MEMORANDUM
RM-6032-PR
JUNE 1969

GRAPHIC ROCKET:
SCENARIO OF A FILMED REPORT

B. W. Boehm, V. R. Lamb, R. L. Mobley and J. E. Rieber

PREPARED FOR:
UNITED STATES AIR FORCE PROJECT RAND

The RAND Corporation
SANTA MONICA • CALIFORNIA

THIS DOCUMENT HAS BEEN APPROVED FOR PUBLIC RELEASE AND SALE; ITS DISTRIBUTION IS UNLIMITED.
MEMORANDUM
RM-6032-PR
JUNE 1969

GRAPHIC ROCKET:
SCENARIO OF A FILMED REPORT
B. W. Boehm, V. R. Lamb, R. L. Mobley and J. E. Rieber

This research is supported by the United States Air Force under Project RAND—Con-
tract No. F44620-67-C-0045—monitored by the Directorate of Operational Requirements
and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF.
Views or conclusions contained in this study should not be interpreted as representing
the official opinion or policy of the United States Air Force.

DISTRIBUTION STATEMENT
This document has been approved for public release and sale; its distribution is unlimited.
PREFACE and SUMMARY

This Memorandum illustrates the use of an interactive computer-graphics system in the analysis of aerospace vehicle designs. The system allows the user to rapidly specify or modify a design and flight plan for an aerospace vehicle, and to view curve displays of the resulting vehicle performance, all directly at the computer-graphics console.

The Memorandum is directed primarily toward interactive graphic system developers and aerospace engineers. For the latter, an explanation of special computer-graphics terminology is provided in the Appendix.
CONTENTS

PREFACE and SUMMARY .................................. 111

Section
   I. INTRODUCTION ................................... 1
   II. TEXT OF THE FILM ............................... 5

Appendix
   GLOSSARY OF INTERACTIVE GRAPHICS TERMINOLOGY ... 47

REFERENCES ............................................ 49
I. INTRODUCTION

Graphic ROCKET is an operational, interactive, computer-graphics system currently being used as a performance analysis aid for the preliminary design and evaluation of aerospace vehicle systems. It provides a means for rapidly specifying and evaluating the performance of a wide range of aerospace vehicle designs and flight plans.

The hardware available to the system includes an IBM 360/40 with a 5-microsecond cycle time and 65,536 32-bit words of core storage, with four IBM 2311 disk units for peripheral storage. The interactive graphics terminal includes an IBM 2250 cathode-ray-tube display with light pen, keyboard, and function keys. Hardcopy is provided by a S-C 4060 graphic output device.

Program organization is based on the ROCKET program [1], a general-purpose, trajectory-analysis program developed in 1961 and maintained and extended since in a batch-processing mode. Much of ROCKET's organization was modified to make it operate more naturally and efficiently in an interactive environment. For example, ROCKET allowed the user one pass at the FORTRAN compiler during each run. This provided much flexibility but took too much time for an on-line user; instead, Graphic ROCKET provides lists of options specifiable via the light pen.

ROCKET also provided for branching, allowing a user to specify up to 36 trajectory variations in his initial run statement. Graphic ROCKET does not provide this feature because a user operating interactively can afford to wait until he has seen the results of his base-case run before he specifies his variations, which he can then do more intelligently.

The language consists of a series of "pages" shown on the cathode-ray-tube display for specifying initial conditions, environmental models, and flight plans. Control
boxes on each page allow him to skip back and forth from page to page with the light pen as he defines the problem he wants to solve. A series of "Help" pages provide first-order explanations of definitions and procedures. The pages were designed at the 2250 console and interfaced to the FORTRAN computational routines in a smooth, simple manner via the POGO system [2].

Two display control pages allow the user to specify variables for graphic display and to change displays during the run.

The system also includes a simple file-management scheme that allows the analyst to store and recall inputs and outputs from his previous runs, or store and recall such reference data as tabular representations of atmospheric, propulsion, and aerodynamic properties.

These capabilities have been illustrated in a 15-minute sound film showing the use of Graphic ROCKET in analyzing an air-launched satellite booster for hurricane photography.

The remainder of this Memorandum consists of the text of the film, illustrated by S-C 4060 hardcopy versions of the graphic displays.
II. TEXT OF THE FILM

You are watching a researcher design and analyze an air-launched satellite vehicle with the aid of a computer. Using computers to design aerospace vehicles is not new; what may be new to you is the way the researcher is using the computer and communicating with it.

Here, the user is specifying directly his design and flight plan with the light pen and the on-line keyboard and then watching graphs of the resulting vehicle performance on the TV-like display screen as the vehicle's trajectory is being computed.

Traditionally, computer assistance has been a much longer and more tedious process for the aerospace engineer involving: 1) the manual transcription of design curves and parameters onto keypunch forms, 2) the two-hour to two-day delays for keypunching and for each computer run, and 3) the manual transcription of numerical computer printouts into performance curves. Most irritating to the engineer are the inevitable trivial errors that are recognized and corrected within seconds of seeing the output, but still consign him to another two-hour to two-day wait for his corrected results.
With the system you have been watching, called Graphic ROCKET, the engineer can accomplish all his debugging and performance analyses in an interactive dialogue with the computer. At any time during the computation, if he does not like the results, he can go back, respecify part of the design and flight plan, and try again immediately.

When the engineer is through for the day, he can file his work on a disk-storage device so it will be available the next time he needs it.

The following real-life example illustrates how Graphic ROCKET works. The original problem was to design a satellite system that would allow pictures of a hurricane to be taken and would return the photos within six hours.

After some preliminary analysis, the most promising approach was determined to be an air-launched booster using the flight plan shown here. So Graphic ROCKET will be used to build a model of the vehicle. A number of flights will be simulated using different launch conditions, pullup maneuvers, and vehicle design parameters; also the displays of the resulting performance will be compared.
### FILE PAGE

#### OPEN FILE

<table>
<thead>
<tr>
<th>RECALL ITEM</th>
<th>SEQNO.</th>
<th>DESCRIPTION</th>
<th>FILE CURRENT RUN AS ITEM NO.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>A ROCKET ENTRY ATTEMPT</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>123</td>
<td>LAUNCH BURGAR</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2</td>
<td>ATTEMPTED ENTRY WITH TABLES</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>122</td>
<td>LAUNCH SIMULATION</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>-555</td>
<td>AIR LAUNCHED SATELLITE</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>-123</td>
<td>BURGAR RUN WITH A TABLE</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>-554</td>
<td>AIR LAUNCHED SATELLITE</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>-191</td>
<td>SPUTNIK 1 MAKES A COMEBACK</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>-554</td>
<td>AIR LAUNCHED SAT-MOVIE VERSION</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>-554</td>
<td>AIR LAUNCHED SAT-REFERENCE RUN</td>
</tr>
</tbody>
</table>

#### CLEAR STORAGE AREA

- **QUIT**
- **CONTROL PAUSE**
- **HELP**
- **PAGE FORWARD LTO EARTH MODEL**
We will start with the File page on the screen, which lets us store complete runs on a disk-storage device and recall them at any time by pointing to the appropriate box with the light pen. To save time, I will read in the model from a previous run and only show a few changes before executing the run.

Note the boxes at the bottom of the page. They indicate the pages I can visit next. With the light pen, I will choose the Earth Model page.
EARTH MODEL

ENTER ANY COMMENTS IDENTIFYING YOUR RUN HERE:

AIR LAUNCHED SATELLITE
11-12-68

FILL IN DESIRED VALUES FOR YOUR VEHICLE AND FLIGHT PLAN:

RUN IDENT. NO. (3 DIGITS) 355.05519

EARTH OBLATENESS

\( \times \) SPHERICAL EARTH MODEL
\( \square \) OBLATE EARTH MODEL

EARTH ROTATION

\( \times \) NONROTATING EARTH MODEL
\( \square \) ROTATING EARTH MODEL

ATMOSPHERE OPTION

\( \square \) NO ATMOSPHERE
\( \times \) EXPONENTIAL ATMOSPHERE
\( \square \) TABULAR ATMOSPHERE NO. 320000

SUN-MOON EFFECTS

\( \square \) NO SPECIAL ACTION
\( \times \) COMPUTE LUNAR ACCELERATIONS
\( \square \) COMPUTE SOLAR ACCELERATIONS
\( \square \) COMPUTE LUNAR & SOLAR ACCELERATIONS
The Earth Model page allows us to specify: 1) the type of atmosphere model desired, 2) if we want an oblate or spherical earth, and 3) whether a rotating or non-rotating earth is desired.

We have asked for a tabular atmosphere and given Graphic ROCKET a table number (#2). Let us go to the Table page.
TABLE PAGE
LISTS OF AVAILABLE TABLES

- ATMOSPHERE (1-15) -
- PROPULSION (16-30) -
- GUIDANCE (31-46) -
- AXIAL AERO (46-60) -
- NORMAL AERO (61-75) -
- MOMENTS (76-88) -

1. 1959 ARDC MODEL ATMOSPHERE
2. 1963 U.S. STANDARD ATMOSPHERE
3. MIL-STD-21A TROPICAL ATMOSPHERE 30 NOV 1954
4. MIL-STD-21A POLAR ATMOSPHERE 30 NOV 1958
5. SPECIAL ATMOSPHERE FOR J. LEEDEKSE
6. SPECIAL ATMOSPHERE PLUS 2 SIGMA

SELECT TABLES FOR SECTION 0

- ATMOSPHERE 2.809506
- PROPULSION 0
- AXIAL AERODYNAMICS 0
- NORMAL AERODYNAMICS 0
- PITCH GUIDANCE 0
- YAW GUIDANCE 0
- MOMENTS [XCG & MI VS. WSP] 0
- ACP VS. ALPHA 0

READ IN A TABLE  HELP  CHECK INPUTS  RETURN
This page shows the tabular functions available on disk and the ones we have elected to use during this run. The upper half shows the names of the atmosphere tables; we can see the names of the other functions by pushing the appropriate box.

Now we will return to the Earth Model page and then go directly to the Initial-Conditions page.
INITIAL CONDITIONS

INITIAL CONDITION OPTION:

- GRODST. LAT. & EARTH VEL.
- GRODST. LAT. & INERTIAL VEL.
- GRODST. LAT. & EARTH VEL.
- GRODST. LAT. & INERTIAL VEL.
- INITIAL X, Y, Z, & X, Y, Z
- INITIAL ORBITAL CONDITIONS

TIME (SEC): 0
ALTITUDE (FT): 0
LATITUDE (DEG): 0
LONGITUDE (DEG): 0
VELOCITY (FT/SEC): 775,000
FLIGHT PATH ANGLE (DEG): 0
FLIGHT AZIMUTH (DEG): 90
WEIGHT (LB): 59,950,000
ANGLE OF ATTACK (DEG): 0
SIDESLIP ANGLE (DEG): 0

HELP!
This page lets us specify how we will put in our initial conditions and then provides us with the appropriate fields for the values of the initial conditions. Thus, if we specify "Initial Orbital Conditions" with the light pen, a new set of fields for providing, for example, initial semimajor axis, eccentricity, and inclination will replace the fields that provide altitude, latitude, longitude, and others.

The last run had an initial flight-path angle of 0°. We want to see how our performance is affected if we change this to 10°, so we will make this change via the on-line keyboard.
HELP FOR INITIAL CONDITIONS

* FOR MORE RIGOROUS DEFINITIONS OF ANGLES, SEE MANUAL.

* GEOCENTRIC AND GEODENTIC LATITUDES ARE SLIGHTLY (>1°) OF ARC) DIFFERENT ON AN OBLATE EARTH. THEY ARE EQUAL ON A SPHERICAL EARTH. WHEN IN DOUBT, GEOCENTRIC IS THE MOST COMMON USAGE.

* EARTH VELOCITY IS MEASURED RELATIVE TO A ROTATING EARTH. INERTIAL VELOCITY IS THEN EQUIVALENT TO EARTH VELOCITY + EARTH ROTATION RATE.
In case you do not know what a flight-path angle is, Graphic ROCKET has a "Help" page that gives more detailed definitions of some trajectory jargon. Most Graphic ROCKET control pages have similar "Help" pages. These make Graphic ROCKET self-contained; you do not need expert explanations continually. Here we see that the flight-path angle is the angle between the local horizontal (parallel to the earth's surface) and the velocity vector (the direction the vehicle is going).
Now we would like to increase the pullup angle to go along with the higher initial flight-path angle. How do we do this? This is a type of vehicle guidance, so we will go to the Control page to see how to arrive at guidance.

The Control page, which can be reached from every basic Graphic ROCKET page, shows how all the pages of Graphic ROCKET are connected—you can see the File, Earth Model, and Initial Conditions pages that we have worked on. Normally a user progresses along the circle, specifying section conditions (also, e.g., aerodynamics and propulsion) and then specifying and viewing his output curves. But Graphic ROCKET does not require this order. The user can hop back and forth along any of the arrowed lines on the Control page.

We see that "guidance" is reached via the Section Conditions page; we will go there next.
AIR LAUNCHED SATELLITE FOR HURRICANE PHOTOGRAPHY

CIRCULARIZE ORBIT

SECTION 2
SUSTAINED CONSTANT ATTITUDE
BOOSTER PULLUP

SECTION 3
SUSTAINED CONSTANT ATTITUDE
BOOSTER PULLUP

SECTION 1
BOOSTER PULLUP
Before we explain Section Conditions, let us return to our picture. The model of our vehicle must be built in sections to account for the differences in flight plan and vehicle configuration along the trajectory.

In Section 1 we will fire the booster engine and have the vehicle do a pullup maneuver by keeping the vehicle's axis at a high angle of attack above the flight path. After we have reached a satisfactory attitude angle, we will stop Section 1 and start Section 2, which guides the booster at a constant attitude angle. When the booster stage burns out, we switch to Section 3, where we drop off what is left of the booster and continue with a smaller sustainer engine for our propulsion. We will talk about the coast phase later.
### Section Conditions

<table>
<thead>
<tr>
<th>Section</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>☑</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Same</td>
<td>☑</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Disp</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **None**: No conditions for this section
- **Same**: Same as previous section
- **Disp**: Display section conditions for modification

---

**Page Back (Tracking)**

**Control Pause**

**Page Forward (Display List)**
Through the Section Conditions page, Graphic ROCKET allows us to define ten different trajectory sections. In each section we can choose different ways of specifying, for example, termination conditions, outputs, aerodynamics, propulsion, and guidance. Each one has three standard options: 1) none, 2) same as the previous section, or 3) display for modification. Let us see how this works in our example.

We want to modify the pullup maneuver, which is in the first section of the trajectory, so we will indicate Section 1 with the light pen. Then we will go to the guidance line and hit the "display for modification" box.
SECTION 1
PITCH GUIDANCE

PITCH GUIDANCE OPTION:
- None
- Constant Control Angle
- Constant Control Angle Rate
- Control Angle From Table
- Control Angle Rate From Table
- Extra Pitch Guidance Option 1
- Extra Pitch Guidance Option 2
- Vertical Rise
- Special Option No. (2, 4, 0) 1.000000

PITCH CONTROL ANGLE OPTION:
- None
- Angle of Attack, Alpha
- Attitude Angle, Theta
- Inertial Attitude Angle, Theta
- Flight Path Angle, Gamma (Thrust Control)
- Flight Path Angle, Gamma (arrow, Control)
- Thrust Deflection Angle, Tau, Alpha
- Extra Control Angle Option
- Special Option No. (2, 4, 0) 1.000000

<table>
<thead>
<tr>
<th>TILT ANGLE (Deg)</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>PITCH GUIDANCE MULTIPLIER</td>
<td>1.000000</td>
</tr>
<tr>
<td>CONTROL ANGLE (Deg)</td>
<td>0</td>
</tr>
<tr>
<td>CONTROL ANGLE RATE (Deg/Sec)</td>
<td>0.0</td>
</tr>
<tr>
<td>PITCH FUNCTION NO. (see tables)</td>
<td>1</td>
</tr>
</tbody>
</table>

TO YAW GUIDANCE OPTIONS

<table>
<thead>
<tr>
<th>LIST</th>
<th>TABLES</th>
<th>CHECK</th>
<th>INPUTS</th>
<th>RETURN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
This page provides all the available options for guiding the vehicle along the direction of flight. In the previous run, we see that we were using the angle of attack to control the vehicle's attitude: we wanted this angle held constant at a value of $20^\circ$ through the first section. For this run we will change the angle of attack to $25^\circ$ at the keyboard.

You can see that this page allows us to specify several different options for guiding the vehicle. For example, if we wanted to use a table of inertial attitude angles versus time, we would indicate the boxes for "control angle from the table" and the "inertial attitude angle"; then we would specify the table number either directly or from the Table page.
HELP FOR GUIDANCE

If side forces are small, you can assume that all the vectors shown lie in the same plane (the plane of the page) and then:

Vehicle attitude angle = Angle of attack + Flight path angle + Inertial attitude angle + Inertial range angle

Thrust control varies the thrust deflection angle (\( \tau_{\alpha \tau} \)) to achieve the desired flight path angle. Aero control varies the angle of attack (\( \alpha \)).

A tilt angle instantaneously tilts the vehicle (and the velocity vector) at the start of the section. It is essentially equivalent to a rather laborious use of the pitch control options.
If we are not sure how all these angles are defined, we can go to a "Help" page that provides a diagram showing how the flight-path angle, angle of attack, attitude angle, and inertial attitude angle are related. A "Return" box brings us back to the Guidance page, where another "Return" takes us to Section Conditions.
## Section Conditions

<table>
<thead>
<tr>
<th>Section</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>NONE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAME</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DISP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

- **NONE** - No conditions for this section
- **SAME** - Same as previous section
- **DISP** - Display section conditions for modification

---

**Control Page**

- Page Back (Tracking)
- Control Page
- Page Forward (Display List)
For this run, we would also like to have the vehicle fly at a higher angle in the second and third sections. First, we will indicate Section 2; then hit "guidance-display" again.

The Pitch Guidance page reappears, now indicating we are in Section 2. Once again we are holding a constant control angle, but this time it is the vehicle-attitude angle. The old value is 25°, but we will change that to 40° on this run and then "Return" to the Section Conditions page.

During Section 3, the sustainer-engine stage, we want the vehicle to hold the same attitude angle as in Section 2. We can do this easily by changing to Section 3 and simply hitting the "guidance-same" box.

Everybody must verify at least once that this works, so we will page to "guidance-display". We now have 40° for the value of the control angle.

We are ready to run the trajectory, but first we must specify which trajectory variables we want to display. So we will return first to the Section Conditions page and then to the Display List page.
### List of Variables Available for Display

<table>
<thead>
<tr>
<th>Basic</th>
<th>Aerodynamic</th>
<th>Orbital</th>
<th>Guidance</th>
<th>Tracking</th>
<th>Special</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1-16)</td>
<td>(17-32)</td>
<td>(33-48)</td>
<td>(49-64)</td>
<td>(65-74)</td>
<td>(75-110)</td>
</tr>
</tbody>
</table>

- **33** Semi-major axis, osculating orbit
- **34** Eccentricity
- **35** Inclination
- **36** Longitude, ascending node
- **37** Argument of perigee
- **38** True anomaly
- **39** Inertial velocity
- **40** Inertial flight path angle
- **41** Orbital energy
- **42** Apogee altitude
- **43** Perigee altitude
- **44** Surface range to impact
- **45** Velocity at apogee
- **46** Circ. velocity at apogee
- **47** But. in circ. orbit, apogee

### Select Display Variables

**SAVE IN AREA #**  |  **VARIABLE #** |  **IDENTIFIER (OPTIONAL)** |  **1: CURRENT RUN** |  **2: PREVIOUS RUN** |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>HIGI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>ALTI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHANGE</td>
<td>3</td>
<td>WC1T</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AREA No.</td>
<td>4</td>
<td>HAP1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>47</td>
<td>WC1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>RZ0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>RZ0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>RZ0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>42</td>
<td>HAP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>47</td>
<td>RZ10</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- [Page Back: (To Section Comds.)]
- [Control Page:]
- [Help:]
- [Page Forward: (Construct Display:)]
The bottom half of this page provides ten areas in which we can store values of a trajectory variable for display. To save time, we have set these up in advance. To find out to which variables the variable numbers refer, we use the lists on the top half of the page: #8 is range, #2 is altitude, #9 is weight. To find the others, we hit the appropriate box: #42 is apogee altitude and #47 is the final weight in circular orbit at apogee. Since these two variables are the ones that really help us to evaluate our design, let us return to our picture and explain them.
AIR LAUNCHED SATELLITE
FOR HURRICANE PHOTOGRAPHY

CIRCULARIZE ORBIT

COAST TO ApOGEE

SECTION 3
(SUSTAINED CONSTANT ATTITUDE)
BOOSTER (POWDER ON)

SECTION 2
(BOOSTER CONSTANT ATTITUDE)

SECTION 1
(BOOSTER PULLUP)
If we burn out the sustainer engine at the point indicated, the vehicle will coast to the indicated apogee altitude where we can apply more thrust to circularize the orbit. We could create a fourth and fifth section to do this, but some analytic formulas will compute the apogee altitude and the final weight in circular orbit directly from the vehicle's position and velocity at this point.

Not only does this save time, but it lets us do a similar calculation at other points on the trajectory. For example, if the sustainer stopped thrusting at any previous point, the vehicle would coast to a lower apogee but would have more final weight in circular orbit at this altitude. We could do this, using the current vehicle design, for each point along the trajectory, producing a curve that shows the orbital payload weight attainable at each orbital altitude.

This curve, which is the most interesting for evaluating different designs, can indicate which designs give us the most payload in orbit for any range of orbital altitudes. Now we will go back and watch Graphic ROCKET produce some curves like this.
<table>
<thead>
<tr>
<th>AREA NO.</th>
<th>IS STORED NO</th>
<th>IDENT</th>
<th>VARIABLE</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>RNG1</td>
<td>RANGE</td>
<td>NM</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>ALT1</td>
<td>ALTITUDE</td>
<td>FT</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>WGT1</td>
<td>WEIGHT</td>
<td>LB</td>
</tr>
<tr>
<td>4</td>
<td>42</td>
<td>MAF1</td>
<td>M APOG</td>
<td>NM</td>
</tr>
<tr>
<td>5</td>
<td>47</td>
<td>WPC1</td>
<td>WP CIRC</td>
<td>LB</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>R20</td>
<td>RANGE</td>
<td>NM</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>H20</td>
<td>ALTITUDE</td>
<td>FT</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>R20</td>
<td>WEIGHT</td>
<td>LB</td>
</tr>
<tr>
<td>9</td>
<td>43</td>
<td>HA20</td>
<td>M APOG</td>
<td>NM</td>
</tr>
<tr>
<td>10</td>
<td>47</td>
<td>WPC20</td>
<td>WP CIRC</td>
<td>LB</td>
</tr>
</tbody>
</table>

**SPECIFY AXES BY**

<table>
<thead>
<tr>
<th>AREA</th>
<th>VARIABLE</th>
<th>LOWER LIMIT</th>
<th>UPPER LIMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>INDEPENDENT VARIABLE</td>
<td>.4</td>
<td>.8888</td>
<td>132.8888</td>
</tr>
<tr>
<td>DEPENDENT VARIABLE 1</td>
<td>.2</td>
<td>.9999</td>
<td>13223232</td>
</tr>
<tr>
<td>DEPENDENT VARIABLE 2</td>
<td>.2</td>
<td>.9999</td>
<td>13223232</td>
</tr>
</tbody>
</table>

**CHANGE UNITS**

CHANGE VARIABLE NO. ___ TO UNITS OF ________

1

**JUMP BACK**

(To Sect. Comb.)

**PAGE BACK**

(Change Stored Vars.)

**CONTROL**

PAGE

**HELP**

(Compute & Display)
We will go to the Display-Control page and indicate which variables will be examined first. Let us use range as our independent variable (#8). As we enter it on the keyboard, some suggested lower and upper limits for the graph appear. We will accept these; however, we can always change them if necessary. Similarly, we specify altitude (#2) and weight (#9) as our dependent variables.

Now we can run the trajectory and display the resulting performance curves by hitting the "Compute & Display" box.
First, we see the axes and the labels: altitude and weight versus range. Then a pair of curves appears, their labels indicating that they refer to our previously stored comparison run. Then the corresponding curves for our current run appear as they are calculated.

A "grid" button allows us to see where we reach 100,000 feet, for example. If we want to change scale or change variables on any of the axes, we can simply return to the Display Control page and do so.
Observing the comparative curves of weight in circular orbit versus orbital altitude, we see that the higher initial flight-path angle and pullup angles have given about 10 to 20 percent extra payload.
SECTION 1
NORMAL AERODYNAMICS OPTIONS
NORMAL AERODYNAMICS OPTION:

- KONE
- CONSTANT NORMAL AERO. COEF. CN
- CONSTANT LIFT/DRAG OR CN/CA RATIO
- CONSTANT CN/ALPHA (ANGLE OF ATTACK, DEG.)
- CN/ALPHA FROM TABLE
- CN FROM TABLE
- EXTRA NORMAL AERO. OPTION 1
- EXTRA NORMAL AERO. OPTION 2
- SPECIAL OPTIONS

CN/ALPHA (ANGLE OF ATTACK, DEG.) 15.00001

NORMAL AERODYNAMICS MULTIPLIER 0
Now we will go back through the Control page and the Section Conditions page to the aerodynamics pages, since we want to see how much our performance would be improved if we could increase aerodynamic lift by a factor of two.

The lift coefficient has been specified as proportional to the angle of attack, so we double the proportionality constant, return to the Section Conditions page, and specify that the aerodynamics in the second and third sections will be the same as in Section 1.

After we have gone back through the Control page to the Display List page, we must decide which of our previously stored outputs to discard to make room in our ten display areas for the outputs of our current run. After we decide, we go to Display Control to set up our display axes; then the machine computes and displays the comparative graphs.
Display: Air-Launched Satellite

Altitude (ft)

Range (n mi)

Page back (to change display)  File or quit  Griu  Control page  Hold  Go
You are seeing the graph appear about three times as fast as it actually happened since we had to film the action at one-third normal speed to avoid producing an artificial flicker on the scope. The apparent speed is similar to what we would get on an IBM 360/50, instead of the model 40 we have here.†

The previous graph showed that the higher lift produced a more lofted trajectory. Now we will set up the display to observe the difference it made in the possible payload at various orbital altitudes.

†The actual computation time on the 360/40 for this trajectory was about 25 sec.
As the new graph of weight in circular orbit versus orbital altitude appears, we see that the added lift gave some added payload at the lower orbital altitudes, but made very little difference at the higher altitudes. On this basis, we can decide whether or not to investigate lifting boosters in any more detail.

Similarly, we can change other features of the vehicle design (e.g., propulsion, staging, guidance plans) and quickly get a feel for the design regions that look promising. At any time we can push a button that will transfer our current display to an S-C 4060 film plotter for hardcopy. Also, we can file our current inputs and outputs at any time.

Graphic ROCKET has been tried on several design problems at RAND and has proved quite useful so far in giving engineers a quick and thorough feel for the implications of possible design choices. The program will run on any IBM 360-model 40 and also on any higher-numbered models, if they are equipped with disk packs and an IBM 2250 CRT terminal.
Appendix

GLOSSARY OF INTERACTIVE GRAPHICS TERMINOLOGY

**Button (control box, control area)**

A user-specified rectangular area in a CRT display. A tablet-stylus or light-pen interrupt whose coordinates lie within this area sets a flag that can be detected by the user's program.

**Control area**

See "button."

**Control box**

See "button."

**CRT (Cathode Ray Tube)**

An electronic device consisting of an electron tube with a phosphorescent glass face and an electron beam that strikes this face. Under control of a computer, the beam can be used to draw lines and characters on the face of the CRT.

**Cursor**

A small bright underscore that appears in a row of characters on a CRT display. The cursor indicates the position at which data will be entered from the typewriter keyboard. Normally, the cursor is advanced one space every time a character is typed. Special keys on the typewriter keyboard can be used to advance, backspace, or jump the cursor.

**Display (page)**

A set of information that appears on the CRT face, and that can be manipulated by a computer program as a single entity. A display may consist of any combination of lines and characters. More than one display may be on the face of the CRT at a given time.
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hit</td>
<td>See &quot;push.&quot;</td>
</tr>
<tr>
<td>Light pen</td>
<td>A device that can be used to input information to a computer by pointing</td>
</tr>
<tr>
<td></td>
<td>to a lighted area on the CRT face. A light pen does not have the accuracy</td>
</tr>
<tr>
<td></td>
<td>or flexibility of the RAND Tablet; however, it is much cheaper.</td>
</tr>
<tr>
<td>Page</td>
<td>See &quot;display.&quot;</td>
</tr>
<tr>
<td>Push (hit)</td>
<td>To cause a light-pen or tablet-stylus interrupt within the area of</td>
</tr>
<tr>
<td></td>
<td>a &quot;button&quot; (q.v.).</td>
</tr>
<tr>
<td>RAND Tablet</td>
<td>A graphic input device invented at RAND that allows a computer user to</td>
</tr>
<tr>
<td></td>
<td>communicate with a computer by drawing freehand on an electronic &quot;tablet&quot;</td>
</tr>
<tr>
<td></td>
<td>surface with a special stylus.</td>
</tr>
<tr>
<td>Screen</td>
<td>The face of a CRT.</td>
</tr>
<tr>
<td>Scrolling</td>
<td>To move a virtual display, only a portion of which can be seen on the</td>
</tr>
<tr>
<td></td>
<td>screen at any given time. A &quot;Scroll&quot; is moved by &quot;pushing&quot; either of two</td>
</tr>
<tr>
<td></td>
<td>arrows attached to the scroll display. One arrow moves the scroll</td>
</tr>
<tr>
<td></td>
<td>forward, the other backwards. In this way, portions of the scroll move</td>
</tr>
<tr>
<td></td>
<td>into view, and other portions move out of view.</td>
</tr>
</tbody>
</table>
REFERENCES

