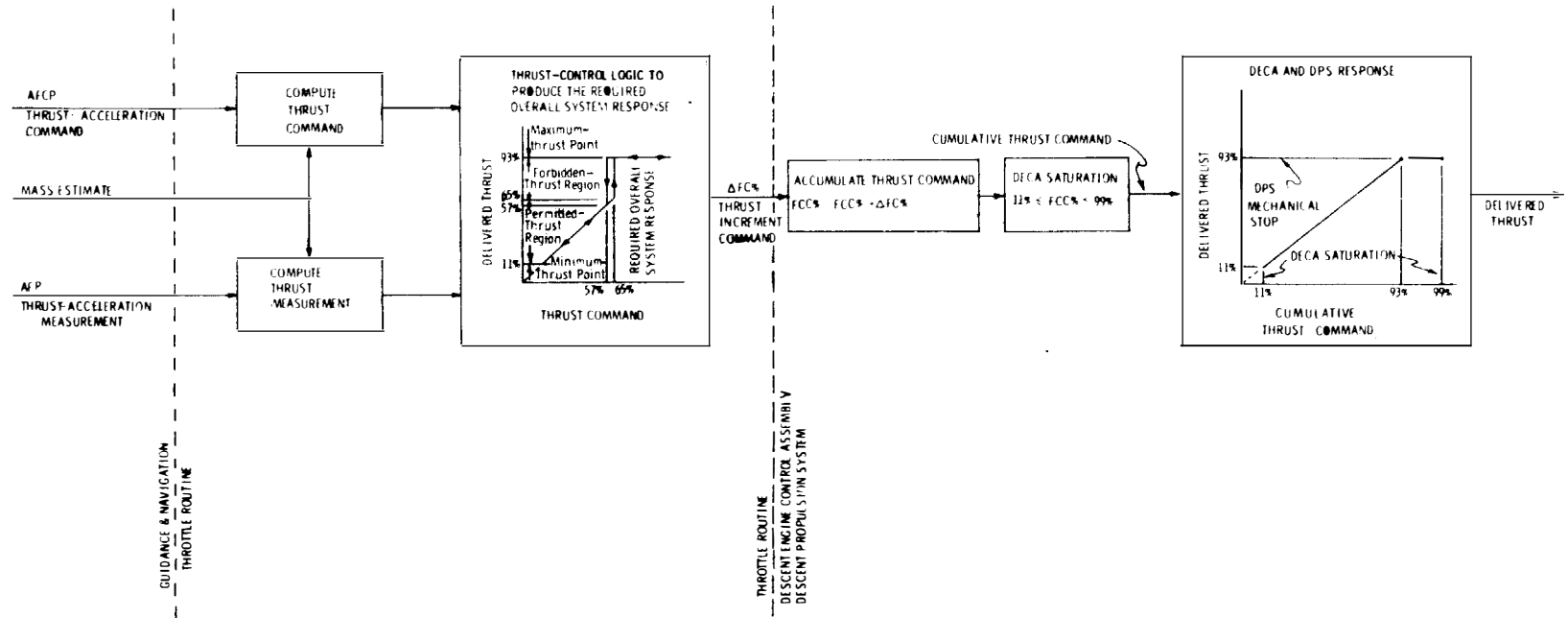


Figure 17 DPS Thrust Control

Figure 18 illustrates the Throttle Routine computations. The thrust command and thrust measurement are computed using the input mass estimate (Eqs. (18.1)–(18.2)). The input thrust-acceleration measurement is the average over the preceding sample interval, during which a thrust increment command was issued producing an instantaneous thrust profile as illustrated in Figure 19. Therefore, to obtain the current sample-instant thrust, Eq. (18.3) corrects the thrust measurement by adding the thrust correction increment computed the previous cycle.

The thrust-control logic for providing the required overall system response illustrated in Figure 17 is to pick one of four possible thrusting policies according to the regions of the preceding and present thrust commands (FCOLD% and FC%), and to reset the thrust command if necessary to satisfy DPS constraints. Equation (18.4) or (18.8) resets the thrust command to the thrust actually anticipated. A thrust command augment (FCAUG%) is computed that either drives the DECA into saturation if the policy is to initiate or retain maximum thrust (Eq. (18.5) or (18.9)), or corrects for the region between the DPS mechanical stop and the DECA saturation value if the policy is to initiate thrusting within the permitted-thrust region (Eq. (18.7)). No thrust command augment is required when the policy is to continue thrusting within the permitted-thrust region. No equivalent thrust-control logic is needed at the minimum-thrust point because minimum thrust would occur only if the commander could issue five or more downward ROD commands within a single P66 guidance sample interval, practically impossible.

The thrust increment command (Eq. (18.12)) is composed of the actual thrust increment $\Delta FA\%$, plus the thrust command augment FCAUG% to drive the DECA in or out of saturation, when required.

Preparatory to computing the thrust correction increment for the succeeding pass, Eq. (18.13) computes the total effective transport lag. The terms in the effective transport lag are 1) the computation duration $t - t_{SI}$, 2) the estimated DPS time constant of 0.08 second, and 3) the effective DECA delay equal to half the time required to output the thrust increment command pulse train at 85% thrust change per second. As long as the actual thrust increment ΔFA (Figure 19) is contained entirely within the sample interval Δt , it is clear that, as LAG approaches zero, the thrust measurement (obtained by differencing accelerometer readings at the sample instants) approaches the sample-instant thrust F . Similarly, as LAG approaches the sample interval Δt , the thrust measurement must be augmented by an amount approaching the actual thrust increment ΔFA to obtain the sample-instant thrust F . From this heuristic argument, it is apparent that the thrust correction increment which must be added to the thrust measurement to yield the sample-instant

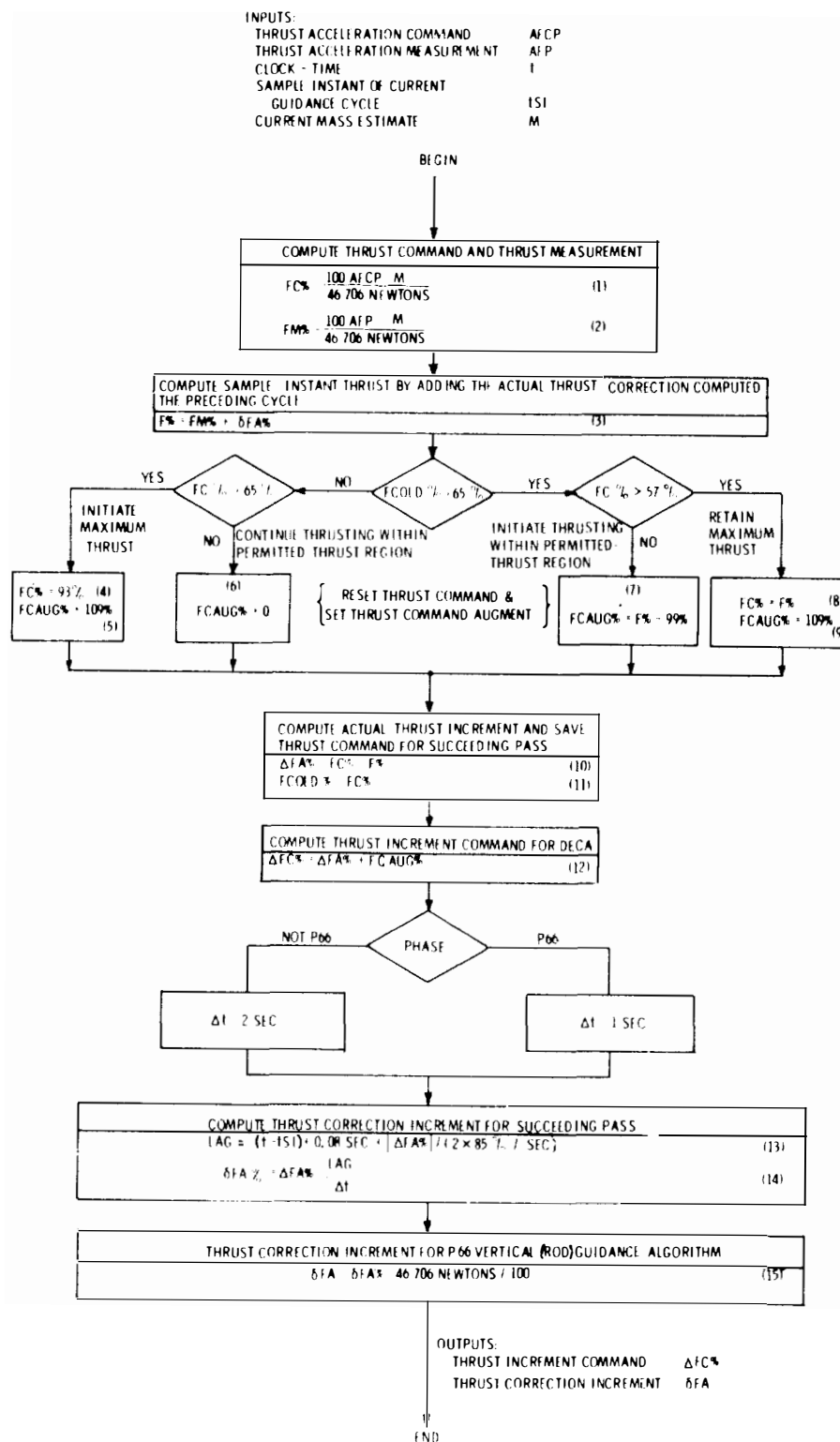


Figure 18 Throttle Routine

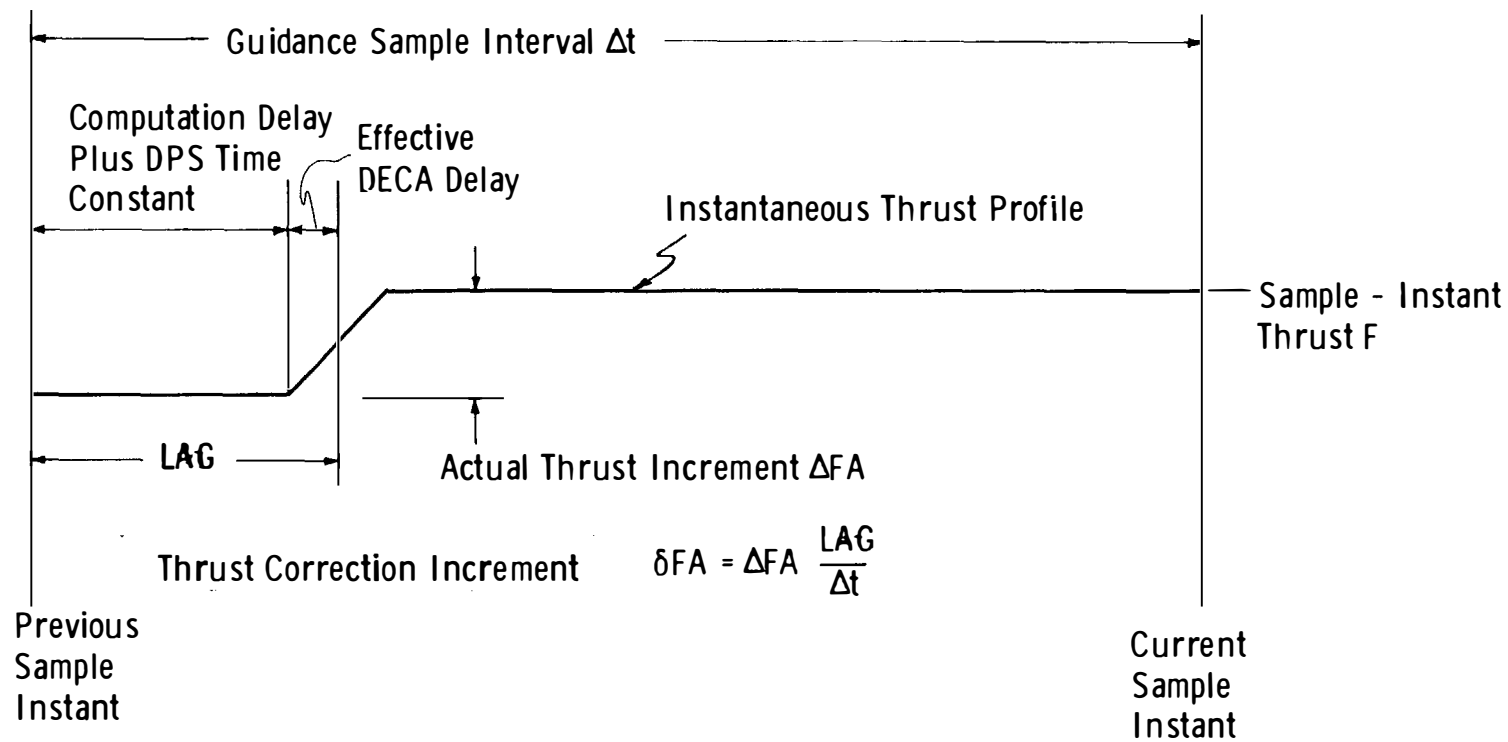


Figure 19 Thrust Dynamics Within a Single Guidance Sample Interval

thrust is proportional to LAG as computed by Eq. (18.14). A rigorous derivation of this result is presented in Appendix A of reference 11. The sole purpose of Eq. (18.15) is to interface the P66 Vertical (ROD) Guidance Algorithm.

With the thrust command $FC\%$ either within the permitted-thrust region or reset to the value which will actually be achieved, $\Delta FA\%$ is an accurate prediction of the actual thrust increment, and $\delta FA\%$ or δFA is an accurate thrust correction increment. $\delta FA\%$ or δFA is slightly in error when initiating thrusting within the permitted-thrust region. The slight error is due to neglecting $FCAUG\%$ in the computation of LAG.

BRAKING-PHASE AND APPROACH-PHASE TARGETING PROGRAM

The targeting program generates mission-dependent data for the P63 Ignition Algorithm and for the P63,P64 Guidance Algorithm. All data are expressed in guidance coordinates. The ignition algorithm requires nominal initial altitude, range, and speed data that determine ignition time and indirectly determine the throttle-control duration. The guidance algorithm requires targets for P63 that provide an efficient transfer and targets for P64 that provide a trajectory meeting several constraints on geometry, visibility, and thrust. Described in detail in reference 5, the P64 constraints provide a fast shallow approach phase more akin to an airplane approach than a helicopter approach, although the terminal-descent phase is essentially vertical in helicopter fashion. The landing site must be approached along a nearly straight-line path depressed typically 16° from horizontal, terminating typically at 30 m altitude 11 m ground range. The landing site must be visible continuously until a few seconds before approach-phase terminus, and the DPS thrust must begin at around 57% and must lie continuously in the 11 to 65% region.

Geometry, visibility, and thrust during approach cannot be specified explicitly. Visibility depends upon the position and attitude profiles, and these profiles (with the thrust and mass profiles) are constrained to satisfy the laws of physics. The guidance algorithm will provide the transfer from any arbitrary initial state (within bounds) without regard to any visibility or thrust constraints. The task of the targeting program is to set up the P64 initial state and guidance targets such that suitable visibility and thrust profiles are realized implicitly.

During the final portion of P63, and throughout P64, the guidance algorithm will generate a trajectory whose position vector is a quartic polynomial function of time, as shown in Figure 20. Targeting consists of 1) defining each of the two polynomials and 2) extracting the guidance targets as the position vector and its derivatives at a target point, lying on the polynomial, substantially beyond phase terminus.

The P64 targeting concept is to construct the approach-phase quartic by imposing necessary and sufficient constraints. With quartic degree, five independent constraints may be imposed in each of three axes. The nominal trajectory is arbitrarily made planar, requiring the Y-components in guidance coordinates of position and all its derivatives to be zero and leaving two axes to specify. Because the initial state can be controlled by the preceding braking-phase guidance, all five constraints in each of the two remaining axes may be specified arbitrarily. Since these ten constraints — called a P64 constraint set — completely determine the

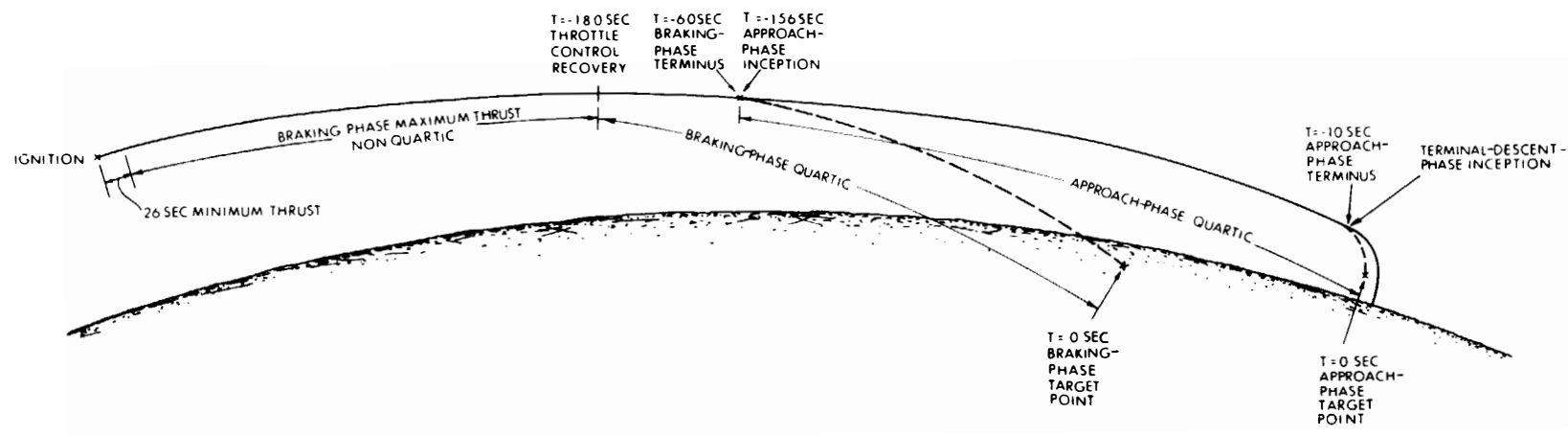


Figure 20 Composition of the Lunar-descent Trajectory

P64 trajectory, the geometry and visibility profiles can be determined in closed form, and the thrust profile can be determined from a prior knowledge of mass. Thus P64 targeting consists of generating closed-form solutions for a number of P64 constraint sets and picking one which provides adequate visibility and thrust. Specification of P64 constraint sets is reduced to a two-dimensional search, as will be described in the following section.

The P63 targeting process is not so clean. Because the engine must be run at fixed maximum thrust for most of the phase, the guidance commands are not satisfied, and therefore, as shown in Figure 20, the maximum-thrust portion of P63 is not quartic. When throttle control is recovered, generation of a quartic is begun. But the throttle recovery point is not close to any target point. Therefore the state vector at this point cannot be controlled, and we have no closed-form solution for it. Since the initial position and velocity on the braking-phase quartic must be free, there remain only three constraints, in each of two axes, which can be imposed arbitrarily. The guidance algorithm permits a fourth constraint in one axis by solving for the current target-referenced time such as to satisfy a constraint on the ZG-component of jerk. Thus a P63 constraint set composed of seven constraints is specified arbitrarily, and the remaining three conditions required to define the braking-phase quartic are determined iteratively by simulation. Three or four iterations are generally required because there is bilateral interaction between the targets and the simulation.

Constraints

The P64 constraint set is constructed as follows:

1. Four constraints at a specified target-referenced terminal time TAPF: Two terminal vertical constraints, specified by the mission commander, are the terminal altitude ($RAPFG_X = 30\text{-m}$ typically) and altitude rate ($VAPFG_X = -1\text{-m/sec}$ typically). Two terminal horizontal constraints, imposed by the choice of effective P66 horizontal time constant τ , are that the terminal position, velocity, and acceleration shall be related by $RAPFG_Z = AAPFG_Z \tau^2$, $VAPFG_Z = -AAPFG_Z \tau$. These P66 compatibility constraints cause the pitch commands at P64 terminus and P66 inception to be identical (avoiding a pitch transient at the phasic interface) and cause the P66 algorithm to null the horizontal position error as well as the horizontal velocity error, without position feedback. Because the P66 horizontal algorithm feeds back the prior acceleration command, and because of the transport delay, an effective τ of 8 seconds has been found satisfactory rather than the 5 seconds used by the P66 algorithm.

2. Four constraints at an unspecified target-referenced midpoint time TAPM: The midpoint constraints are specified by the commander according to his sense of safety and compatibility with a possible manual transition to P66. Typically, he may specify -5-m/sec altitude rate at 150-m altitude, and a 16° slope, completely determining the midpoint state $\underline{R}APMG, \underline{V}APMG$ given $RAPMG_Y = VAPMG_Y = 0$.
3. Two constraints at an unspecified target-referenced initial time TAPI: The initial position is arbitrarily specified to lie on the 16° path and to provide an approach phase of typically 7.5-km length, determining the initial position vector $\underline{R}APIG$ given $RAPIG_Y = 0$.

This completes the P64 constraint set except for specifying the times TAPM and TAPI at which the midpoint and initial constraints apply. These times are determined by running the Approach-phase Targeting Routine over the two-dimensional sweep of values of TAPM and TAPI. From the cases run, one is picked that exhibits suitable attitude and thrust behavior (based on an a-priori P64 initial mass estimate). If subsequent simulation proves the mass estimate excessively in error, the initial thrust will be unsatisfactory, and an alternate case must be picked.

The seven P63 constraints are specified as follows: Four constraints are specified by compatibility of the terminal state on the braking-phase quartic with the initial state on the approach-phase quartic. Two constraints are imposed on terminal acceleration by requiring the terminal thrust to be 57% and by specifying the terminal pitch angle. The final constraint is imposed on the horizontal component of terminal jerk by requiring zero rate of change of thrust at terminus. The terminal pitch angle, typically around 60° , is chosen by trial and error to minimize propellant consumption.

Approach-phase Targeting

Figure 21 illustrates the Approach-phase Targeting Routine. Normally, this routine is first run separately in search of targets for the approach phase, and then run jointly with the Braking-phase Targeting Routine (Figure 22) to determine targets for the entire lunar descent.

In the XG-axis (altitude), the terminal acceleration, jerk, and snap are computed by Eq. (21.2), which is obtained immediately from

$$\begin{bmatrix} \text{RAPMG}_X \\ \text{VAPMG}_X \\ \text{RAPIG}_X \end{bmatrix} = \begin{bmatrix} 1 & \text{TMF} & \text{TMF}^2/2 & \text{TMF}^3/6 & \text{TMF}^4/24 \\ 0 & 1 & \text{TMF} & \text{TMF}^2/2 & \text{TMF}^3/6 \\ 1 & \text{TIF} & \text{TIF}^2/2 & \text{TIF}^3/6 & \text{TIF}^4/24 \end{bmatrix} \begin{bmatrix} \text{RAPFG}_X \\ \text{VAPFG}_X \\ \text{AAPFG}_X \\ \text{JAPFG}_X \\ \text{SAPFG}_X \end{bmatrix} \quad (14)$$

where TMF and TIF are the midpoint and initial terminus-referenced times computed by Eqs. (21.1).

In the YG-axis, position and all its derivatives are zero to produce a planar trajectory.

In the ZG-axis, Eq. (21.4) is obtained by substituting the P66 compatibility constraints $\text{RAPFG}_Z = \text{AAPFG}_Z \tau^2$, $\text{VAPFG}_Z = -\text{AAPFG}_Z \tau$ into the ZG-axis version of Eq. (14) and inverting. Equations (21.5) and (21.6) complete the definition of the approach-phase quartic. It remains to compute the approach-phase targets as the position vector and its derivatives at the target point on the quartic.

For a quartic polynomial, a 5×5 state transition matrix $\Phi(T_1, T_0)$ can be defined by

$$X_1 = \begin{bmatrix} \underline{R}_1 \\ \underline{V}_1 \\ \underline{A}_1 \\ \underline{J}_1 \\ \underline{S}_1 \end{bmatrix} = \begin{bmatrix} 1 & (T_1-T_0) & (T_1-T_0)^2/2 & (T_1-T_0)^3/6 & (T_1-T_0)^4/24 \\ 0 & 1 & (T_1-T_0) & (T_1-T_0)^2/2 & (T_1-T_0)^3/6 \\ 0 & 0 & 1 & (T_1-T_0) & (T_1-T_0)^2/2 \\ 0 & 0 & 0 & 1 & (T_1-T_0) \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \underline{R}_0 \\ \underline{V}_0 \\ \underline{A}_0 \\ \underline{J}_0 \\ \underline{S}_0 \end{bmatrix} = \Phi(T_1, T_0) X_0,$$

where $\underline{R}_i$ to $\underline{S}_i$ are row vectors. $\Phi(T_1, T_0)$ can be derived using linear systems theory,

$$\dot{X} = \begin{bmatrix} \dot{\underline{R}} \\ \dot{\underline{V}} \\ \dot{\underline{A}} \\ \dot{\underline{J}} \\ \dot{\underline{S}} \end{bmatrix} = \begin{bmatrix} \underline{V} \\ \underline{A} \\ \underline{J} \\ \underline{S} \\ \underline{0} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \underline{R} \\ \underline{V} \\ \underline{A} \\ \underline{J} \\ \underline{S} \end{bmatrix} = \alpha X$$

with the solution

$$\Phi = e^{\alpha(T_1 - T_0)} = I + \alpha(T_1 - T_0) + \alpha^2 (T_1 - T_0)^2 / 2 + \alpha^3 (T_1 - T_0)^3 / 6 + \alpha^4 (T_1 - T_0)^4 / 24,$$

where I is the 5 x 5 identity matrix. The exponential series is zero after the fifth term because $\alpha^i = 0$ for $i > 5$. All the properties of state transition matrices can be applied to scalar and vector polynomials.

Equations (21.7) yield the complete target and initial states by using state transition matrices and the definition of target-referenced target time as zero.

Braking-phase Targeting

To target P63, we must completely determine the braking-phase quartic shown in Figure 20. Seven of the ten necessary conditions are determined in closed form, although three are based on a P63 terminal mass estimate which must be updated by simulation. The remaining three conditions necessary to define the quartic are determined iteratively by simulation. The terminal pitch angle θ_{PBRF} is a fixed input to the Braking-phase Targeting Routine.

Figure 22 illustrates the routine. Four conditions are specified by setting the P63 terminal position and velocity equal to the P63 initial state* (Eqs. (22.1)). A unit vector in the terminal-thrust direction is computed from the terminal pitch angle θ_{PBRF} (Eq. (22.2)), and the terminal acceleration is calculated by Eq. (22.3) using the terminal thrust FBRF, the P63 terminal mass estimate MBRF, and allowing for lunar gravity GM. The XG-component of terminal jerk must be determined by simulation and is therefore set to zero for the first iteration (Eq. (22.5)). The ZG-component of terminal jerk is computed by Eq. (22.5) to produce zero rate of change of thrust at terminus, accounting for the estimated terminal mass flow rate computed by Eq. (22.4); the jerk coefficient KJ, typically 1.2, accounts for the XG-component of thrust. The terminal snap must be determined by simulation and is therefore set to zero (Eq. (22.6)) for the first iteration. This completes the first-iteration definition of the braking-phase quartic.

* Not shown is the capability of the targeting program to set the P63 terminal state to a backwards extrapolation of the P64 initial state to allow for a short transition during which the acceleration is assumed to change linearly with time. This capability is not always used, and to show it would unnecessarily complicate the presentation of Figure 22.

Braking-phase targets are computed by Eq. (22.7), using the state transition matrix and the definition of target-referenced target time as zero. Using the computed targets, a simulation is run to produce corrected data.

The nominal initial range used by the ignition algorithm is corrected by Eq. (22.8) to correct the error in the target-referenced time of throttle control recovery.

The simulation produces a braking-phase quartic satisfying the target values of position, velocity, acceleration, and ZG-component of jerk. The remaining conditions necessary to define the quartic can be obtained from the current state on the last pass of the braking-phase simulation. The equation for the current state,

$$\begin{bmatrix} \underline{R}_G \\ \underline{V}_G \end{bmatrix} = \begin{bmatrix} 1 & T & T^2/2 & T^3/6 & T^4/24 \\ 0 & 1 & T & T^2/2 & T^3/6 \end{bmatrix} \begin{bmatrix} \underline{R}_{BRTG} \\ \underline{V}_{BRTG} \\ \underline{A}_{BRTG} \\ \underline{J}_{BRTG} \\ \underline{S}_{BRTG} \end{bmatrix},$$

is readily solved to yield the achieved target jerk and snap according to Eq. (22.9). Solving for the ZG-component of achieved target jerk provides a check on the computation of T by the guidance algorithm; agreement between achieved and input values is typically to seven places.

In preparation for correcting estimates at the terminus, the complete state at terminus is computed by Eq. (22.10). Equation (22.10) yields a terminal state at the specified terminal time TBRF precisely, whereas the state $\underline{R}_G, \underline{V}_G$ applies at the time T which may differ from TBRF by up to the 2-second granularity.

Equation (22.11) corrects the P63 terminal mass estimate using the rocket equation. Equations (22.12) – (22.14) correct the terminal acceleration, jerk, and snap using the corrected P63 terminal mass estimate, the achieved XG-component of terminal jerk, and the achieved XG- and ZG-components of terminal snap.

Finally, state convergence test quantities are computed by Eqs. (22.15) – (22.17). Since only three conditions ($\underline{JBRFGA}_X$, $\underline{SBRFGA}_X$, and $\underline{SBRFGA}_Z$) defining the braking-phase quartic are sought iteratively, only three convergence criteria

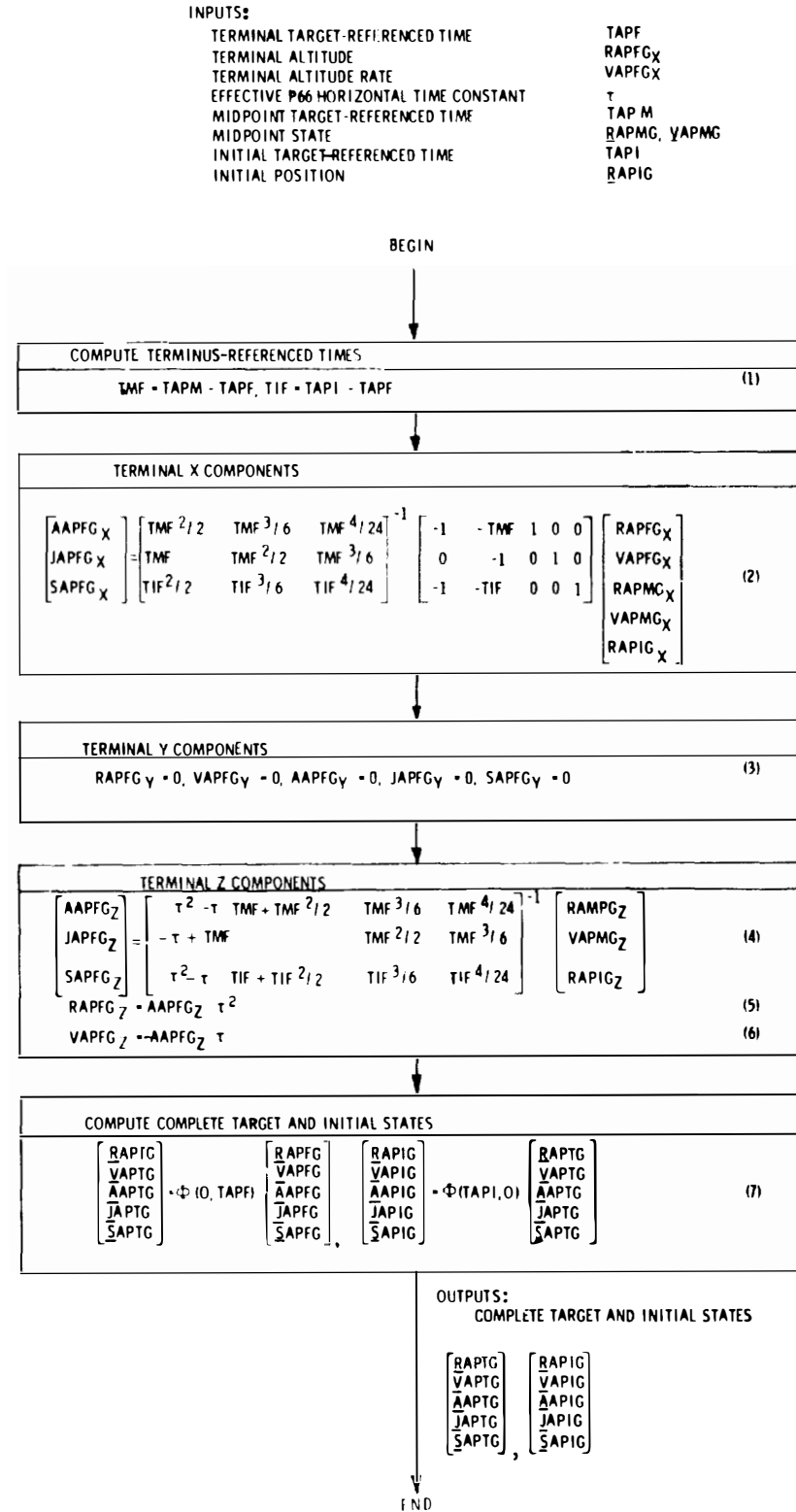


Figure 21 Approach-phase Targeting Routine

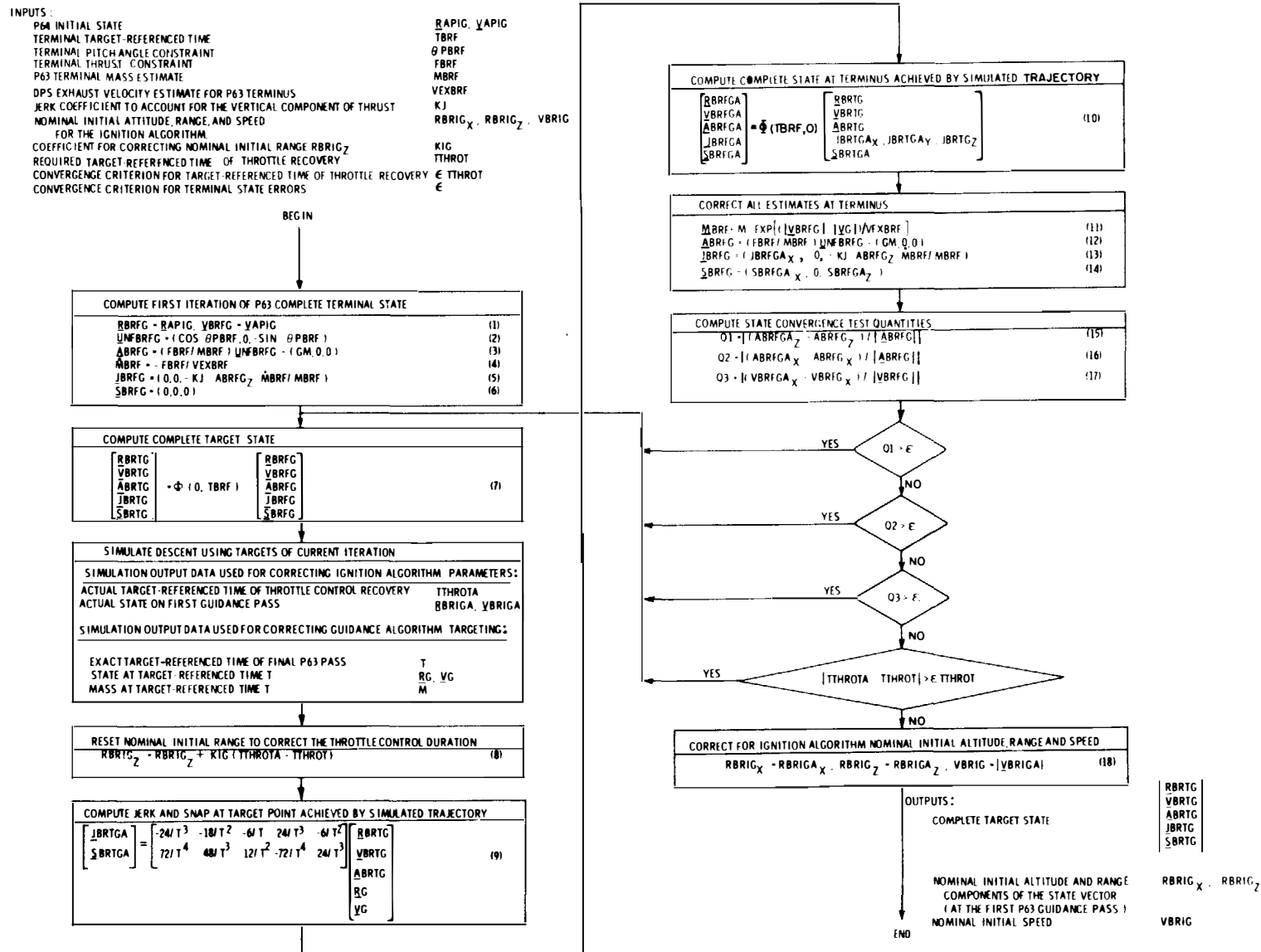


Figure 22 Braking-phase Targeting Routine

are needed. The three criteria chosen are important for guidance performance and are related nonsingularly to three conditions sought. If any one of the state convergence tests fails, or if the throttle control recovery time convergence test fails, the braking-phase targets are corrected and another simulation is run; otherwise the targeting is concluded by correcting the ignition algorithm inputs per Eq. (22.18).

REFERENCES

1. Kriegsman, B.A., "Radar-Updated Inertial Navigation of a Continuously-Powered Space Vehicle", IEEE Aerospace Systems Conference, Seattle, Washington, July 11-15, 1966.
2. Widnall, W.S., "Lunar Module Digital Autopilot", Journal of Spacecraft and Rockets, Vol. 8, No. 1, January 1971.
3. Cherry, G.W., "E Guidance — A General Explicit, Optimizing Guidance Law for Rocket-Propelled Spacecraft", MIT Instrumentation Laboratory Report R-456, August 1964.
4. Klumpp, A.R., "A Manually Retargeted Automatic Descent and Landing System for LEM", MIT Instrumentation Laboratory Report R-539, March 1966.
5. Klumpp, A.R., "A Manually Retargeted Automatic Landing System for the Lunar Module (LM)", Journal of Spacecraft and Rockets, February 1968.
6. Moore, T.E., G.G. McSwain, and J.D. Montgomery, "Guidance Laws for Controlling Off-Nominal LM Powered Descent Trajectories Back to the Nominal", Internal Note MSC-EG-69-9, Project Apollo, NASA, Manned Spacecraft Center, Houston, Texas, February 28, 1969.
7. McSwain, G.G. and T.E. Moore, "False High Gate Targeting for LM Powered Descent", Internal Note MSC-EG-68-07, NASA, Manned Spacecraft Center, Houston, Texas, May 27, 1968.
8. Yang, T.L., "A Targeting Scheme for Fuel Optimal Rocket Trajectories — With Applications to the LM Descent Braking Phase", Technical Memorandum TM-71-2014-1, Bellcomm January 22, 1971.
9. MIT Charles Stark Draper Laboratory Report R-567, "Guidance System Operations Plan for Manned LM Earth Orbital and Lunar Missions Using Program Luminary 1D", Section 5, Guidance Equations, Rev. 9, December 1970.

10. Eyles, D.E., "Apollo LM Guidance and Pilot-Assistance During the Final Stage of Lunar Descent — Software Considerations", Fourth IFAC Symposium on Automatic Control in Space, Dubrovnik, Yugoslavia, September 6-10, 1971.
11. Klumpp, A.R. and G.R. Kalan, "Elimination of Noise and Enhancement of Stability and Dynamic Response of the Apollo LM Rate-of-Descent Program", MIT Charles Stark Draper Laboratory Report E-2543, October 1970.
12. Klumpp, A.R., "FINDCDUW — Guidance Autopilot Interface Routine", MIT Instrumentation Laboratory LUMINARY Memo No. 27, Rev. 1, September 26, 1968.
13. Bennett, F.V., "Lunar Descent and Ascent Trajectories", AIAA Eighth Aerospace Sciences Meeting, New York, January 19-21, 1970.
14. Bennett, F.V., "Mission Planning for Apollo Lunar Module Descent and Ascent", to be published as a NASA Technical Note.

R-695

DISTRIBUTION LIST

Internal:

P. Adler	G. Kalan
S. Albert	D. Keene
R. Battin	J. Kernan
L. Berman	A. Klumpp (150)
H. Blair-Smith	R. Larson
F. Brunswick	J. Laning
S. Copps	B. McCoy
L. Drane	R. Metzinger
T. Edelbaum	D. Millard
A. Engel	P. Mimno
D. Eyles	D. Moore
D. Farrar	J. Nevins
P. Felleman	N. Pippenger
S. Femino	J. Reed
T. Fitzgibbon	P. Rye
D. Fraser	J. Scanlon
R. Gilbert MIT/KSC	R. Schlundt
K. Glick	C. Schulenberg
E. C. Hall	N. Sears
M. Hamilton	R. Stengel
P. Heinemann	P. Volante
D. Hoag	R. Weatherbee
A. Hopkins	P. Weissman
I. Johnson	CSDL/TDC (10)
M. Johnston	Apollo Library (2)
	Group 23A (18)

External:

NASA/RASPO	(1)
DELCO	(3)
Kollsman	(2)
Raytheon	(2)
MSC:	(37&1R)

National Aeronautics and Space Administration
Manned Spacecraft Center
Houston, Texas 77058

ATTN: Apollo Document Control Group (BM 86)	(18&1R)
M. Holley (EG 14)	(2)
T. Gibson (FS 5)	(1)
B. Kirkland (FM 2)	(1)
B. Taylor (FM 2)	(1)
J. Alphin (FM 2)	(1)
W. Bolt (FM 2)	(1)
D. Jezewski (FM 8)	(1)
F. Bennett (FM 2)	(1)
J. Garman (FS 5)	(1)
D. Scott (CB)	(1)
A. Worden (CB)	(1)
P. Conrad (CB)	(1)
J. Young (CB)	(1)
K. Mattingly (CB)	(1)
F. Haise (CB)	(1)
E. Mitchell (CB)	(1)
R. Gordon (CB)	(1)
V. Brand (CB)	(1)
D. Cheatham (EG)	(1)
R. Chilton (EG)	(1)

KSC:	(1&1R)
------	--------

National Aeronautics and Space Administration
J. F. Kennedy Space Center
J. F. Kennedy Space Center, Florida 32899
ATTN: Technical Document Control Office
R. Pearson (CFK)

(1)

LRC:	(2)
------	-----

National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia
ATTN: Mr. A. T. Mattson

GA:	(3&1R)
-----	--------

Grumman Aerospace Corporation
Data Operations and Services, Plant 25
Bethpage, Long Island, New York
ATTN: Mr. E. Stern

NAR: (4&1R)

North American Rockwell, Inc.
Space Division
12214 Lakewood Boulevard
Downey, California 90241
ATTN: T. R. Watson; CSM Data Management
D/096-402 AE 99

NAR RASPO: (1)

NASA Resident Apollo Spacecraft Program Office
North American Rockwell, Inc.
Space Division
12214 Lakewood Boulevard
Downey, California 90241

GE: (1)

General Electric Company
Apollo Systems
P.O. Box 2500
Daytona Beach, Florida 32015
ATTN: E.P. Padgett, Jr./Unit 509

HDQ: (1)

National Aeronautics and Space Administration
Washington DC 20546
ATTN: G. Cherry

JPL: (2)

Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91103
ATTN: R. Morris
W. Breckenridge

