N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED
IN THE INTEREST OF MAKING AVAILABLE AS MUCH
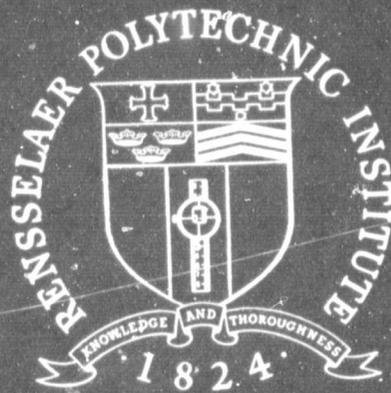INFORMATION AS POSSIBLE

Rensselaer Polytechnic Institute

Troy, New York 12181

RPI TECHNICAL REPORT MP-67

AN ADVANCED TERRAIN MODELER
FOR AN AUTONOMOUS PLANETARY ROVER

by

Erwin L. Hunter

. A Study Supported by the

National Aeronautics and Space Administration

Grant NSG-7369

School of Engineering
Rensselaer Polytechnic Institute
Troy, New York

July 1980

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

ABSTRACT

A roving vehicle capable of autonomously exploring the surface of an alien world is under development at Rensselaer Polytechnic Institute. An advanced terrain modeler to characterize the possible paths of the rover as hazardous or safe is presented in this report. This advanced terrain modeler has several improvements over the Troiani modeler that include: A crosspath analysis, better determination of hazards on slopes, and methods for dealing with missing returns at the extremities of the sensor field. The results from a package of programs to simulate the roving vehicle are then examined and compared to results from the Troiani modeler.

ACKNOWLEDGEMENT

1. INTRODUCTION

Although present funds for extraterrestrial exploration have been reduced for the immediate future, it is highly probable that at some time in the future national interests will once again turn towards expanding our knowledge of the solar system we live in. When man does decide to explore new worlds it will be necessary to have the required technological background to proceed with the exploration so interest in the project will not be lost while the technology base is being developed. Therefore, it is important to develop this technology required for the exploration of alien worlds now, before the actual exploration mission is planned.

When it is decided to explore an alien world, such as Mars, the first step will be to send robot explorers to investigate the world and provide for the maximum safety of human explorers, if they follow. A major advantage of these robot explorers is that a great deal of expense will be saved over a human exploration mission. The main savings will be from excluding a return flight in the mission, thus costly and heavy return rockets do not have to be launched with the mission. Also more risks may be taken since a

mission failure will not result in the loss of human life. This could result in more exploration and studies being conducted as well as less money spent on the project.

There are many approaches being studied for possible robot explorers spanning the range in sophistication from "dumb", uncontrolled, tumbleweed type explorers to rocket powered gliders capable of many take offs and landings. One practical concept that offers a high degree of controlled mobility in this range of robot explorers is a roving vehicle.

There are two sensor systems that would be necessary for a surface roving type vehicle. The first sensing system is a long range sensor that would provide the rover with the general route to follow to the destination while avoiding major terrain features that would slow down the progress of the rover. This sensor system is analogous to a road map for a long drive. This sensor system could be provided by satellites orbiting the planet or even pictures already taken of the surface of the planet. Large, Earth-based computer systems could spend a great deal of time preparing possible routes for the rover since the routes would not be required in a real-time sense. The second sensor system needed would be a short range system capable of identifying smaller hazards like small craters and boulders that would not be seen by the long range sensor system. This system would have to be incorporated as part

of the rover in order to detect the possible hazards and would have to make use of computing facilities on the planet or the rover because of the time delay involved in communication with Earth-based computer facilities.

There are many approaches for constructing a short range sensor system including complex video systems and simple laser-photodetector triangulation systems. Since the laser-photodetector triangulation system takes a minimal amount of data and the amount of work necessary to translate this data into usable information is small, it is the most feasible approach from a computational point of view.

The roving vehicle being developed at Rensselaer uses a laser-photodetector system as the short range sensor and previous work has been done at Rensselaer to develop terrain modelers to characterize the terrain as safe or hazardous for the rover [1,2]. These methods included pattern analysis, slope calculations using area analysis, and a sophisticated method developed by Troiani which used estimated slope and step-height calculations. The advanced terrain modeler in this report is an evolution of the Troiani concepts that refines some of the procedures and adds a crosspath analysis to the modeler.

2. SYSTEM CONFIGURATION

The planetary rover under development at Rensselaer Polytechnic Institute has been divided into two systems, the hardware system and the software system. The hardware system consists of all the mechanical and electrical systems used by the rover, the computer controlling it, and the telemetry system connecting them. The software system is composed of all the computer programs used to run or simulate the running of the rover.

2.1 Description of Hardware System

The rover consists of a platform mounted on four individually powered wheels. Mounted on the platform is a two meter mast with the elevation scanning laser at the top and the detector array half the way up the mast. The platform also holds all the mechanical and electrical elements of the rover.

Each wheel is powered by a 1/6 HP electric motor giving the entire rover the ability to climb approximately 30 degrees or to travel at approximately 0.2 meters/second on level terrain. Each wheel is one half meter in diameter giving the rover the ability to negotiate steps less than 0.25 meters or holes and trenches less than 0.25 meters wide

without difficulty.

The front wheels of the rover are attached in a manner that allows them to be individually raised and lowered without affecting the roll of the rover. Thus the roll of the rover is determined solely by the roll of the rear wheels. The pitch of the rover is determined by the average height of the rear wheels and the average height of the front wheels.

The data on the terrain in front of the rover are supplied by an elevation scanning laser/multidetector system mounted on a mast on the rover which scans through 15 azimuths as shown in Figure 1. This system will simulate up to 32 lasers by using one laser and a rotating mirror system. By precisely controlling the speed of the rotating mirror and the timing of the laser pulses it is possible to fire laser pulses in 32 different elevation angles. The system will have 40 photo-detectors, or a configuration of detectors that appears to be 40 detectors to all the rest of the systems, mounted one meter up the mast.

When a laser pulse is fired each detector responds by indicating whether it saw the reflection from the laser pulse. A two laser/two detector system is shown in Figure 2 to illustrate how a multilaser/multidetector system works. Notice that each laser pulse intersects each detector cone once creating four line segments of laser-detector
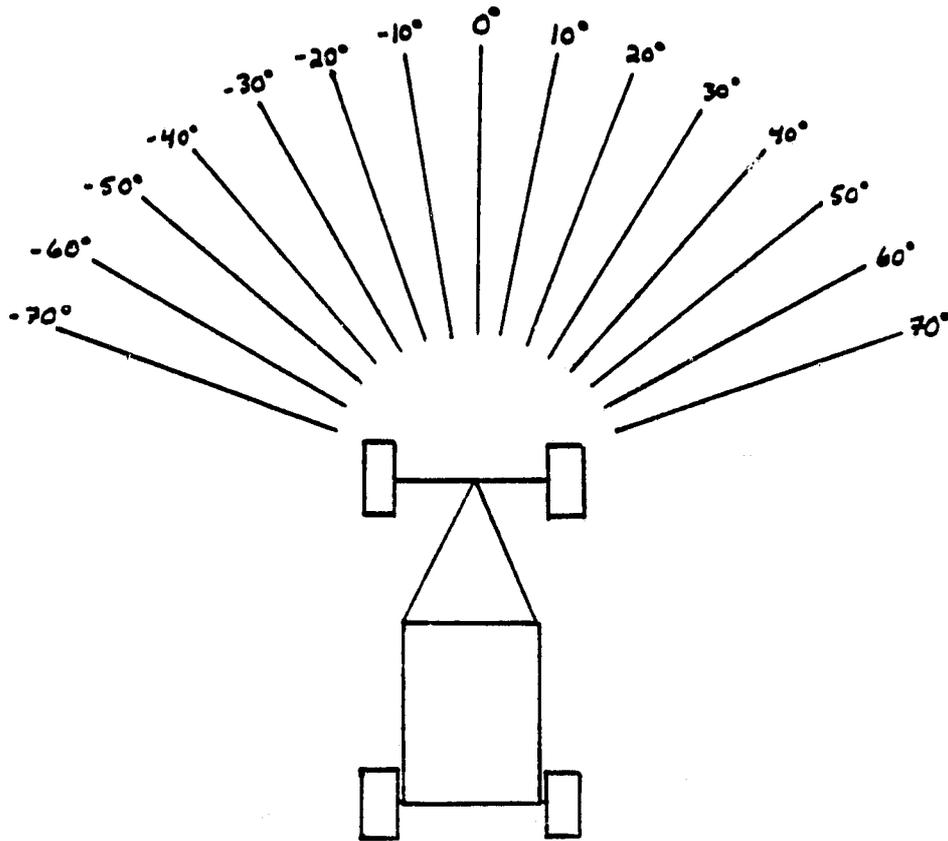
FIGURE 1

FIFTEEN AZIMUTHS OF VIEW FOR ROVER

MAST →

LASER BEAMS

2

1

DETECTORS

LINES OF
INTERSECTION
(a,b,c,d)
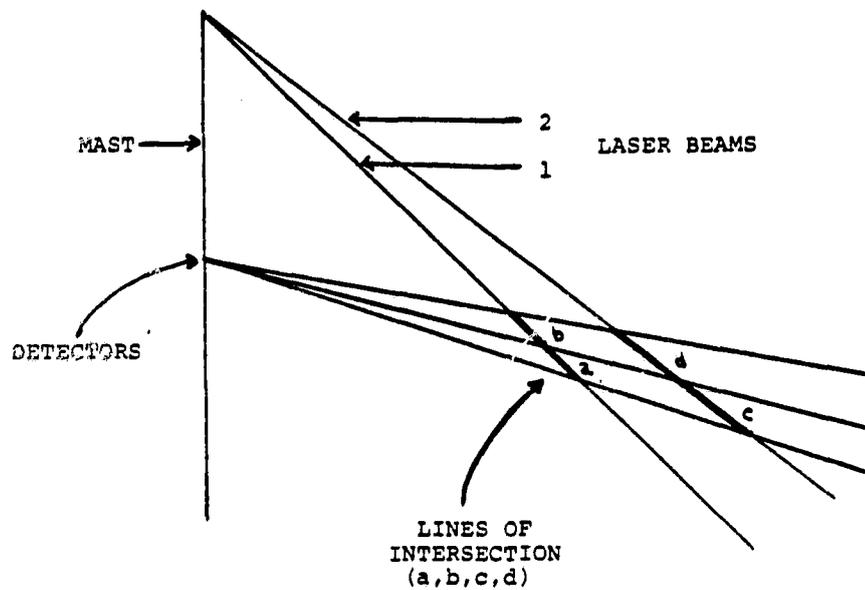
FIGURE 2

TWO LASER/TWO DETECTOR SYSTEM

intersection. If the terrain passes through any of these line segments of intersection then the laser beam will be reflected off the terrain and the detector will sense it when the laser is fired.

The computing power for the rover is provided through a telemetry link to a PRIME 550 computer. This 32-bit computer has an average instruction execution time of approximately one microsecond. The system can take up to 8 M-bytes of random access memory and is configured with two 80 M-byte disk drives.

2.2 Description of Software System

Presently there are two software packages in us_ on the Mars Rover Project. The first package is the real time software which actually controls the rover and makes the path selection decisions based on the data sent to the computer from the rover. The second package is the simulation software which models an input terrain for the modeling and path selection algorithms, then models the response of the rover from the outputs of these routines.

The real time software consists of a modeler routine, a path selection routine and several system routines. The modeler routine takes the laser-detector data from the rover and determines which of the 15 azimuths are safe for the rover to travel and which ones are not. The path selection routine uses the outputs from the modeler

- 8 -

routine along with the knowledge of previous hazards encountered and the final destination of the rover to determine an optimal path for the rover. The system routines take care of functions like translating the information from the telemetry link into a useful format for the modeler routine, translating the outputs from the path selection routine into a format that can be sent to the rover as a command, and processing the interrupts as they occur.

The simulation software consists mainly of a modeler routine, a path selection routine, a sensor routine, a terrain set-up routine, and motion routines. The modeler and path selection routines perform the same functions as in the real time software. The terrain set-up routine sets up the model for the test terrain and stores it for later reference by the other routines. The sensor routine sets up the locations of the laser-detector intersections for the particular laser-detector system being used and determines which detector sees the terrain for each laser pulse being simulated. The motion routines keep track of the pitch and roll of the rover and whether it has collided with any hazards on the simulated terrain.

The two major routines, the modeler and the path selection routines are common to both software packages, so work is in progress to integrate the two software packages into one system that uses common modeler and path selection

routines for both the real time and the simulation systems. This will eliminate the need to write and debug a modeler or a path selection routine for the simulator then, after it works with the simulator software, rewriting it for the real time software.

PART 3

3. THE TERRAIN MODELER

The terrain modeler uses the laser-detector data
to determine which of the 15 azimuths are safe for the rover
to travel and which ones are not. There are several
techniques for trying to determine whether an azimuth is
safe but the advanced terrain modeler uses an improvement on
the algorithm developed by Troiani. The slope of the
terrain is bounded and if the bounds are less than
30 degrees then the terrain is considered safe, if the
bounds are greater than 30 degrees it is considered
hazardous, and anything in between is considered uncertain.

3.1 Historical Review

In May of 1977 Gary Maroon submitted a report[1]
that conducts an initial investigation into the development
of the terrain modeler for the rover under development at
Rensselaer. This report covered a numerical technique using
estimation of slopes and a pattern analysis method.

The numerical technique determined the
intersection of each laser pulse and the center line of each
detector cone from which the laser was detected. Then the

area of the cross section along that azimuth of the terrain was generated by connecting each of these returns. From this area the slope was calculated using the formula:

$$SLOPE = TAN^{-1}\left[\frac{2 * AREA}{(LENGTH)^2}\right]$$

where the length indicated is the length of the slope being examined.

The pattern analysis technique uses differences in the laser-detector returns to classify hazards such as large boulders, craters, positive inclinations, and negative inclinations. The rules for determining these four types of hazards were determined and shown to be reasonably effective in distinguishing these hazards from safe terrain.

In May of 1978 Nicholas Troiani submitted a report[2] that presented a working and tested terrain modeler that evolved from the numerical technique presented by Maroon. Troiani simplified and improved Maroon's slope calculation technique by determining the height and range of two points on the terrain and setting the slope equal to the difference in the heights of the two points divided by the difference in range of the two points. The Troiani modeler also took into account the slope of the rover and the possibility of step hazards. This modeler detected most hazards, but like Maroon's work did not account for crosspath hazards.

Further work is in progress to develop new terrain modelers for the rover and hazard detection system at Rensselaer. This research has split into two branches, one is investigating pattern analysis methods of detecting hazardous terrains and the other is continuing the work done in the Troiani and advanced terrain modelers.

3.2 Acquisition and Representation of Data

The data is collected on each azimuth scanned by firing a laser pulse in 32 angles. Each time the laser is fired each detector returns either a zero indicating that the particular detector did not see the laser or a one indicating that the detector did see the laser.

This information is encoded immediately at the rover by sending back to the computer just the number of the detector position that saw the laser instead of the string of zeros and ones. In this manner the entire array of information for a sweep of 15 azimuths can be stored in one 15 by 32 array. Figure 3 shows a possible terrain configuration and laser-detector intersections for one azimuth on a 15 laser by 15 detector system. The return from the rover is shown below in Table 1.

FIGURE 3   SAMPLE TERRAIN AS VIEWED BY A
FIFTEEN LASER FIFTEEN DETECTOR SYSTEM
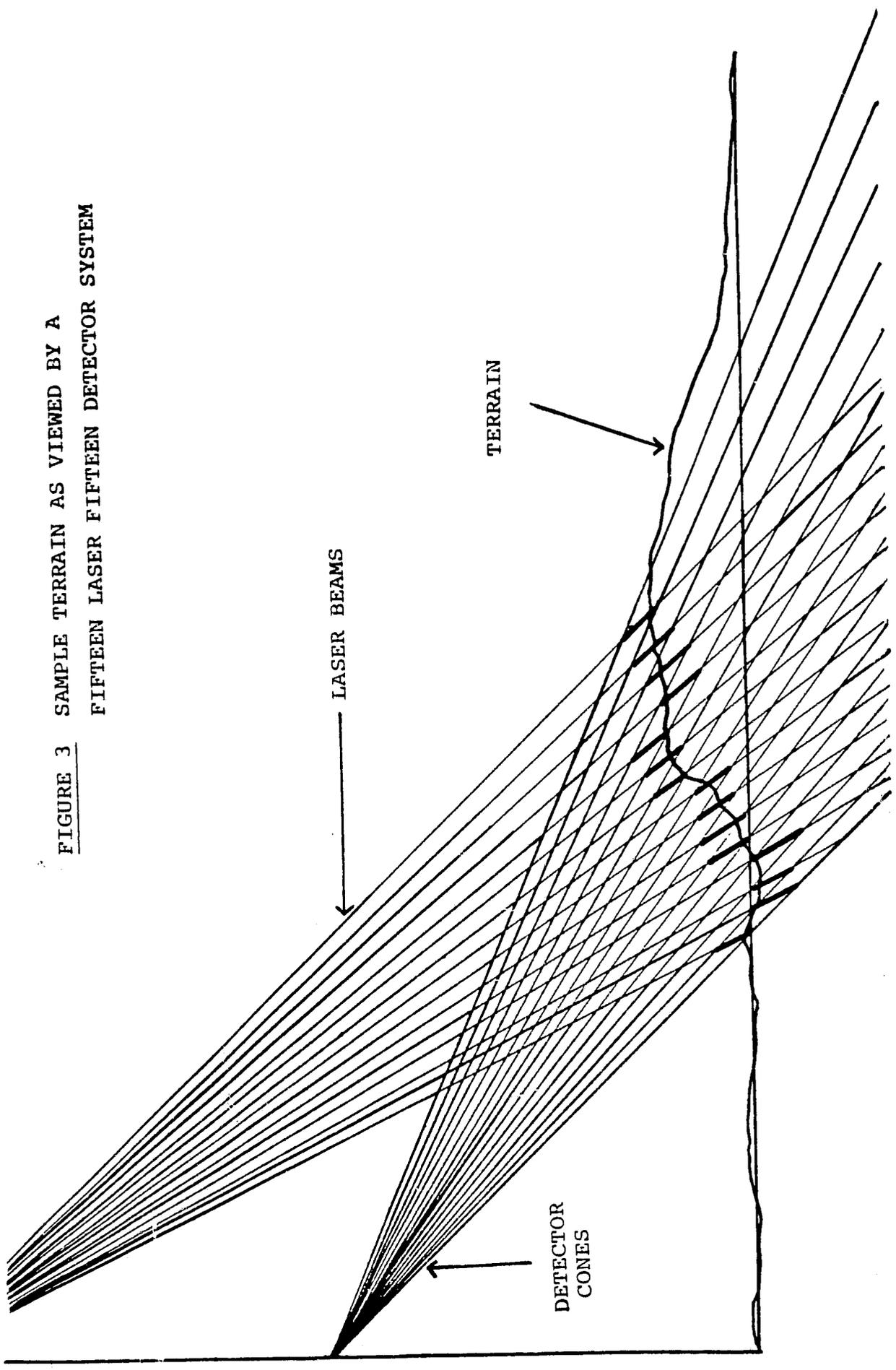
LASER 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

---

1 1 2 3 5 6 7 8 10 11 12 12 13 14 15

TABLE 1

Laser-Detector Data Returned from Terrain in Figure 3

Next these data are put into the more useful form
of an array of relative values. This is done by subtracting
the number of the laser pulse from the number of the
detector that saw the reflection from the laser. The values
in Table 1 are shown again as a relative array in Table 2.

LASER 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

---

0 -1 -1 -1 0 0 0 0 0 1 1 1 0 0 0 0

TABLE 2

Relative Array Constructed from Data in Table 1

In order to derive the most information from the
relative array the laser-detector system must be set up as a
quasi-linearized array. In a quasi-linearized array each
laser is aimed to intersect the center of the corresponding
detector cone at the position of level terrain.   The
corresponding detector cone is that cone which has a number
that is the same as the number of the laser pulse.   Thus,
the array of relative values indicates the extent to which
the actual terrain differs from being level as shown in
Figure 4. If the detector that saw the laser pulse is above
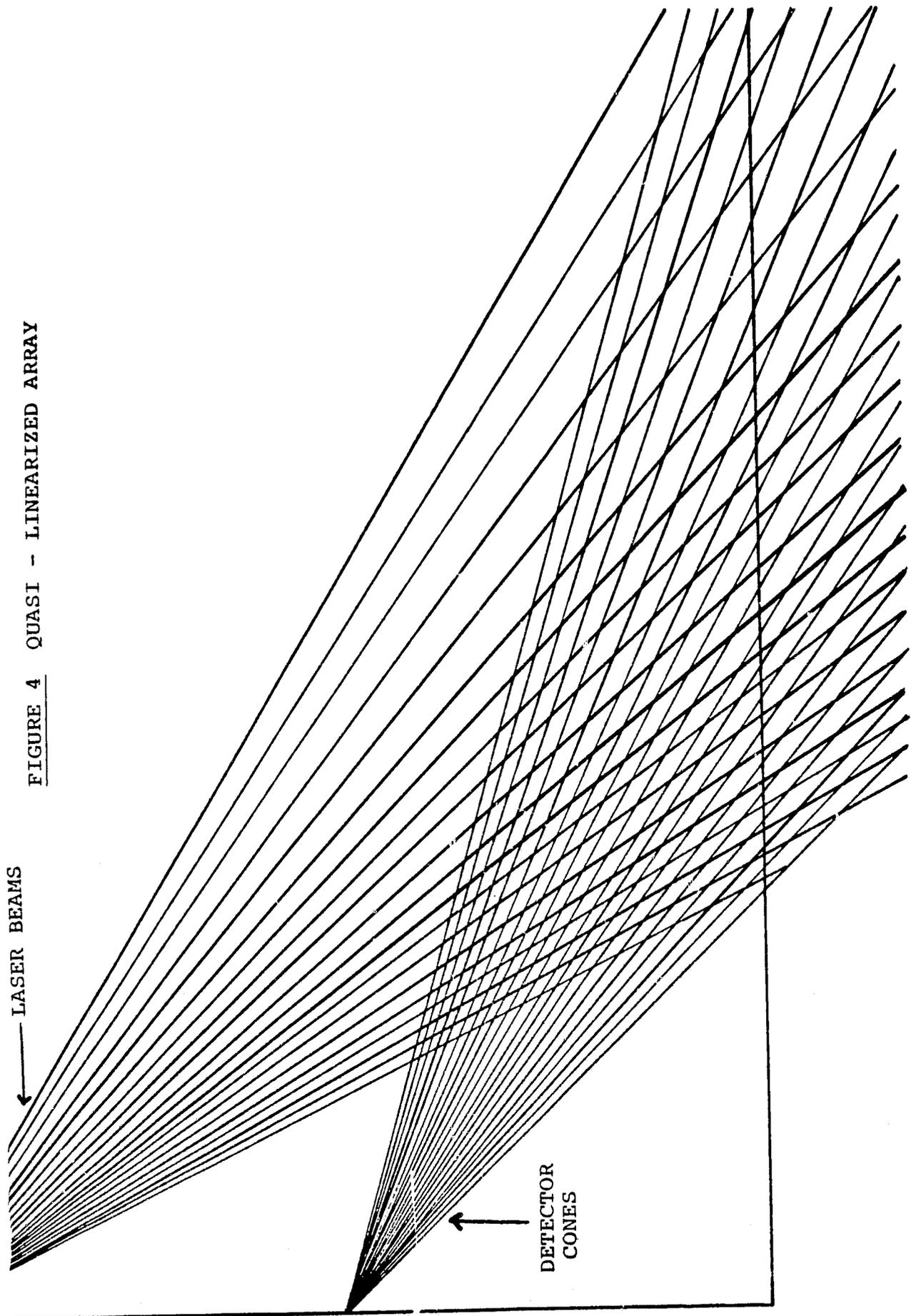
FIGURE 4   QUASI – LINEARIZED ARRAY

LASER BEAMS

DETECTOR CONES

the detector that would have seen level ground, then the corresponding value in the relative array is positive. On the other hand, if the detector that saw the laser shot is below the detector that would have seen level ground, then the value in the relative array is negative.

The reason the rover system at Rensselaer has 40 detectors but only 32 lasers is because of this quasi-linearized array set up. By examining Figure 4 it can be seen that the last laser pulse (the higher numbered laser pulses) have to travel further before they intersect a detector field. In many laser-detector configurations this means that the distance to most of the detector cones is out of the range of the laser. Since these lasers add no useful data to the system, they are simply not included in the system design.

A missing return occurs when none of the detectors indicate that the reflection from the laser pulse was seen, as would happen if the rover was looking at a deep hole or other terrain feature that blocked the detectors from seeing the laser pulse. Some terrain features that cause missing returns are shown in Figure 5. When a missing return is encountered the rover returns a zero for the laser firing. When the return is transformed into a relative array this zero is changed into three asterisks. The return from the rover and relative array for the azimuth in Figure 5 are shown in Table 3 and Table 4 respectively.

- 17 -

FIGURE 5   SAMPLE TERRAIN CAUSING MISSING RETURNS
ON A 15 LASER/20 DETECTOR SYSTEM

TERRAIN

LASER BEAMS
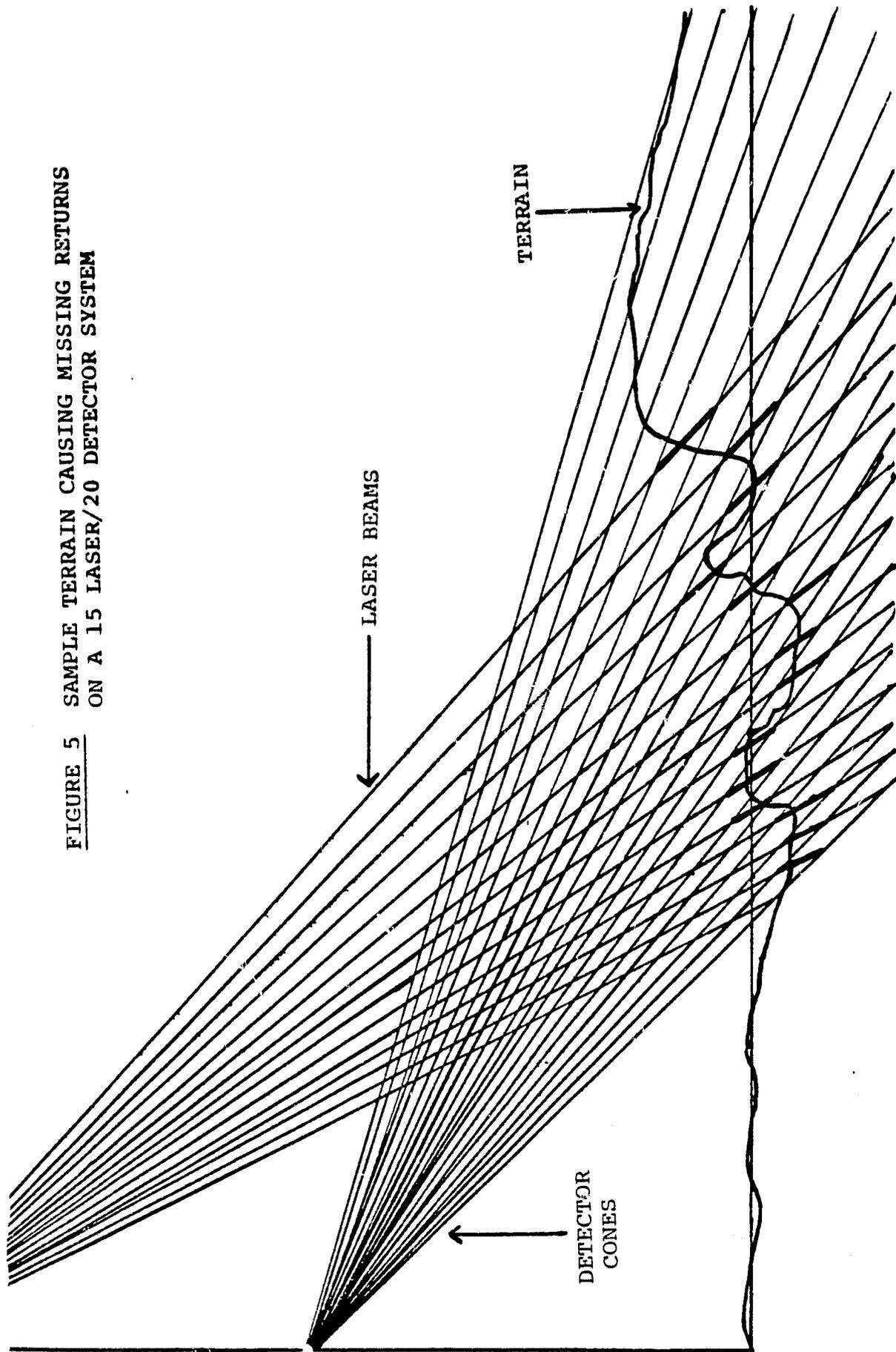
DETECTOR
CONES

LASER	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	0	1	2	3	5	6	7	0	8	9	11	13	0	14	16

TABLE 3

Laser-Detector Data Returned for the Terrain in Figure 5

LASER	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	***	-1	-1	-1	0	0	0	***	-1	-1	0	1	***	0	1

TABLE 4

Relative Array for the Terrain in Figure 5

One problem not yet discussed is the quantization
of the data. It should be noted that each detector cone
does not see a point but instead sees the intersection of a
line and a cone which is a line segment. Thus, if a
detector sees the laser reflection, this indicates that the
terrain passes somewhere through this line segment. In
theory this line segment can be made arbitrarily small by
decreasing the size of the sensor cone at the cost of adding
more detectors to the system or making the overall detector
field smaller.

This quantization of the input data means that the
exact position of the terrain can not be known in a real
system. The best that can be done is to construct an

envelope that contains all the possible terrains representable by this data. Then if all possible terrains in this envelope are safe for the rover the terrain represented by the data must be safe. Likewise if all possible terrains in this envelope are hazardous for the rover, the terrain represented by the data must be hazardous. The design objective in setting up a fixed number of detector cones is to minimize the number of possible safe terrains that will be classified in the same envelope as hazardous terrains while keeping the overall detector field large enough to see that safe terrain is not avoided because it can not be seen.

A significant advantage of the quantization is that terrain features with a low amplitude will in a sense be filtered out. Any terrain feature with an amplitude less than one half the length of the line segment of the laser-detector cone intersection will generally not appear in the laser-detector data. This inherently decreases the amount of information that the computer system must process, making the elevation scanning laser/multidetector system much more feasible computationally.

Previous reports have suggested that the high frequency components of the terrain are also lost because of the sampling effect associated with the discrete lasser-detector system, but this turns out not to be so. Even in the unlikely situation of poles or spikes pointed directly

at the laser in such a manner that none of the lasers hit them, as shown in Figure 6, the high frequency components can still be detected. In Figure 6 the spikes shown generate missing returns and give the array of relative returns shown in Table 5.

LASER	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	0	0	0	1	***	***	***	***	0	0	0	***	***	***	***

TABLE 5

Relative Array for Terrain in Figure 6

FIGURE 6   HIGH FREQUENCY TERRAIN COMPONENTS

LASER BEAMS

DETECTOR CONES

TERRAIN
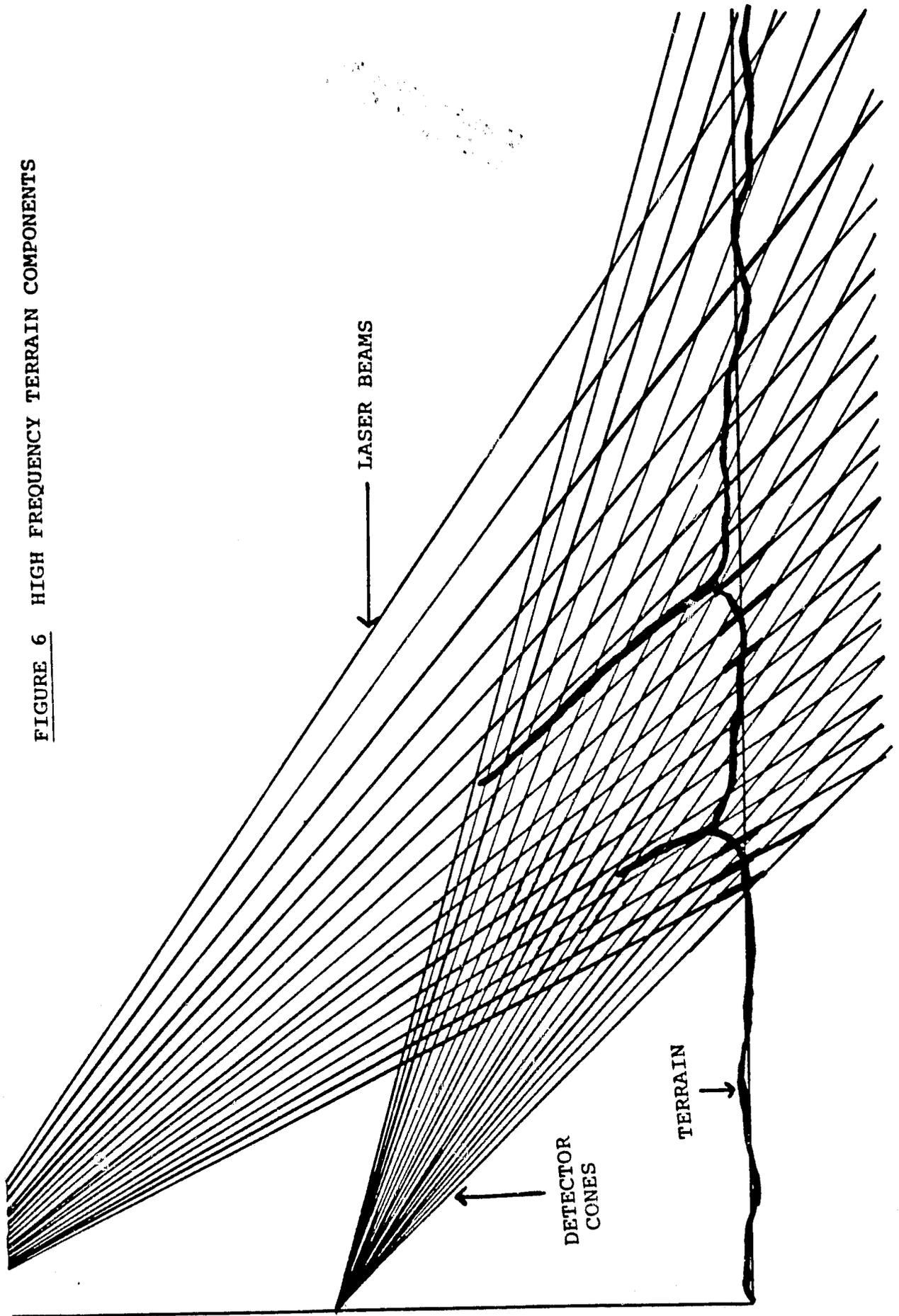
3.3 Definition of a Hazard

The definition of a hazard for the planetary rover
at Rensselaer has been kept simple to minimize the amount of
computation that must be done to find the hazards, while
remaining elaborate enough to encompass all possible hazards
the rover might run into. The rover has basic capabilities
and limitations that will be used to define a hazard.
Because the tires on the rover are 0.5 meters in diameter,
steps greater than 0.25 meters and craters and trenches
wider than 0.25 meters can not be negotiated.   As stated
previously in this report, the motors are not powerful
enough to bring the rover up a slope greater than 30 degrees
nor are they powerful enough to maintain control over the
rover on a downward slope less than -30 degrees.   In
addition to these specifications the roll of the rover can
not be greater than 30 degrees.   The   strictness of these
specifications is to insure the safety of the rover, but
they can easily be changed through program inputs if they
prove too strict.

Using the above specifications a hazard is defined
to be any terrain feature that would force the pitch or roll
of the rover to be greater than 30 degrees or less than
-30 degrees, would force the rover to negotiate a step
(either positive or negative in height) greater   than

0.25 meters in height, or would force the rover to cross a crater or trench wider than 0.25 meters. Later in this report this definition will be translated into an algorithm and tests from a digital computer will be examined.

3.4 Calculation of Upper and Lower Bounds on Slopes

The quantization of the laser-detector data makes it impossible to determine the actual position of the terrain within the line segment of intersection between the laser beam and the detector cone. Through any two of these line segments an infinite number of lines with widely varying slopes may be constructed as shown in Figure 7. Therefore, since the actual slope of the terrain can not be computed from the laser-detector data the upper and lower bounds on the magnitude of the slope must be computed for the terrain as shown in Figure 8. If the upper bound on the magnitude of the slope is less than 30 degrees then the magnitude of the slope of the terrain that passes through these line segments must also be less than 30 degrees. Likewise if the lower bound on the magnitude of the slope is greater than 30 degrees then the magnitude of the slope of the terrain must be greater than 30 degrees. But if the upper bound is greater than 30 degrees and the lower bound is less than 30 degrees, then the magnitude of the slope can not be classified as less than or greater than 30 degrees.

LASERS
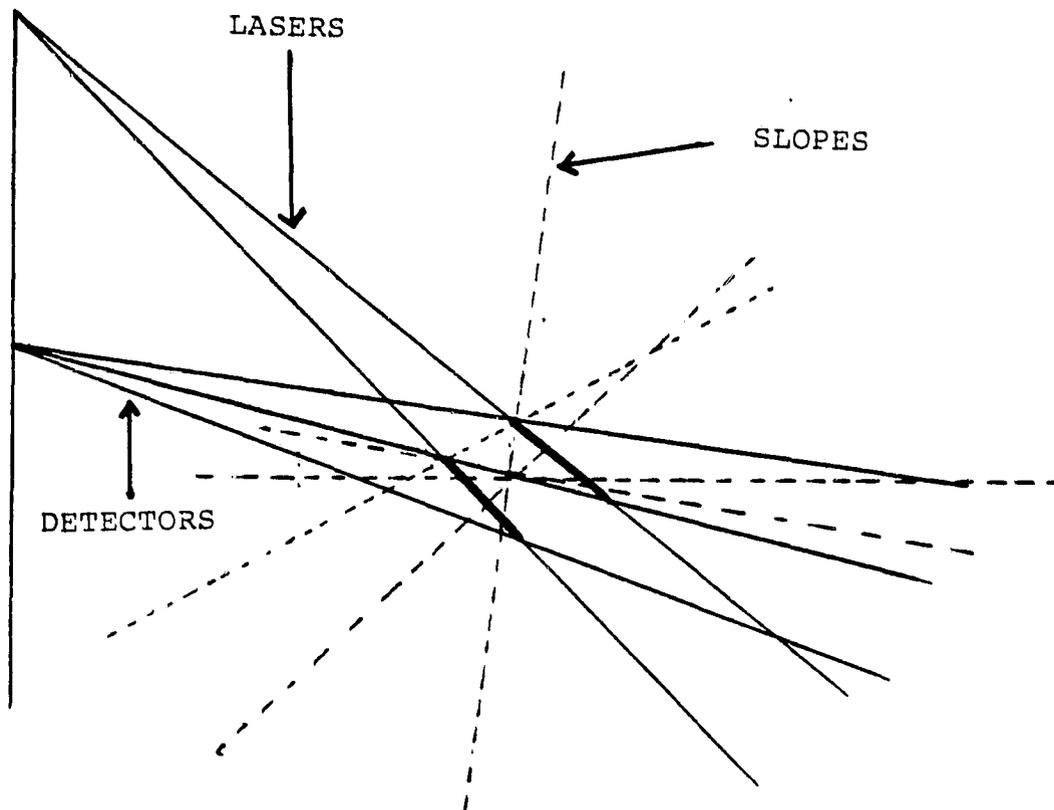
SLOPES

DETECTORS

FIGURE 7

POSSIBLE SLOPES THRU TWO LASER-DETECTOR
SEGMENTS OF INTERSECTION

UPPER BOUND ON
POSSIBLE SLOPES

LOWER BOUND ON
POSSIBLE SLOPES
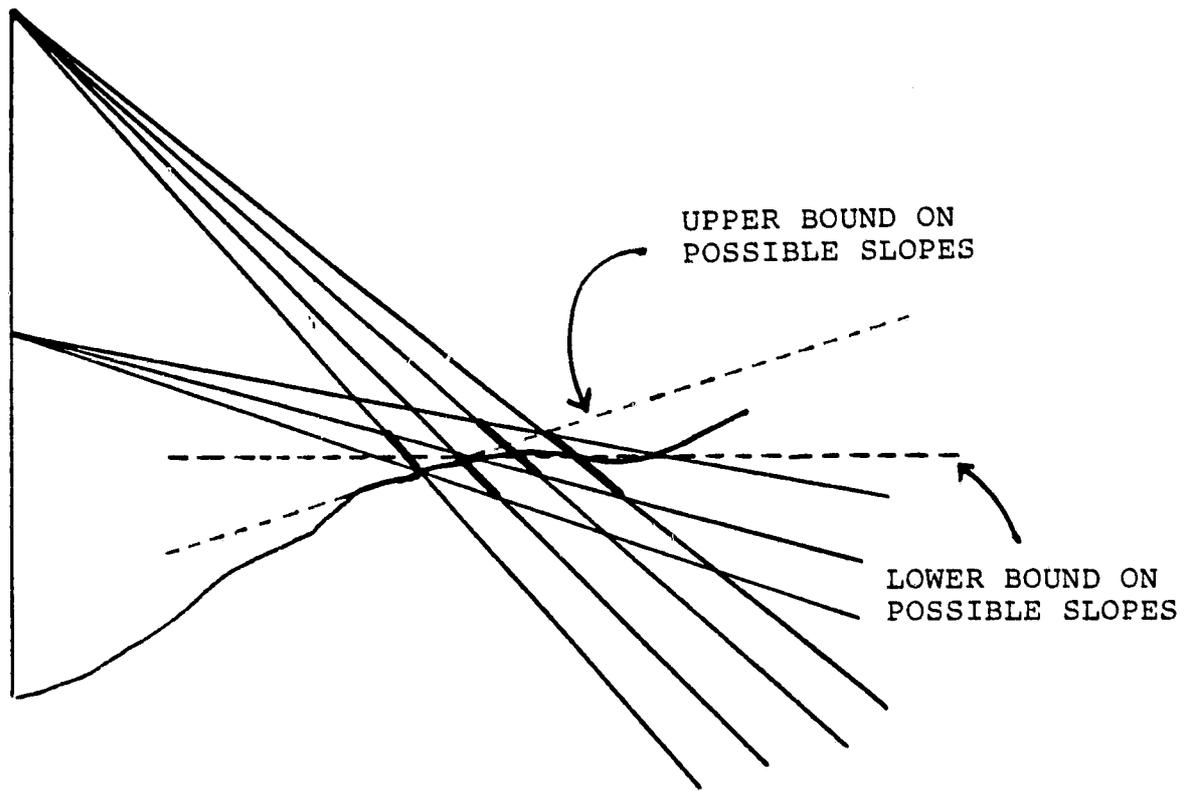
FIGURE 8

UPPER AND LOWER BOUNDS ON A POSSIBLE TERRAIN

The procedure for calculating the upper bounds on the magnitude of the slopes using the data contained in the relative array is shown in Figure 9 and the steps are as follows:

1. Determine the lower endpoint of the line segment for the laser-detector intersection before the first jump (points labelled A on Figure 9).

2. Determine the higher endpoint of the line segment for the laser-detector intersection after the last jump (points labelled B on Figure 9).

3. For negative features select the same laser-detector intersections, but use the opposite endpoints.

4. Compute the slope of the straight line passing through these two points.

The difficulty in the procedure lies in determining where the jump starts and ends. The start of the first jump is found by scanning through the relative array for one azimuth, starting with the lowest laser firing, until a nonzero value is found. This is the start of the first jump. Scanning through the relative array continues until a possible height differential of 0.25 meters is found between the line segment of intersection before the jump and the any laser firing above it. If a possible height differential of 0.25 meters is

- 27 -

POINT (b)

0.25 m

UPPER BOUND
ON SLOPE

POINT (a)

POINT (a)

UPPER BOUND
ON SLOPE

0.25 m

POINT (b)
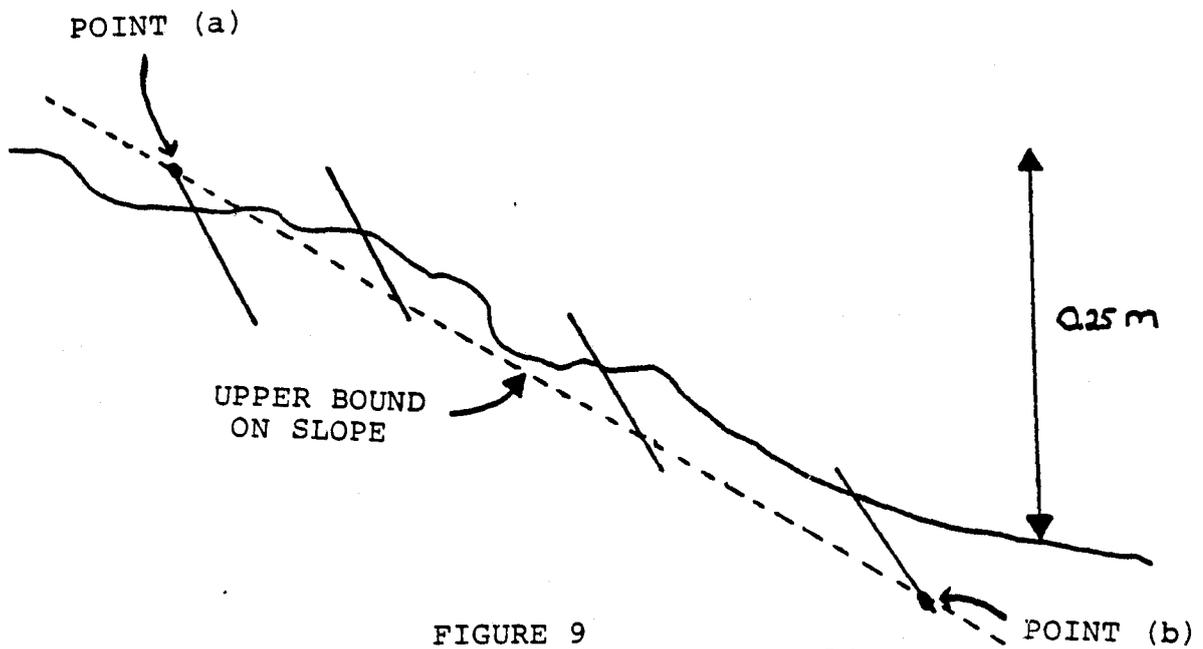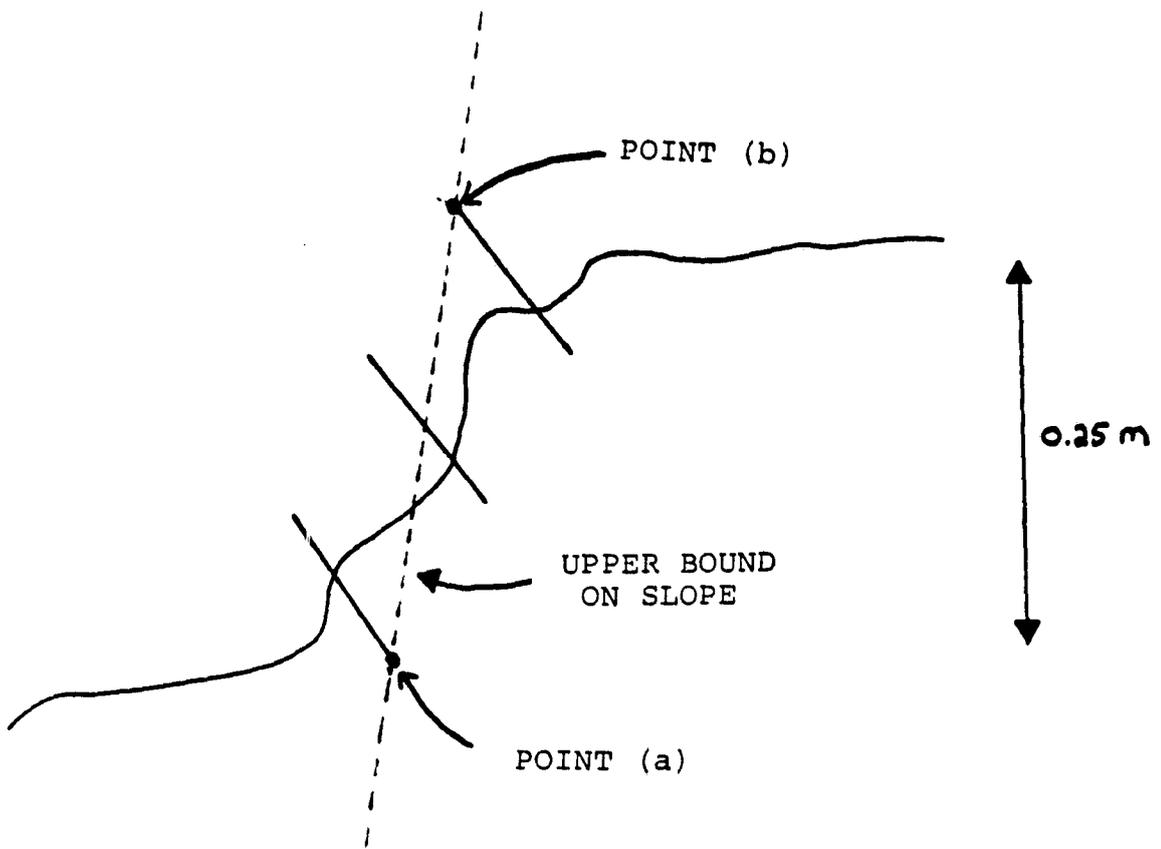
FIGURE 9
EXAMPLES OF CALCULATING UPPER BOUNDS

found then the maximum slope is calculated using the procedure outlined above. The start of the next jump is found by searching the rest of the relative array on the azimuth for a value that differs from the last value of the last jump. The process continues until the highest laser on the azimuth is checked. Examples of the calculation of the upper bounds on slopes are given in Figure 9.

The lower bounds on the slopes are calculated in an analogous fashion:

1. Determine the higher endpoint of the line segment for the laser-detector intersection before the first jump.

2. Determine the lower endpoint of the line segment for the laser-detector intersection after the last jump.

3. For negative features select the same laser-detector intersections, but use the opposite endpoints.

4. Compute the slope of the straight line passing through these two points.

The beginnings and the ends of the jumps are calculated exactly as they were for the upper bounds except that the height differential must be at least 0.25 meters. This is to insure that any slope calculated rises above 0.25 meters since any rise less than this is not a hazard regardless of how steep the slope is.

3.5 Analysis of the Data

The advanced terrain modeler examines each azimuth of the relative array one at a time to determine which azimuths are safe and which ones contain hazards. Any terrain that can not be classified either as safe or hazardous is avoided once the rover gets closer than one meter. This is to give the rover enough distance to avoid the possible hazard.

The initial test for a safe azimuth is whether there are enough consecutive missing returns to imply the presence of a hole or trench wider than 0.25 meters across. If the missing returns start at a distance of less than one meter from the rover then the azimuth is immediately flagged as hazardous and the next azimuth is examined. If the string of missing returns starts more than a meter from the rover and does not include the final laser shot on the azimuth, then the azimuth is marked as hazardous and the range to the first missing return is stored Instead of proceeding to the next azimuth the modeler continues in case any closer hazards are present.

If the final laser pulse is included in the string of missing returns the azimuth is not flagged as hazardous unless the string starts at a range of less than one meter from the rover. If the last laser pulse is in the string of missing returns, then the terrain on the azimuth probably

rises outside the range of the overall detector field as shown in Figure 10. This normally does not constitute a hazard unless the slope is too steep so the rover is allowed to approach closer to get a better view of the situation.

After checking for missing returns the modeler determines the roll of the rover in the azimuth being scanned by using the current pitch and roll of the rover. If the magnitude of the roll computed is greater than 30 degrees then the azimuth is flagged as unsafe and the next azimuth is examined. If the roll computed is greater than 20 degrees then the modeler checks for any positive steps in a crosspath sense. This entails checking the azimuth directly to the left of the current azimuth to determine if any values are greater than the corresponding values on the current azimuth. If any are found then the azimuth is flagged as hazardous and the next azimuth is examined. If the roll computed is less than minus 20 degrees then the modeler checks for any negative crosspath steps, flagging any found as hazardous before continuing with the next azimuth. Then, if the magnitude of the roll in the current azimuth is less than 20 degrees, any crosspath steps greater than 0.25 meters in magnitude are flagged as hazards. If no hazardous crosspath terrain features are found then all the upper bounds on the inpath slope are calculated as described in Section 3.4. If no upper bounds on the magnitude of the slope in the azimuth

FIGURE 10   TERRAIN RISING OUTSIDE OF SENSOR FIELD
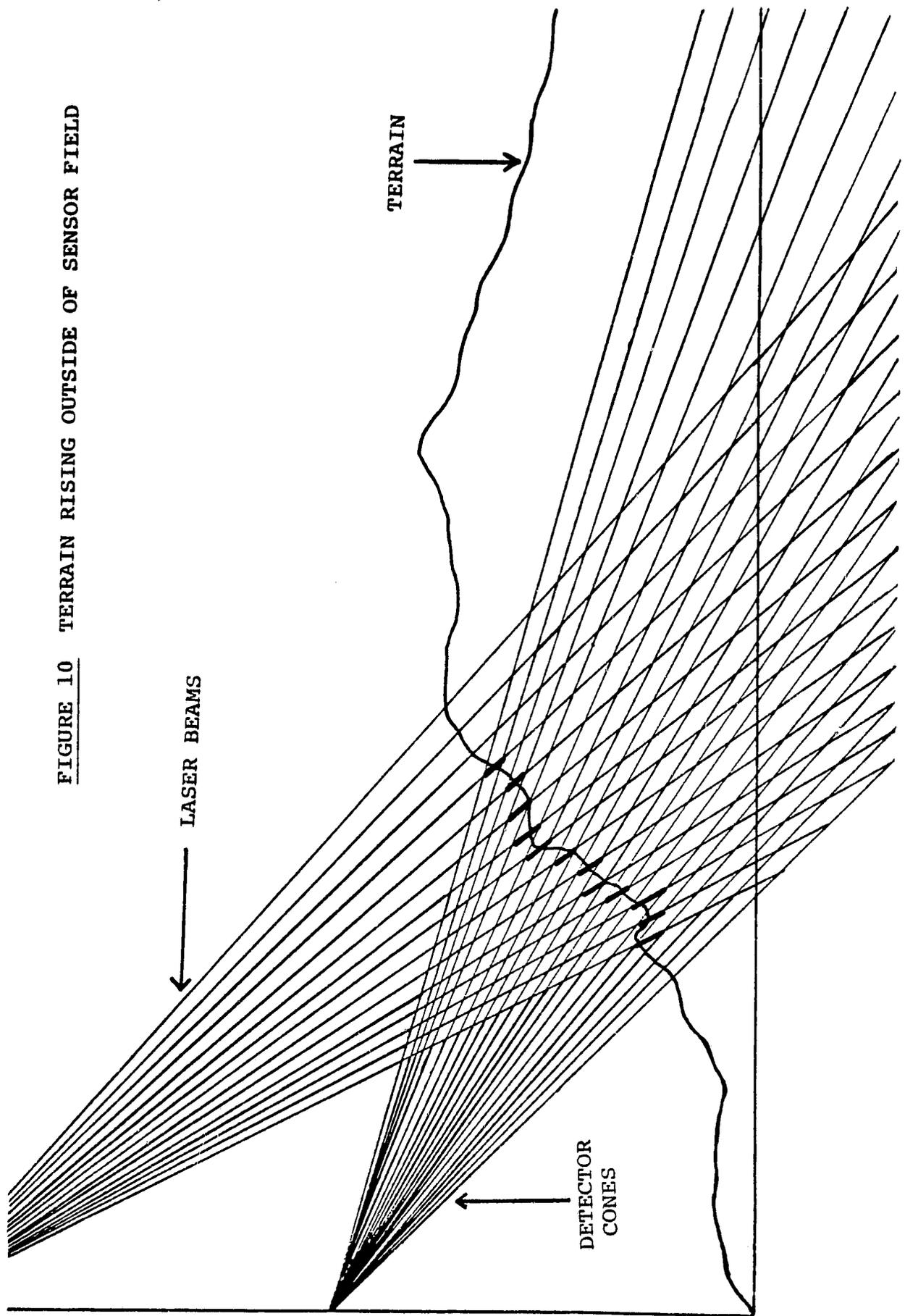
have been found, then the slope is computed through the base of the rover and the upper point on the farthest line segment of laser-detector cone intersection from the rover. This slope is used for the upper bound on the slope of the terrain in the azimuth. Then the pitch of the rover in the azimuth being scanned is computed using the pitch and roll of the rover and the upper bounds on the slopes are adjusted accordingly. If none of the upper bounds are greater than 30 degrees and no previous hazards were found, then the azimuth is flagged as being safe and the next azimuth is examined.

If any of the upper bounds were greater than 30 degrees then the lower bounds on the slope of the azimuth are calculated and adjusted for the pitch of the azimuth. If any of the lower bounds on the slope are greater than 30 degrees then the azimuth is flagged as hazardous. If none of the lower bounds on the slopes are greater than 30 degrees then it cannot be determined whether or not the azimuth is hazardous at this point. Since there is no immediate hazard on the azimuth, the modeler classifies the azimuth as safe and waits until it is closer to the possible hazard before determining whether the azimuth is actually safe. See Figure 11 for a flow diagram of this algorithm.
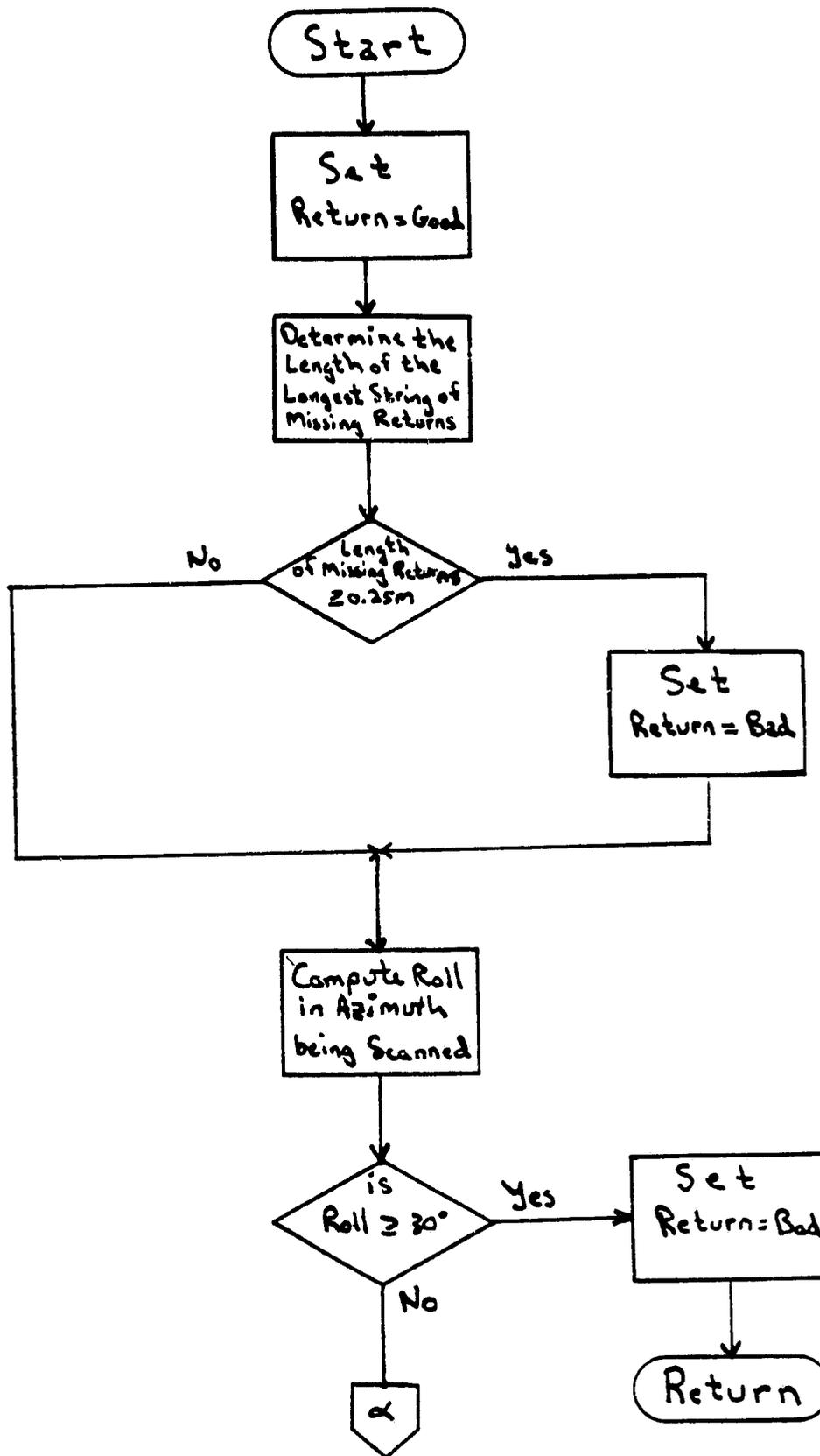
FIGURE 11

FLOW DIAGRAM FOR TERRAIN MODELER
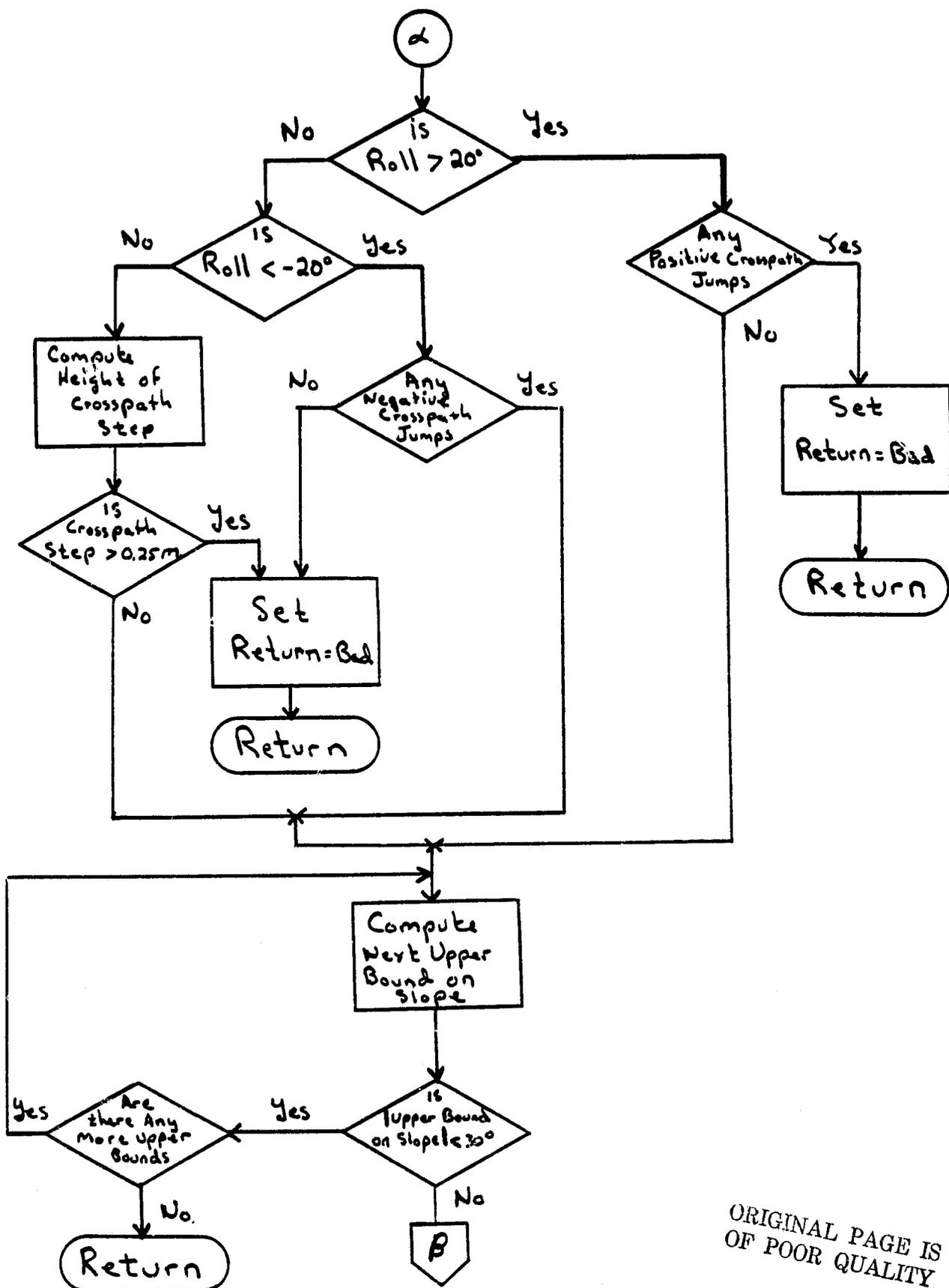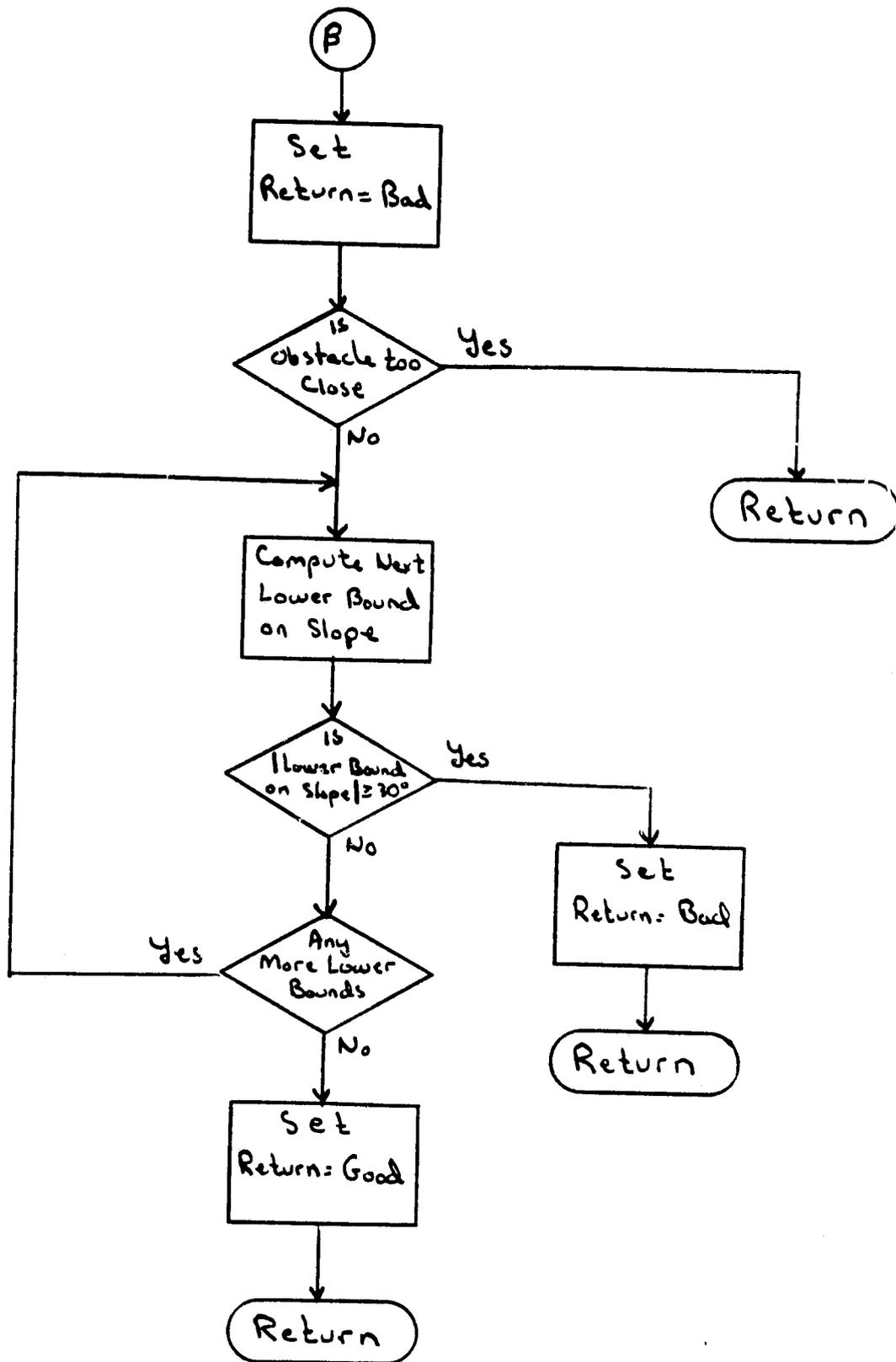
3.6   Comparison of Advanced Modeler with Troiani Modeler

The Troiani terrain modeler did not calculate a maximum slope for any azimuth where there were no jumps larger than 0.25 meters in the relative array. This led to two difficulties, the first stemming from the fact that the rover can not climb a full 0.25 meters step on a slope that is greater than 20 degrees. This is because the effective slope of the climb is greater than the 30 degree climbing ability of the rover. The Troiani terrain modeler solved this problem by signaling any positive steps as hazards when the rover had a pitch greater than 20 degrees. Likewise any negative steps were also flagged as hazards when the pitch of the rover was less than -20 degrees. The second difficulty occurs when the rover is climbing a slope greater then 30 degrees in a manner such that the rover has a pitch less than 30 degrees. If the slope is relatively planar, when the rover looks up the slope, which is greater than 30 degrees, it will appear as level ground in the rover frame of view and will not be flagged as a hazard by the Troiani modeler.

To overcome these problems the advanced terrain modeler calculates at least one maximum slope in every azimuth. This also eliminates the need to search for obstacles when the absolute value of the pitch of the rover is greater than 20 degrees, a procedure that classified many safe terrain features as hazards even though there was

enough information in the array of relative values to classify them as safe.

A second problem in the Troiani terrain modeler resulted from the laser-detector configuration. The detector cone size is kept small in order to keep the range of uncertainty for each laser-detector intersection small but since each detector cone is adjacent to both the cone above and below it, the total sensor field is too small. In fact the rover can look at a steep, but climbable slope and the far end will be outside the overall sensor field so the last laser pulses in the azimuth are not seen by any detectors as depicted in Figure 10. The Troiani modeler flagged this as a hazard but the advanced terrain modeler will not flag a consecutive string of missing returns that includes the last laser pulse as a hazard unless the string extends down the slope far enough to make avoidance necessary. Then, as the rover moves onto the slope and closer to the missing returns a more accurate determination can be made.

Likewise, Figure 12 shows how the first laser pulse will not be seen by any detectors for moderate downgrades. Since these returns are so close to the rover, they should not be ignored as in the case with the missing returns at the end of the azimuth. This situation is improved ·significantly by adjusting the laser pointing angles so that the first laser pulse intersects a detector

FIGURE 12   TERRAIN DROPPING BELOW SENSOR FIELD

LASER BEAMS

DETECTOR
CONES
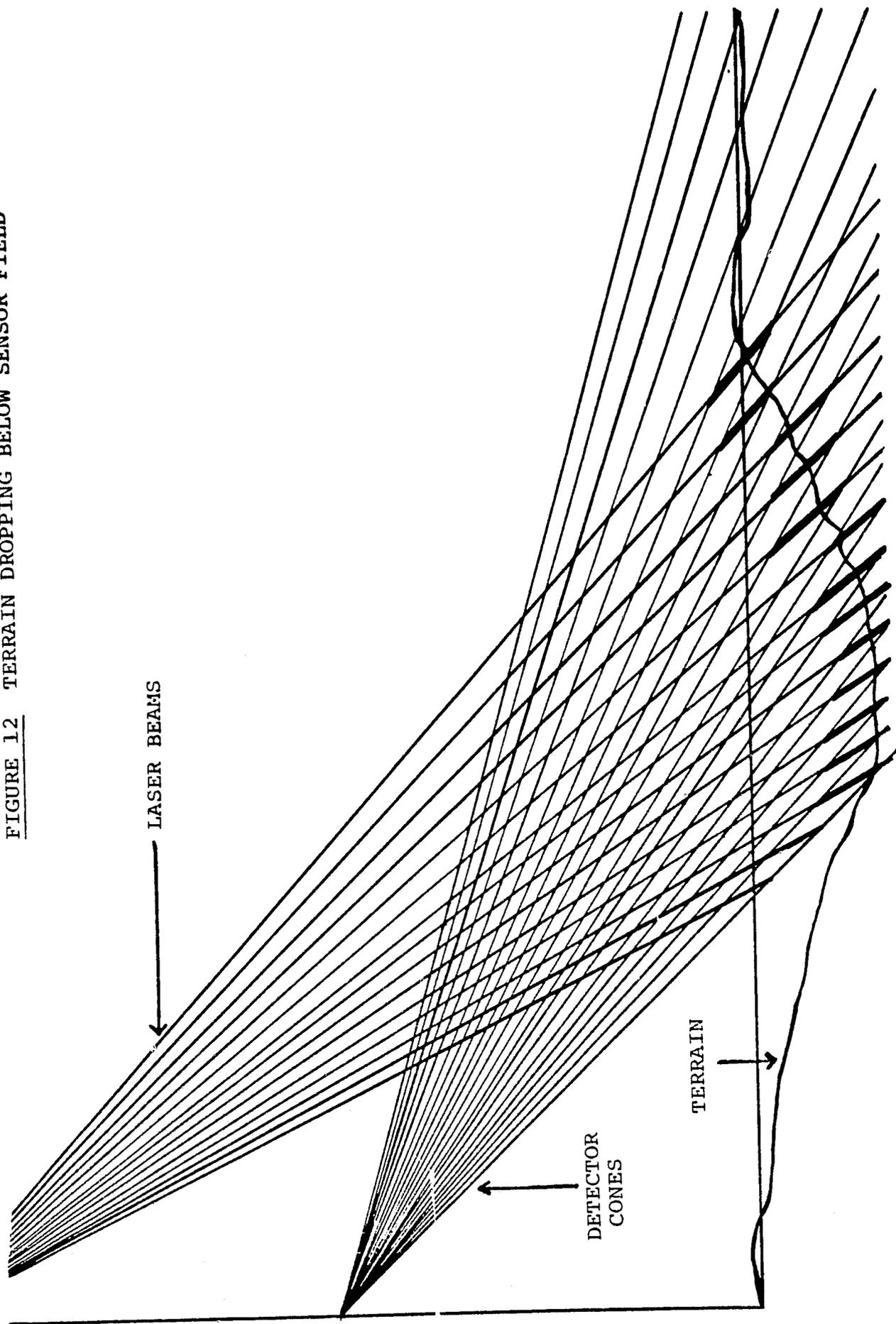
TERRAIN

field other than the first one at level ground. This will leave some "unused" detector cones below level ground for the first laser pulses in case the terrain does slope downward. A multilaser/multidetector system with the laser angles aimed in this manner is shown in Figure 13.

A final problem that none of the previous terrain modelers considered is the problem of crosspath hazards, or hazards that would cause the rover to have a roll greater than 30 degrees. The data from the elevation scanning laser/multidetector system is inherently difficult to analyze accurately in a crosspath sense. Figure 14 depicts the rover, some points where the terrain is being sampled, and the projection of the rover along one of the azimuths. Notice that an approximation to the crosspath difference in the height of the two azimuths can be made by comparing the heights of the same laser pulse on two azimuths. Since the terrain in front of the rover is sampled in a semicircle pattern, this approximation will only hold for azimuths adjacent to each other; thus this approximation can not be used to calculate the slope of the crosspath as was done in the inpath. Instead, the advanced terrrain modeler uses this approximation to search for any hazardous steps in the crosspath sense. Then an approximation to the slope of the rover in that azimuth is made by assuming that the terrain is planar and projecting the pitch and roll of the rover into the azimuth being examined.

FIGURE 13 LASER–DETECTOR CONFIGURATION WITH FIRST LASER PULSE AIMED AT CENTER OF THIRD DETECTOR CONE ON LEVEL GROUND

LASER BEAMS

DETECTOR CONES
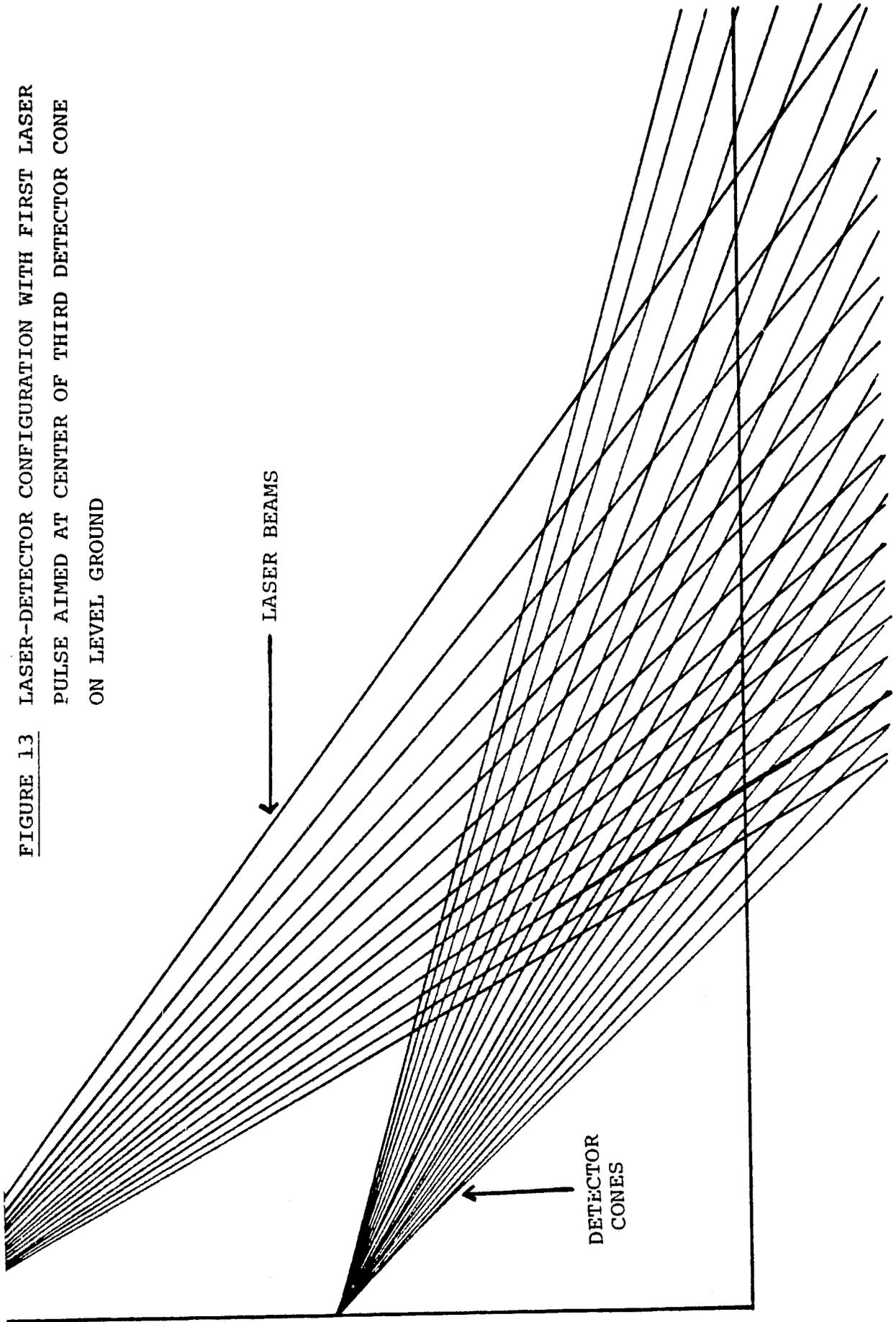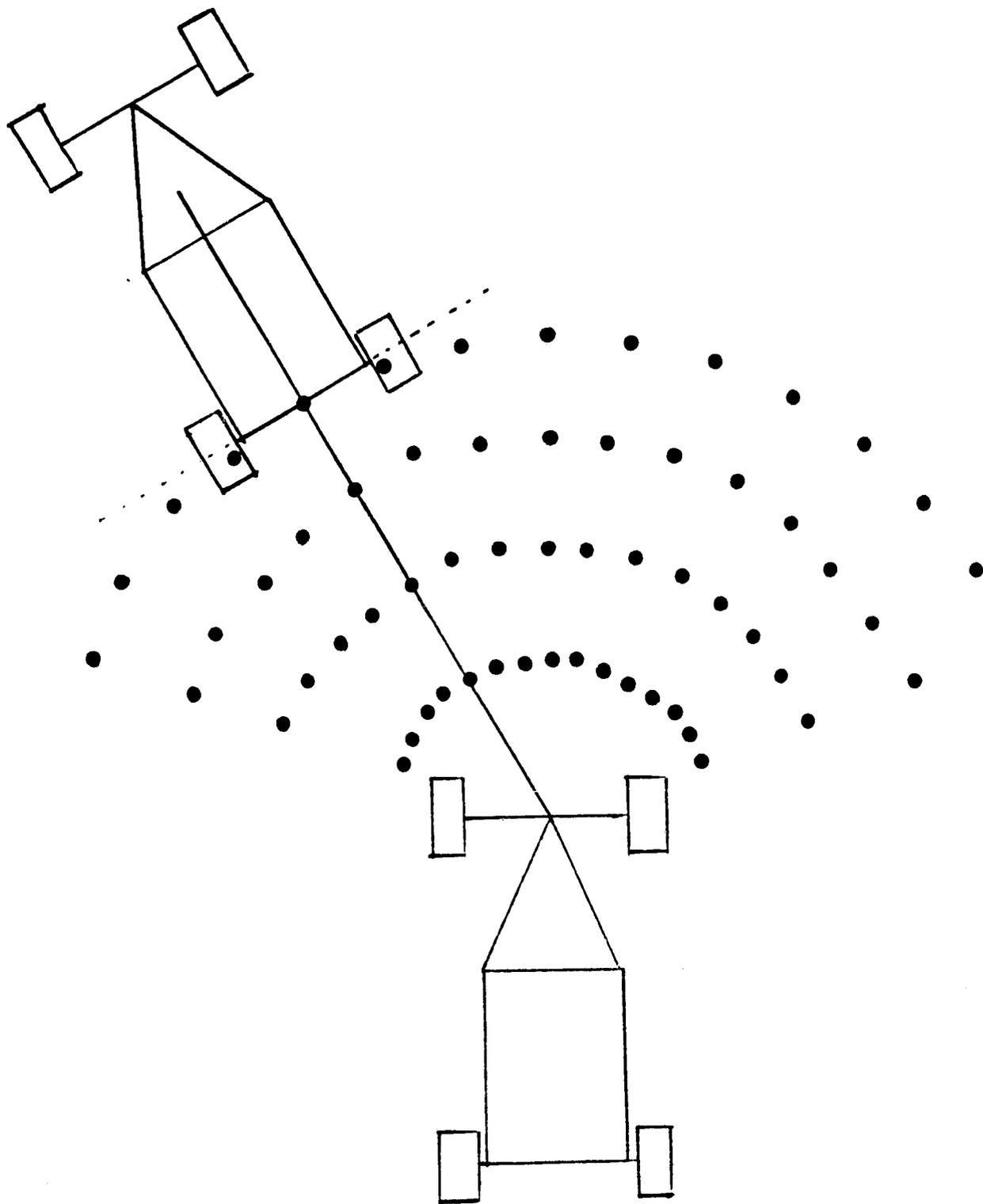
FIGURE 14

PROJECTION OF ROVER IN -30 DEGREE AZIMUTH

Finally a test is made that is similar to the test for an obstacle on a 20 degree slope made by the Troiani modeler. If a crosspath slope with a magnitude greater than 20 degree is detected, then any crosspath step in the same direction as the slope is flagged as hazardous. This is because a step with a height of 0.25 meters on a slope greater than 20 degrees could cause an effective slope greater than 30 degrees.

These improvements result in an advanced terrain modeler that rejects more hazardous terrain configurations and accepts more safe terrains than previous modelers. Results from tests of the advanced terrain modeler are shown and explained in the next section.

4. SIMULATED PERFORMANCE AND RESULTS

Once the terrain modeler routine was written an optimal laser-detector configuration was chosen and the entire system was tested. Tests were chosen to compare the advanced modeler with the Troiani modeler and to determine how well the advanced modeler performed when problems such as crosspaths and complex terrains were encountered.

4.1 Evaluation of Detector Cone Size for a 32 Laser/40 Detector System

This section will explain the procedure for choosing an optimal detector cone size for the rover under development at Rensselaer, then present the results of a search for the optimum cone size. The rover at Rensselaer has a 32 laser/40 detector system and uses the advanced terrain modeler explained earlier in this report. The use of a different terrain modeler or a different laser/detector system could affect the optimum cone size, but the method for determining the optimum cone size would be the same.

In choosing a detector cone size, the objective is to keep the size of the detector cone as small as possible while still maintaining a large enough overall sensor field to see all the possible terrains. If the overall sensor field sees level ground from one meter to two meters in

front of the rover then the rover, which is traveling at
0.2 meters/second and taking one scan every second, will be
able to make about five scans of each possible hazard. When
looking at an uphill slope the rover will see less of the
terrain and thus will have fewer scans of each feature.
Likewise, while looking at a downhill slope the rover will
see more of the terrain and thus will have more scans of
each feature. These phenomena are shown in Figure 15. It
is of no use to the rover to see at ranges closer than
0.5 meters, since the rover is only designed for forward
motion and it could not avoid obstacles that close without
reversing itself. Likewise, the line segments of
intersection become too large at ranges greater than 3 or
4 meters severely limiting the usefulness of the data at
these ranges. Also the maximum range of the laser beam
starts to become a problem at these distances.

In setting up the laser/detector configuration,
missing returns at the far end of the detector field, as
shown in Table 6, pose no problem because the advanced
terrain modeler ignores them. On the other hand, missing
returns close to the rover are a serious problem so the
lasers and detectors must be set up to avoid them for safe
terrains. To solve this problem the first laser should hit
level ground at the center of one of the detector cones
numbered between four and seven, to leave enough detector
cones below the first laser to see a 20 to 25 degree
downslope.

FIGURE 15   LASER-DETECTOR FIELD OF VIEW SEES MORE
OF A DOWNHILL SLOPE THAN AN UPHILL SLOPE
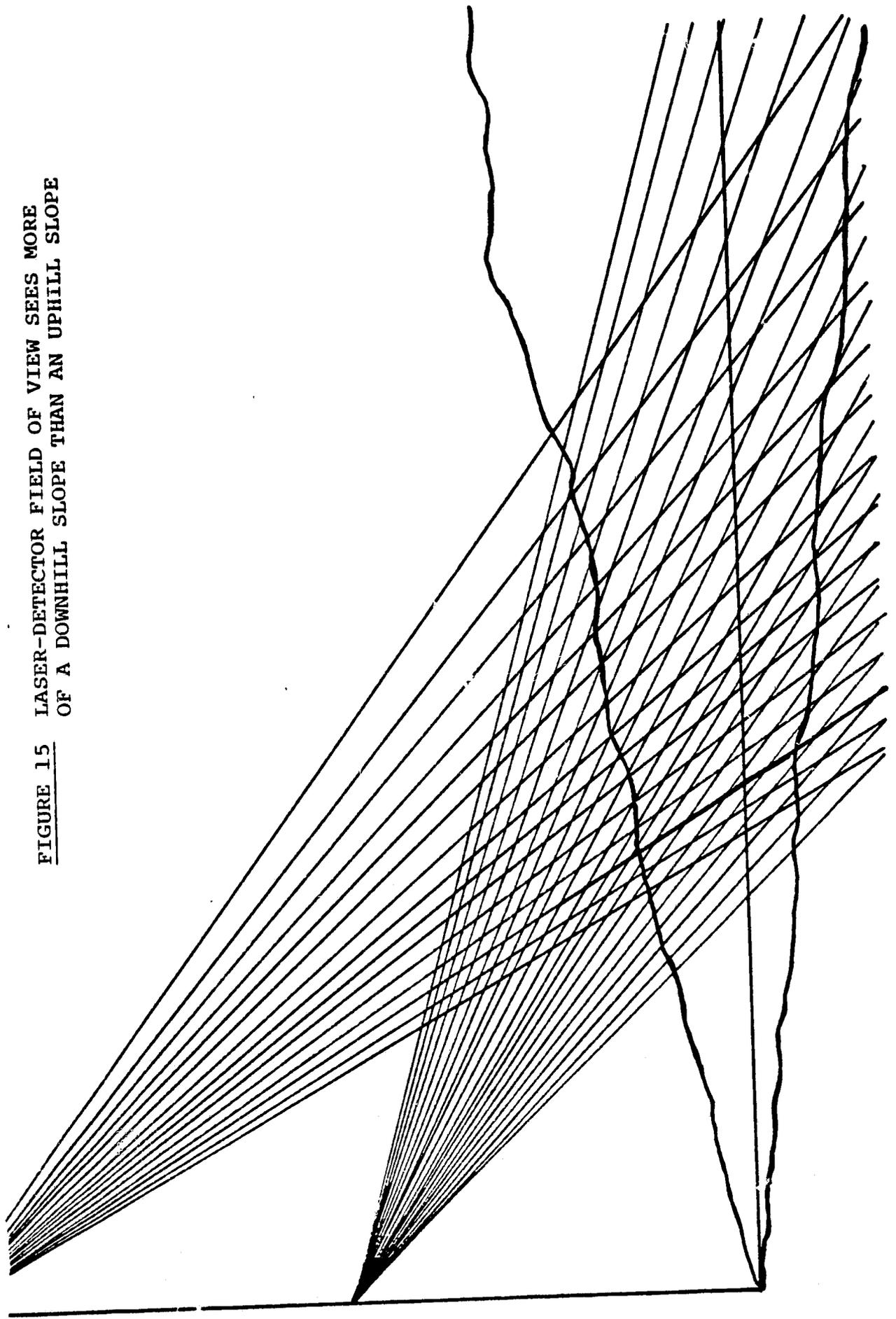
AZIMUTH	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LASER															
32	***	***	***	***	***	***	***	***	***	***	***	***	***	***	***
31	***	***	***	***	***	***	***	***	***	***	***	***	***	***	***
30	4	***	***	***	***	***	***	***	***	***	***	***	***	***	4
29	4	***	***	***	***	***	***	***	***	***	***	***	***	***	4
28	4	5	***	***	***	***	***	***	***	***	***	***	***	5	4
27	4	5	7	***	***	***	***	***	***	***	***	***	7	5	4
26	4	5	7	8	***	***	***	***	***	***	***	8	7	5	4
25	3	5	7	8	9	***	***	***	***	***	9	8	7	5	3
24	3	5	7	8	9	10	10	***	10	10	9	8	7	5	3
23	3	5	7	8	9	10	10	10	10	10	9	8	7	5	3
22	3	5	6	8	9	9	10	10	10	9	9	8	6	5	3
21	3	5	6	8	8	9	10	10	10	9	8	8	6	5	3
20	3	5	6	7	8	9	10	10	10	9	8	7	6	5	3
19	3	5	6	7	8	9	9	10	9	9	8	7	6	5	3
18	3	5	6	7	8	9	9	9	9	9	8	7	6	5	3
17	3	4	6	7	8	9	9	9	9	9	8	7	6	4	3
16	3	4	6	7	8	8	9	9	9	8	8	7	6	4	3
15	3	4	6	7	8	8	9	9	9	8	8	7	6	4	3
14	3	4	6	6	7	8	9	9	8	8	7	6	6	4	3
13	3	4	5	6	7	8	9	9	9	8	7	6	5	4	3
12	3	4	5	6	7	8	8	8	8	8	7	6	5	4	3
11	3	4	5	6	7	8	8	8	8	8	7	6	5	4	3
10	3	4	5	6	7	7	8	8	8	7	7	6	5	4	3
9	3	4	5	6	7	7	8	8	8	7	7	6	5	4	3
8	2	4	5	6	6	7	7	7	7	7	6	6	5	4	2
7	2	3	5	6	6	7	7	7	7	7	6	6	5	3	2
6	2	4	4	5	6	7	7	7	7	7	6	5	4	4	2
5	2	3	4	5	6	6	7	7	7	6	6	5	4	3	2
4	2	3	4	5	6	6	6	7	6	6	6	5	4	3	2
3	2	3	4	5	6	6	7	7	7	6	6	5	4	3	2
2	2	3	4	5	5	6	6	6	6	6	5	5	4	3	2
1	2	3	4	5	5	6	6	6	6	6	5	5	4	3	2

TABLE 6

Illustration of Missing Returns at the Far End

the Relative Array

The actual specifics for the laser/detector configurations are computed as shown in Figure 16. The range for the intersection of the first detector cone and level ground is chosen, usually about 0.7 meters. From this, the pointing angle of the first detector cone can be computed using the formula:

$$Angle = TAN^{-1}\left[\frac{Range}{height\ of\ detector}\right]$$

but since the height of the detector array is one meter this formula reduces to the Arc Tangent of the range. For a range of 0.7 meters the first detector angle is about 25 degrees.

Next the detector cone that the first laser will intersect at level ground is chosen. For a 0.75 degree cone size the seventh detector cone was chosen. The pointing angle of the start of the detector cone that sees the first laser pulse is the pointing angle of the center of the first cone plus the cone size times the number of the detector to see the first laser pulse minus one and a half. The one half is subtracted because the angle of the leading edge of the cone is desired, not the center of the cone. For the example being used this angle is:

$$35.0 + 0.75 * (7 - 1 - 1/2) = 39.1\ deg.$$

FIGURE 16   SPECIFICATIONS FOR LASER–DETECTOR
            CONFIGURATION

HEIGHT OF DETECTOR FIELD
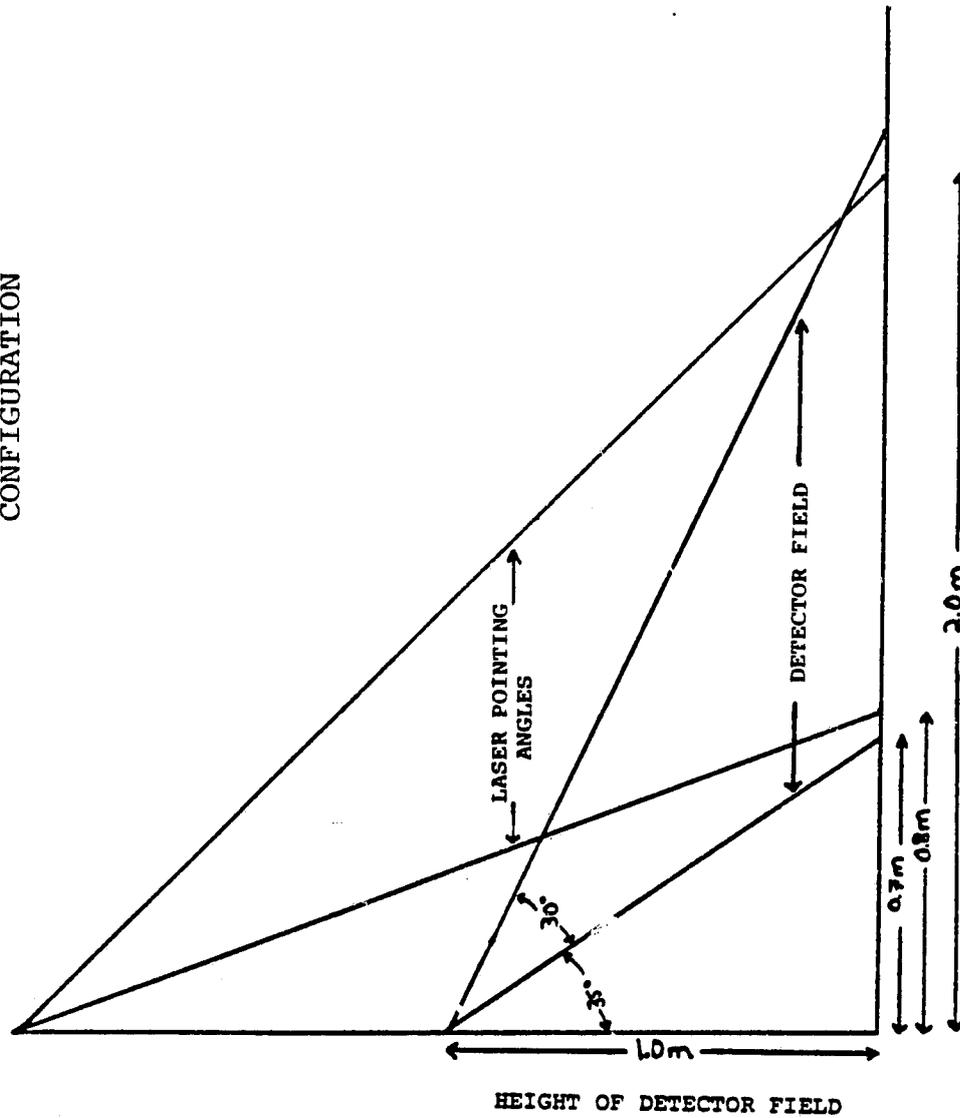
Thus the range of the detector cone that sees the first laser pulse on level ground is the tangent of 39.1 degrees, which is 0.8 meters.

The location of the end of the last detector cone to see a laser pulse at level ground is found by taking the tangent of the sum of the start of the first detector cone to see a laser pulse and the cone size times the number of lasers. For the above example this location is:

$$TAN(39.1 + 0.75 * 32) = 2.0 \text{ meters}$$

The results of these calculations are given for four different cone sizes in Table 7.

size of sensor field (degrees)	cone size (degrees)	range to detector cone (meters)	# of first detector field to see laser	closest laser hit (meters)	farthest laser hit (meters)
30	0.75	0.7	7	0.8	2.0
35	0.875	0.7	5	0.8	2.3
40	1.0	0.7	4	0.8	2.7
60	1.5	0.5	2	0.5	3.8

TABLE 7

Results of Cone Size Calculations

The results of running these laser/detector configurations on the simulator are shown in Table 8. The rover was positioned in front of a hill with a slope between

25 and -25 degrees. If the rover went in a straight line from the start to the target as shown in Figure 17, then in Table 8 the run is labeled as "good". If the rover deviated from a straight line, but still reached the target as shown in Figure 18, then the run is listed as "deviated". If the rover deviated so much that it did not reach the target then the run is listed as "almost" and if the rover did not even try to negotiate the slope the run is listed as "no good".

Slope	Cone Size			
(degrees)	(degrees)			
	0.75	0.875	1.0	1.5
30	No Good	No Good	No Good	No Good
25	Almost	Good	Almost	No Good
20	Good	Good	Deviated	No Good
15	Good	Good	Good	Good
-15	Good	Deviated	Good	No Good
-20	Deviated	No Good	No Good	No Good
-25	Almost	No Good	No Good	No Good
-30	No Good	No Good	No Good	No Good

TABLE 8

Results of Computer Simulations on Varying Cone Sizes

TARGET

CENTER OF
← FRONT AXLE

REAR WHEELS

FIGURE 17

ROVER ADVANCING TO TARGET IN

DIRECT COURSE
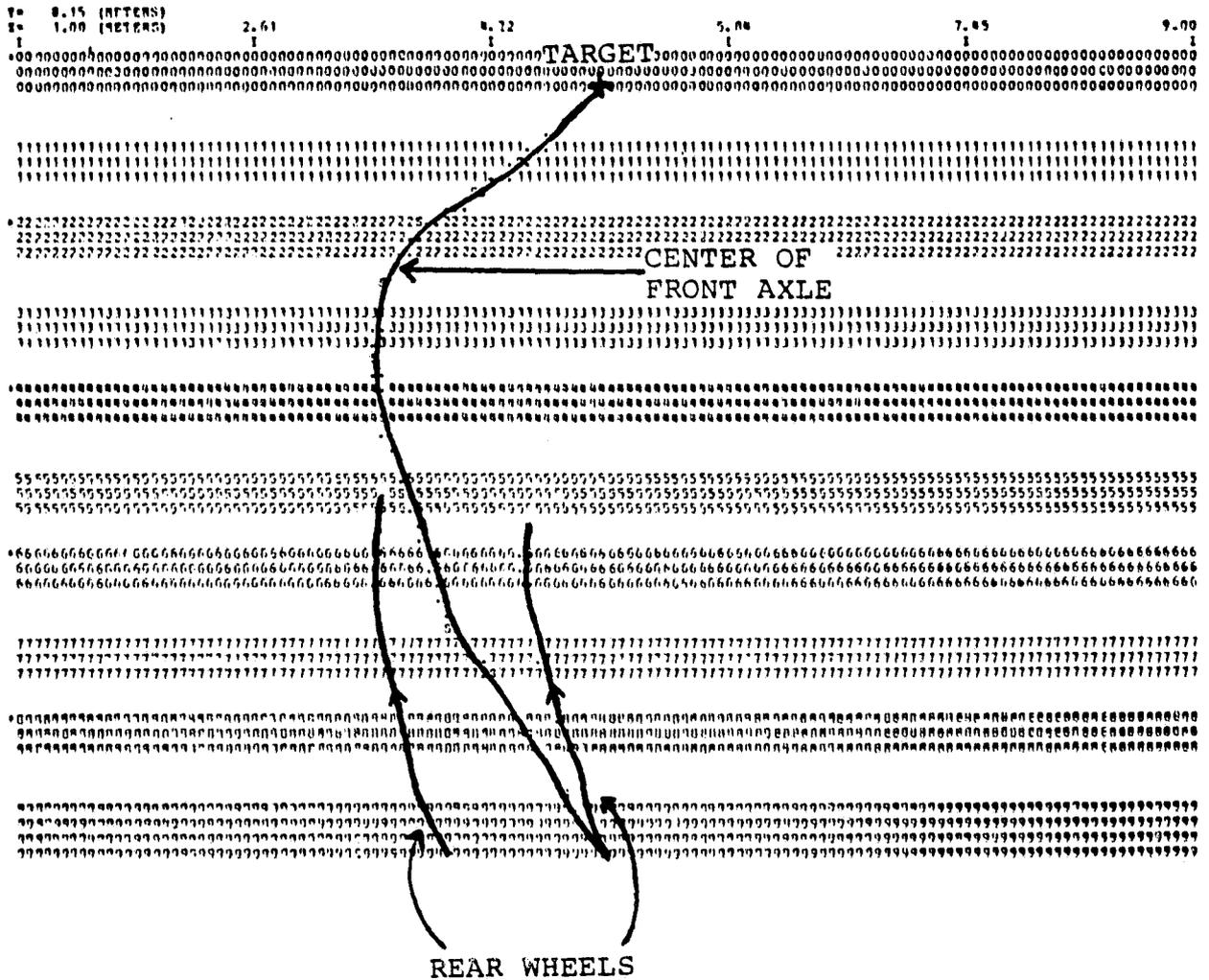
FIGURE 18

ROVER ADVANCING TO TARGET IN

DEVIATED COURSE

Note in Table 8 that as the cone size got smaller the rover generally performed better. This is because the smaller cone size results in a smaller line segment of intersections and thus a better estimate of the slope as is shown in Table 9 which lists the estimates of the slopes made for each cone size. On slopes where there is more than one entry, the other entries are estimates of the slope as the rover moves onto the slope in a straight line and the pitch in parentheses is the pitch of the rover as it looks at the slope.

In Table 9 there are several values for the upper bound on the slope enclosed in parentheses that seem much larger than the rest of the estimates of the slopes. These slopes are generated because at the end of the azimuth the advanced terrain modeler drops the requirement that the upper bound have a possible 0.25 meter jump. This insures that a last upper bound on the slope is calculated on the azimuth, but sometimes results in an overly large slope because the data points a used to calculate the slope are so close together in height and range.

Cone Size (Degrees)

Slope (Pitch) (degrees)	0.75	0.875	1.0	1.5
25 - 1 (0.0)	25.9, 32.5	27.4, 30.0	27.7, 34.4	-
20 - 1 (0.0)	20.9, 24.6	21.5, 27.0	20.7, 28.2, (78.3) ; 19.9	25.7, 33.5
20 - 2 (4.17)	22.2, 25.1	22.6, 29.7	-	-
15 - 1 (0.0)	17.2, 29.8	17.5, 20.9	17.2, 18.5, (80.3) ; 14.5	19.7, 27.1, (81.9) ; 14.9
15 - 2 (3.1)	16.7, (85.3) ; 14.8	17.5, (48) ; 14.9	18.0, 21.2	19. , 31.0 ; 14.9
15 - 3 (6.1)	16.7, (89.6)	15.3	17.3, (89.8)	18.4, (90.0) ; 15.1
15 - 4 (9.2)	15.8	16.4	17.3, (94.2)	27.9, (94.6)
-15 - 1 (0.0)	-17.4, -23.2, -38.5 ; -14.6	-16.5, -31.5	-19.3, -27.7, -40.6 ; -15.2	-23.1, -53.7 ; -14.4, -14.6
-15 - 2 (-3.1)	-18.1, -24.5, (-76.6) ; -15.0	-	-20.1, -24.6, (-74.0) ; -14.8	-
-20 - 1 (0.0)	-21.4, -35.7	-21.3, -38.0	-23.9, -52.2 ; -19.3, -19.1	-
-25 - 1 (0.0)	-25.2, -43.4	-	-	-

TABLE 9

Estimates of Slopes for Varying Cone Sizes

Also note in Table 9 that the lower bounds on the slopes are much closer to the actual value of the slopes than the upper bounds on the slopes. This is because of the terrain set up used rather than any inherent superiority of the lower bound calculations over the upper bound calculations. The terrain was constructed of long, not too steep slopes. Since the lower bound calculations are not made until the smallest possible jump is greater than 0.25 meters, the points used in the lower bound calculation tend to be more separated and estimate the long slope better than the upper bound calculations that use points which are closer together. If the terrain had instead been short, steep, step-like slopes then the upper bound calculations would have resulted in closer estimates to the actual slopes.

From all this testing the best cone size for the 32 laser pulse/40 detector system to be used at Rensselaer in conjunction with the advanced terrain modeler is 0.75 degrees. This sees from 0.8 meters to 2.0 meters in front of the rover on level terrain providing that the first detector cone is set up to intersect level ground 0.7 meters in front of the rover and the seventh detector cone sees the reflection of the first laser pulse off level terrain.

4.2 Tests Comparing the Advanced Modeler with the Troiani Modeler

The advanced terrain modeler has three functional differences from the terrain modeler developed by Troiani. The advanced modeler performs a crosspath analysis, it ignores missing returns at the far end of the relative array and it computes at least one upper bound on the slope in each azimuth. Since the Troiani modeler did not compute crosspath slopes, no comparison is warranted on this subject. The effect of the advanced modeler ignoring the far missing returns is that the laser/detector configuration can have smaller detector cones with the advanced modeler, allowing a more accurate estimate of the slope of the terrain. The effects of varying detector cone sizes were discussed in the previous section. Since the advanced terrain modeler computes the slope in every azimuth it does not have to check for any steps on azimuths with slopes greater than 20 degrees as the Troiani modeler did.

Figure 19 shows the simulated running of the rover up a 20 degree slope with a 0.1 meter block on it under the advanced terrain modeler. It is seen that the advanced modeler has no problem recognizing that this terrain is safe. Figure 20 shows the rover attempting to negotiate the same terrain under control of the Troiani modeler. After the slope of the rover has attained 20 degrees the rover sees the return shown in Table 10. Since the slope of the

- 57 -
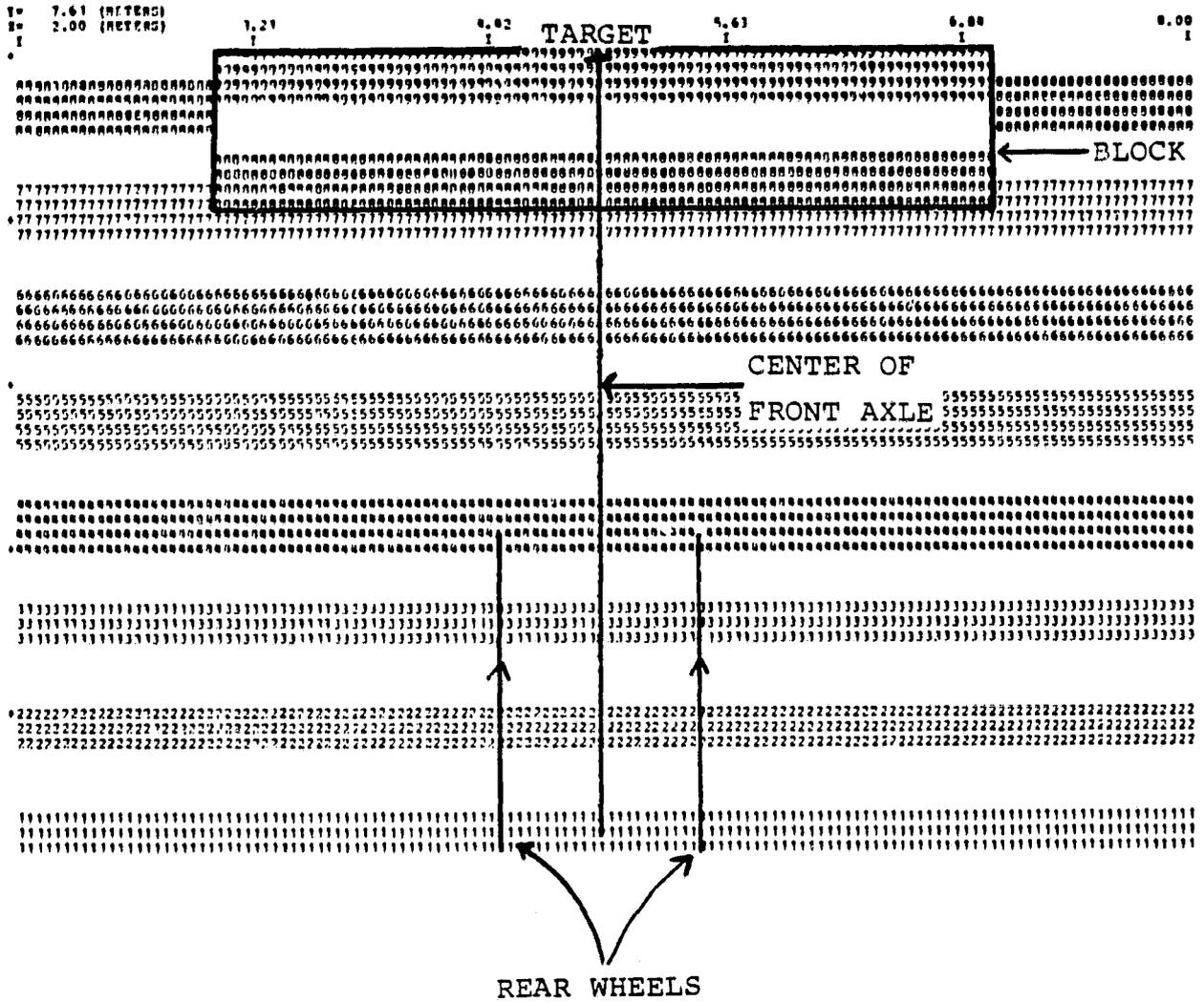
FIGURE 19

ROVER CLIMBING 20 DEGREE SLOPE WITH A SMALL BLOCK

WITH THE ADVANCED TERRAIN MODELER
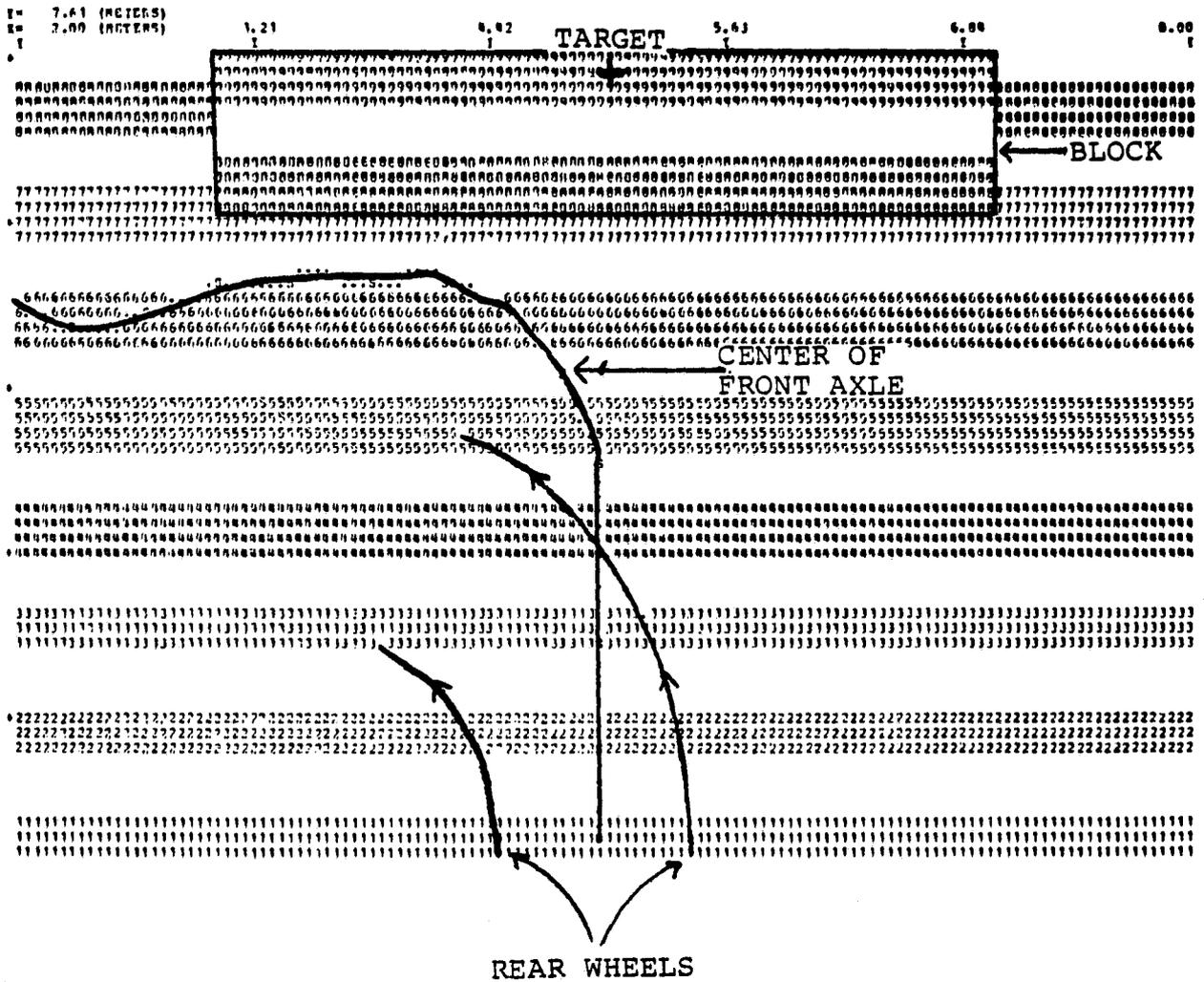
FIGURE 20

ROVER CLIMBING A 20 DEGREE SLOPE WITH A SMALL BLOCK

WITH THE TROIANI MODELER

rover is 20 degrees the Troiani modeler considers this terrain a hazard. The advanced modeler computes that none of the slopes are greater than 25 degrees and thus has no trouble with the slope.

AZIMUTH	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LASER															
32	0	0	2	2	2	2	2	2	2	2	2	2	2	0	0
31	0	0	2	2	2	2	2	2	2	2	2	2	2	0	0
30	0	0	2	2	2	2	2	2	2	2	2	2	2	0	0
29	0	0	2	2	2	2	2	2	2	2	2	2	2	0	0
28	0	0	2	2	2	2	2	2	2	2	2	2	2	0	0
27	0	0	1	2	2	2	2	2	2	2	2	2	1	0	0
26	0	0	1	2	2	2	2	2	2	2	2	2	1	0	0
25	0	0	0	2	2	2	2	2	2	2	2	2	0	0	0
24	0	0	0	2	2	2	2	2	2	2	2	2	0	0	0
23	0	0	0	2	2	2	2	2	2	2	2	2	0	0	0
22	0	0	0	2	2	2	2	2	2	2	2	2	0	0	0
21	0	0	0	1	2	2	2	2	2	2	2	1	0	0	0
20	0	0	0	1	2	2	2	2	2	2	2	1	0	0	0
19	0	0	0	0	2	2	2	2	2	2	2	0	0	0	0
18	0	0	0	2	2	2	2	2	2	2	2	0	0	0	0
17	0	0	0	0	2	2	2	2	2	2	2	0	0	0	0
16	0	0	0	0	1	2	2	2	2	2	1	0	0	0	0
15	0	0	0	0	1	2	2	2	2	2	1	0	0	0	0
14	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
13	0	0	0	0	0	1	2	2	2	1	0	0	0	0	0
12	0	0	0	0	0	1	2	2	2	1	0	0	0	0	0
11	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
10	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 10

Relative Array from Rover on a 20 Degree Slope
Looking at a 0.1 Meter High Block

4.3 Tests of Advanced Modeler on Slopes with Varying Crosspaths

The running of the rover was simulated on several slopes under control of the advanced modeler with crosspath slopes varying from 15 to 30 degrees. Since the terrain that the rover was looking at had the same slope as the rover, the relative array was all zeros as shown in Table 11. When the crosspath slope was less than 30 degrees the rover progressed straight to the target as shown in Figure 21. On slopes greater than or equal to 30 degrees the rover did not go anywhere because of the crosspath slope ahead of it and the possible inpath hazards going up and down the slope.

The same tests were run with a block in the path that showed up in the relative array as shown in Table 12. With a 15 degree crosspath slope the rover progressed straight to the target without any difficulty as shown in Figure 22. When the crosspath slope was increased to 25 degrees the advanced modeler determined that the block was a hazard and avoided the block as shown in Figure 23. Finally when the crosspath was increased to 30 degrees the rover once again stopped before going too far as shown in Figure 24.

AZIMUTH	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LASER															
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0.	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 11

Relative Array from Planar Terrain with same Slope as Rover
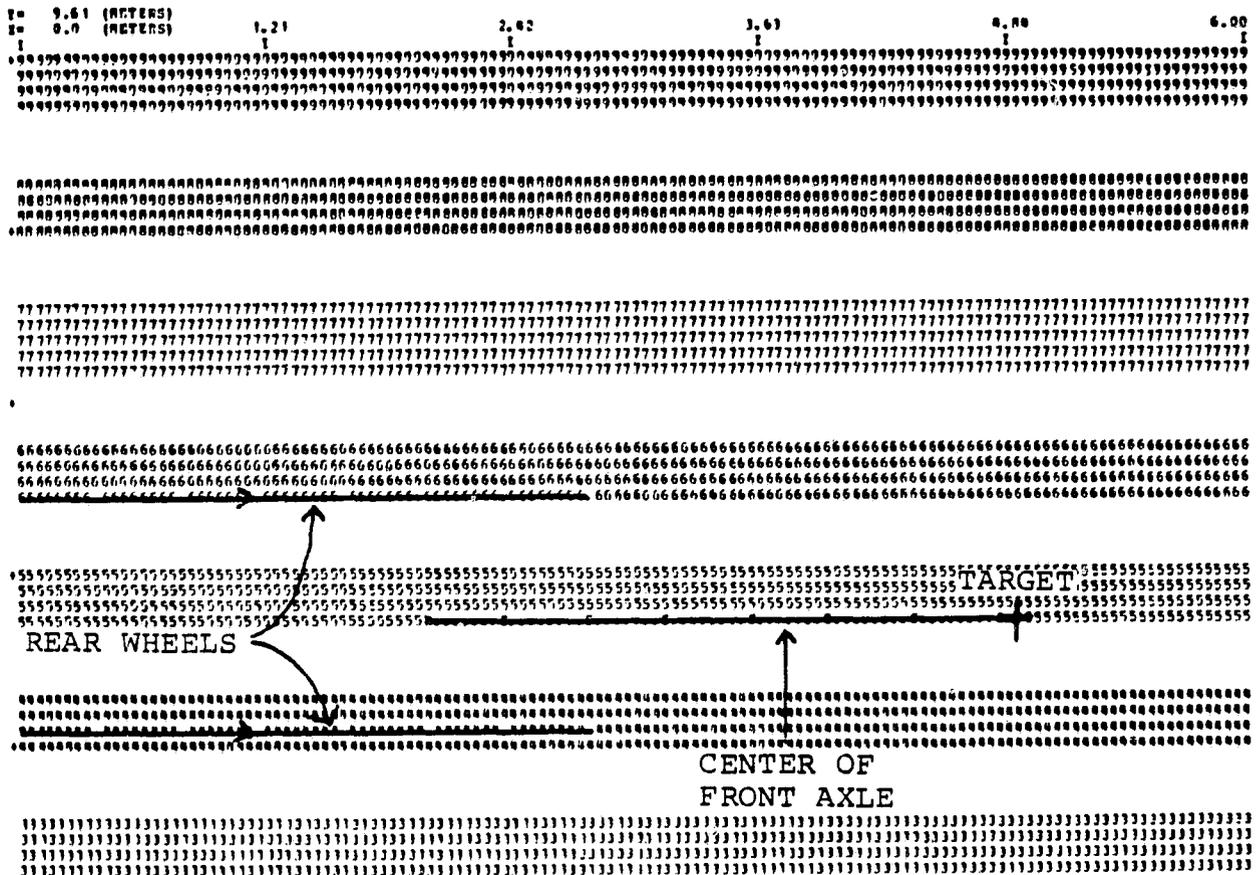
FIGURE 21

ROVER NEGOTIATING CROSSPATH
SLOPE LESS THAN THIRTY DEGREES

AZIMUTH	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LASER															
32	0	0	0	0	0	0	2	2	2	0	0	0	0	0	0
31	0	0	0	0	0	0	2	2	2	0	0	0	0	0	0
30	0	0	0	0	0	0	2	2	2	0	0	0	0	0	0
29	0	0	0	0	***	0	2	2	2	0	0	0	0	0	0
28	0	0	0	0	***	0	2	2	2	0	0	0	0	0	0
27	0	0	0	0	0	2	2	2	2	0	0	0	0	0	0
26	0	0	0	0	0	2	2	2	2	***	0	0	0	0	0
25	0	0	0	0	0	2	2	2	2	1	0	0	0	0	0
24	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
23	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
22	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
21	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
20	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
19	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
18	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
17	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
16	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
15	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
14	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
13	0	0	0	0	0	2	2	2	2	2	0	0	0	0	0
12	0	0	0	0	0	1	2	2	2	1	0	0	0	0	0
11	0	0	0	0	0	0	2	2	2	0	0	0	0	0	0
10	0	0	0	0	0	0	1	2	1	0	0	0	0	0	0
9	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 12

Relative Array from Rover on a Crosspath Slope

with a Block on it
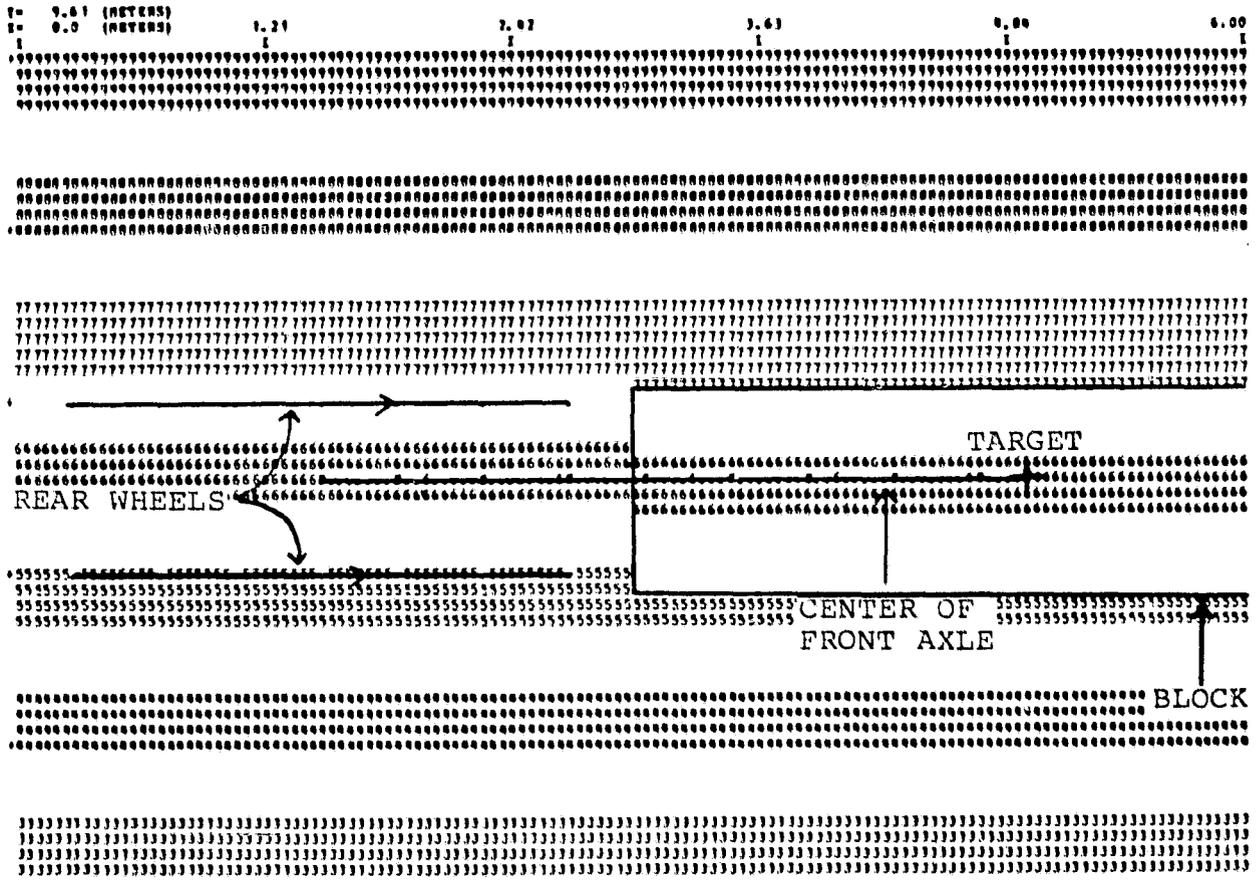
FIGURE 22

ROVER NEGOTIATING 15 DEGREE CROSSPATH
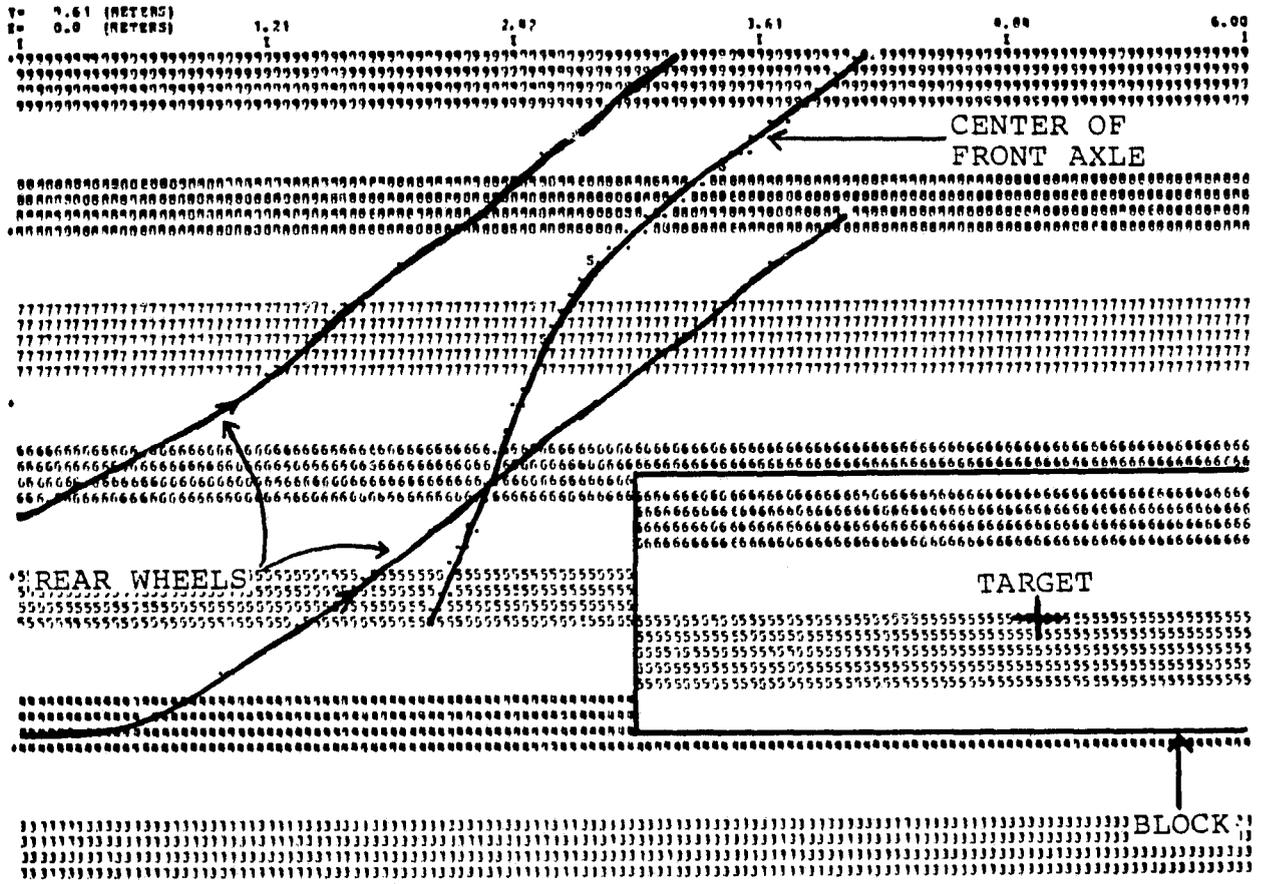
WITH SMALL BLOCK

FIGURE 23

ROVER NEGOTIATING 25 DEGREE CROSSPATH

WITH SMALL BLOCK

FIGURE 24

ROVER ATTEMPTING TO NEGOTIATE 30 DEGREE CROSSPATH

WITH SMALL BLOCK

4.4 Test of Advanced Modeler on Complex Terrain

Figure 25 shows a terrain consisting of a large crater, two smaller craters and four blocks and the path taken by the rover to get from the starting position to the target. Table 13 gives the first relative array seen by the rover. The level terrain outside the crater can be seen as all zeros. The negative numbers show the edge of the large crater and the positive numbers in the center azimuths are the returns from the small block in front of the rover. The missing returns just behind these are from the large block behind the smaller one and are caused by the fact that the top of the large block is outside the sensor field of view.

The rover decides that it can not go straight ahead and moves forward to the left. For the second scan shown in Table 14 the relative array shows the same features as the first scan except that they are to the right of the relative array since the rover turned to the left. As soon as the rover got past the two blocks it turned to the right and arrived at the target as shown in Figure 25.
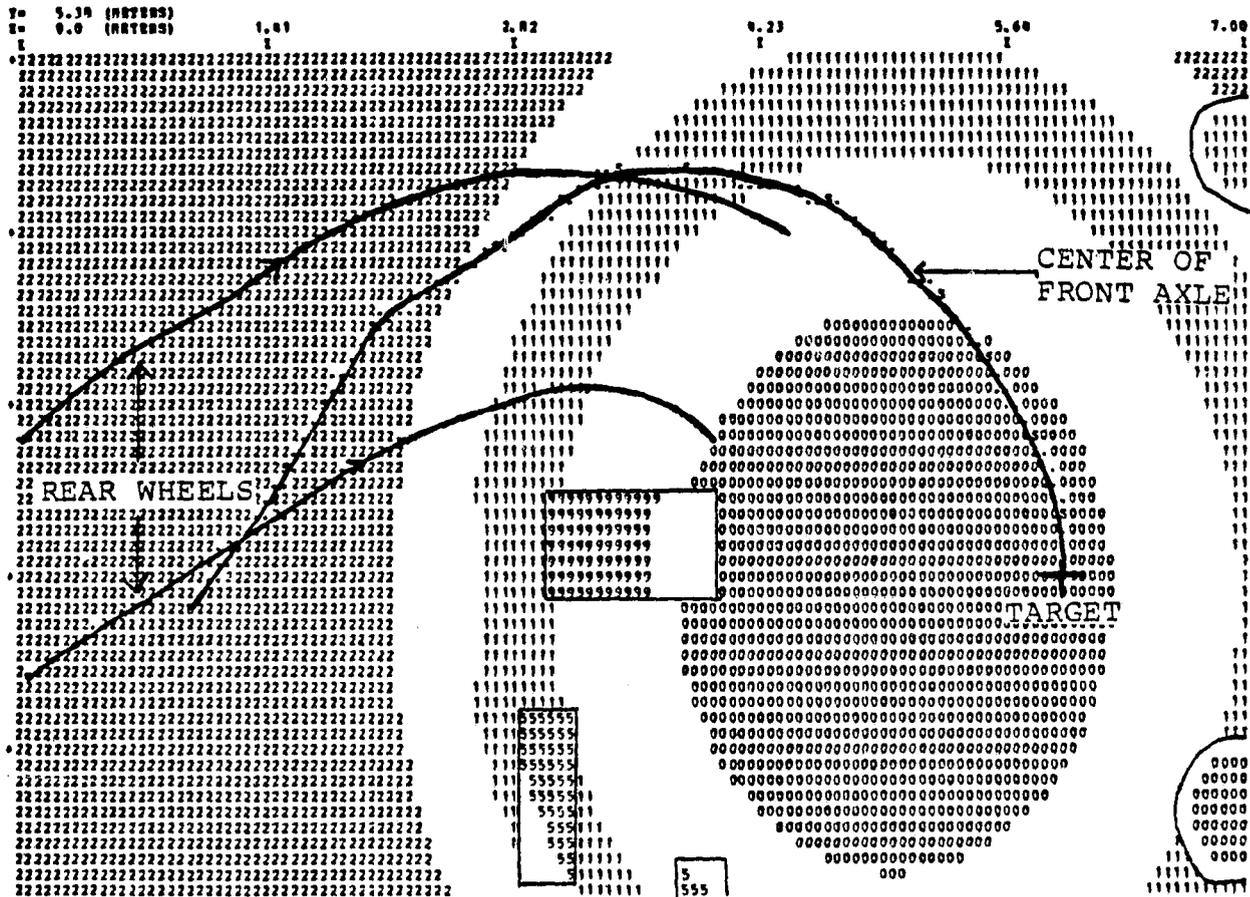
FIGURE 25

ROVER NEGOTIATING COMPLEX TERRAIN

AZIMUTH	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LASER															
32	0	0	0	-1	-2	-3	***	***	***	5	-3	-1	0	0	0
31	0	0	0	-1	-2	-3	***	***	***	4	2	-1	0	0	0
30	0	0	0	-1	-2	-3	***	***	***	3	1	-1	0	0	0
29	0	0	0	-1	-2	-3	***	***	***	2	0	-1	0	0	0
28	0	0	0	-1	-2	-3	***	***	***	1	-1	-1	0	0	0
27	0	0	0	-1	-2	-3	11	11	11	0	-2	-1	0	0	0
26	0	0	0	-1	-2	-2	9	10	9	-1	-2	-1	0	0	0
25	0	0	0	-1	-2	-2	8	9	8	-2	-2	-1	0	0	0
24	0	0	0	0	-2	-2	7	7	7	-2	-2	0	0	0	0
23	0	0	0	0	-1	-2	6	6	6	-2	-1	0	0	0	0
22	0	0	0	0	-1	-2	5	5	5	-2	-1	0	0	0	0
21	0	0	0	0	-1	-2	3	4	3	-2	-1	0	0	0	0
20	0	0	0	0	-1	-2	2	3	2	-2	-1	0	0	0	0
19	0	0	0	0	-1	-1	1	2	1	-1	-1	0	0	0	0
18	0	0	0	0	-1	-1	0	1	0	-1	-1	0	0	0	0
17	0	0	0	0	0	-1	-1	0	-1	-1	0	0	0	0	0
16	0	0	0	0	0	-1	-1	-1	-1	-1	0	0	0	0	0
15	0	0	0	0	0	-1	-1	-1	-1	-1	0	0	0	0	0
14	0	0	0	0	0	0	-1	-1	-1	0	0	0	0	0	0
13	0	0	0	0	0	0	-1	-1	-1	0	0	0	0	0	0
12	0	0	0	0	0	0	-1	-1	-1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 13

Relative Array from First Scan

of Terrain in Figure 25

AZIMUTH	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LASER															
32	0	0	0	0	0	0	0	0	-2	-2	-3	-3	***	***	***
31	0	0	0	0	0	0	0	0	-2	-2	-3	-4	***	***	***
30	0	0	0	0	0	0	0	0	-2	-3	-3	-3	***	***	***
29	0	0	0	0	0	0	0	0	-2	-3	-3	-3	***	***	***
28	0	0	0	0	0	0	0	0	-2	-2	-3	-3	***	***	***
27	0	0	0	0	0	0	0	0	-2	-2	-3	-3	***	***	***
26	0	0	0	0	0	0	0	0	-1	-2	-3	-3	***	***	***
25	0	0	0	0	0	0	0	0	-1	-2	-3	-3	***	***	***
24	0	0	0	0	0	0	0	0	-1	-2	-3	-3	***	***	14
23	0	0	0	0	0	0	0	0	-1	-2	-3	-3	***	15	12
22	0	0	0	0	0	0	0	0	-1	-2	-3	-3	15	14	11
21	0	0	0	0	0	0	0	0	-1	-2	-3	-3	14	12	10
20	0	0	0	0	0	0	0	0	-1	-2	-2	-3	12	11	8
19	0	0	0	0	0	0	0	0	-1	-2	-2	-3	11	9	7
18	0	0	0	0	0	0	0	0	-1	-2	-2	-3	9	8	6
17	0	0	0	0	0	0	0	0	-1	-2	-2	-2	8	7	4
16	0	0	0	0	0	0	0	0	-1	-1	-2	-2	6	5	3
15	0	0	0	0	0	0	0	0	-1	-1	-2	-2	5	4	2
14	0	0	0	0	0	0	0	0	0	-1	-2	-2	4	3	1
13	0	0	0	0	0	0	0	0	0	-1	-2	-2	3	2	0
12	0	0	0	0	0	0	0	0	0	-1	-1	-2	1	1	-1
11	0	0	0	0	0	0	0	0	0	-1	-1	-2	0	0	-1
10	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1
9	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1
8	0	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	-1
7	0	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	-1
6	0	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	-1
5	0	0	0	0	0	0	0	0	0	0	0	-1	-1	-1	0
4	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 14

Relative Array from Second Scan

of Terrain in Figure 25

5. CONCLUSIONS

The advanced terrain modeler presented in this report is a viable solution to the problem of choosing a safe path for the rover at Rensselaer to travel. Crosspath as well as inpath hazards are taken into consideration by the advanced modeler.

Tests have shown that a 32 laser/40 detector configuration with a cone size of 0.75 degrees can distinguish inpath slopes within the range of plus or minus twenty degrees as safe terrain. The modeler can also distinguish crosspath slopes free from hazards inside the range of plus or minus thirty degrees as safe terrain. Crosspath slopes with hazards on them give the modeler a more difficult time, but safe terrain could still be distinguished in the range of plus or minus twenty degrees.

Future work on terrain modelers at Rensselaer should focus on two topics: more advanced methods of performing the crosspath analysis, and alternate methods of hazard detection. The crosspath analysis on the advanced terrain modeler depends on the current pitch and roll of the rover to compute the crosspath slope. It is possible to imagine terrains where the crosspath slope is greatly different from that calculated by the pitch and roll of the

rover, so methods that just use the information in the relative array should be developed for computing the crosspath. The present methods of calculating the crosspath will always be good as the rover approaches very close to the terrain being examined but new approaches might improve the performance of the terrain modeler at greater distances.

The work on alternate methods of hazard detection could follow up on the work done by Maroon[2] involving pattern analysis and possibly try combining some of the numerical methods used in the advanced terrain modeler with the new techniques developed. Developments in these areas could provide interesting results and even suggest more ways to improve the numerical techniques used in the terrain modeler presented in this report.

REFERENCES

[1] G. Maroon, "Development of a Multiple Laser-Detector
     Hazard Detection system for an Autonomous Martian
     Rover," Master of Engineering Project Report,
     Rensselaer Polytechnic Institute, Troy, N.Y., May,
     1977.

[2] N. Troiani, and S. W. Yerasunis, "Procedures for the
     Interpretation and Use of Elevation Scanning
     Laser/Multi-Sensor Data for Short Range Hazard
     Detection and Avoidance for an Autonomous Planetary
     Rover," RPI Technical Report MP-57, Rensselaer
     Polytechnic Institute, Troy, N.Y., July, 1978.

[3] M. T. Krajewski, "Development and Evaluation of a
     Short Range Path Selection System for an Autonomous
     Martian Rover," Master of Engineering Project Report,
     Rensselaer Polytechnic Institute, May, 1976.

[4] S. W. Yerazunis, D. K. Frederick, E. L. Hunter, and N.
     Troiani, "Guidance of an Autonomous Planetary Rover
     Based on a Short Range Hazard Detection System," Tenth
     Annual Pittsburgh Conference on Modeling and
     Simulation, Pittsburgh, Pennsylvania, April, 1979.

APPENDIX

SOURCE CODE FOR SUBROUTINES

DEVELOPED UNDER THIS PROJECT

(FLOW DIAGRAM IS SHOWN IN FIGURE 11)

- MODEL3

- CALSLP

```
      SUBROUTINE MODEL3
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                  C
C     THIS IS THE JUNE 1,1979 VERSION CF SUBROUTINE MODEL3         C
C     THIS SUBROUTINE IS A TERRAIN MODELER WRITTEN BY ERWIN HUNTER C
C     WHICH IS A MODIFICATION CF NICK TROIANI'S CONCEPTS           C
C                                                                  C
C     THE FOLLOWING ARE SCME CF THE ARRAYS USED IN THIS PROGRAM    C
C                                                                  C
C        DATA(I,J) - ACTUAL SENSOR DATA FOR ASMUTH I, LASER J      C
C        DIAG(I,J) - DIAGONALIZED RETURN FOR ASMUTH I, LASER J     C
C        POS(I,J,1) - HEIGHT OF THE ITH LASER, JTH SENSOR INTERSECTION C
C        POS(I,J,2) - RANGE CF THE ITH LASER, JTH SENSOR INTERSECTION  C
C                                                                  C
C                                                                  C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      COMMON /CHOOSE/
     1    NMMOD,NMSEN,NMPSA,NMFAIL,INTVDB,INTSEN,
     2    INTMOD,INTPSA,INTFAL,INTGYR
      COMMON /TIEUP/
     1    THETNU,ALPHA,SLPIN,SLPCRS,TALLOW
      COMMON /DYNMIC/
     1    DWNMAX,UPMAX,VEHLEN,VEHWID,CRSMAX,VELMAX,DT,TURN1,
     2    STRMAX,STPMAX
      COMMON /SENSR/
     1    ASMUTH,LANGLE,RTN
      COMMON /DETECT/
     1    SENTIM,SIMSTP,SENLIM,IJK
      COMMON /SENX/
     1    HITLAS,HITSEN,NUMLAS,NUMSEN,NUMAZ,INTDAT,NMDTPR,LASAGL,
     2    SENGLE,SCON,DATA,DIAG,POS,SEN1ST
C
C
      REAL * 4  RANGE(50),SENGLE(50),LANGLE(50),AZMUTH(50),LASAGL(50),
     1    ASMUTH(50),POS(50,51,2)
      INTEGER * 4  HAZARD(50),RTN(50),DIAG(50,50),SEN1ST
      INTEGER * 2  DATA(50,50)
      LOGICAL * 1  OPMAX
      LIST = 6
C
CCC   THIS IS THE INITIALIZATION SECTION
C
      IF(IJK .GT. 0)  GO TO 5
      READ(5,1000)  XMAX,ZMAX,RNGMIN
 1000 FORMAT(3F10.5)
      IF (INTMOD .EQ. 1)  WRITE(LIST,1050) XMAX,ZMAX,RNGMIN
 1050 FORMAT(//10X,'THIS IS THE INITALIZATION SECTION OF MODEL3',
     1 //10X,'THE MINIMUM ALLOWABLE DISTANCE FOR MISSING RETURNS IS    ',
     1 F10.3,
     2  /10X,'THE MAXIMUM STEP CLIMBABLE IS                            ',
     2 F10.3,
```

```
      3  /10X,'THE MINIMUM RANGE BEFORE AVOIDING A HAZARD IS          ',
      4  F10.3)
         CONVRT = 180.0 / 3.14159
         BCON = 1.0 / CONVRT
         UPMAXD = UPMAX * CONVRT
         DWMAXD = DWNMAX * CONVRT
         RETURN
 5       CONTINUE
C
C  CONSTRUCT THE DIAGONALIZED RETURN FOR THE LASER-SENSOR DATA IN
C  ARRAY DATA
C
         CALL DIAGNL
         DO 7  I=1,50
         HAZARD(I) = 1
 7       CONTINUE
C
C  PROCESS EACH ASMUTH, ONE AT A TIME
C
         DO 100  J=1,NUMAZ
C
CCC  NOW SEARCH FOR ANY MISSING RETURNS
C
C  NALAS IS THE LAST LASER IN EACH AZIMUTH WHICH ONLY HAS SAFE TERRAIN
C        IN FRONT OF IT
C  HAZARD = 1 MEANS THAT NO HAZARDS HAVE BEEN FOUND FOR THIS AZIMUTH YET
C  IMISS IS THE NUMBER OF CONSECUTIVE MISSING RETURNS FOUND SO FAR
C
         NALAS = NUMLAS
         IMISS = 0
C
C   FOR EACH LASER SHOT IN THIS AZIMUTH, DETERMINE IF A SENSOR DETECTED
C  IT
C
         DO 50  I=1,NUMLAS
C
C  DIAG(J,I) IS THE SENSOR WHICH DETECTED THE ITH LASER ON THE JTH
C   AZIMUTH
C  IF DIAG(J,I) = 1000 THEN NO SENSOR SAW THE LASER SHOT
C
         IF(DIAG(J,I) .LT. 1000)  GO TO 30
C
C  IF NO SENSOR SAW THE LASER SHOT THEN:
C     INCREMENT THE NUMBER OF CONSECUTIVE MISSING RETURNS
C     IF THIS IS THE FIRST CONSECUTIVE MISSING RETURN THEN SET ISTRT
C          EQUAL TO THE LASER NUMBER AND FIRSTH TO THE POSITION
C          OF THE LAST HIT
C     THEN CONTINUE WITH THE NEXT LASER SHOT
C
         IMISS = IMISS + 1
         IF(IMISS .LE. 1)  ISTRT = I
         GO TO 50
 10      CONTINUE
C
C  IF TOO MANY MISSING RETURNS WERE FOUND, THEN SET
```

```
C      RANGE = THE RANGE OF THE LAST LASER-SENSOR INTERSECTION WHICH
C      FOUND SAFE TERRAIN AND INDICATE THAT TOO MANY MISSING RETURNS
C      WERE FOUND BY SETTING HAZARD = 2
C
       RANGE(J) = 0.0
       IF(ISTRT .LE. 1)   GO TO 20
       LGOOD = ISTRT - 1
       ILAS = ISTRT - 1
       ISEN = DATA(J,ILAS)
       RANGE(J) = POS(ILAS,ISEN,2)
  20   CONTINUE
       HAZARD(J) = 2
       GO TO 60
  30   CONTINUE
       LGOOD = I
       IF(IMISS .EQ. 0)   GO TO 50
       IF(I .GE. NUMLAS) LGOOD = ISTRT - 1
C
C   DETERMINE THE LENGTH OF THE MISSING RETURN FIELD
C
       ISEN = DATA(J,I) - I + ISTRT
       DIST = ABS(POS(I-1,DATA(J,I)-1,2)-POS(ISTRT,ISEN,2))
       IF(DIST .GE. XMAX)   GO TO 10
C
C   IF THE NUMBER OF CONSECUTIVE MISSING RETURNS FOUND WAS LESS THAN
C   THE NUMBER ALLOWABLE, THEN
C      SET DIAG AND DATA EQUAL TO THE MINIMUM OF THE SURRONDING TERRAIN
C      RESET IMISS TO 0
C
       IFILL = 0
       IF(ISTRT .GT. 1)   IFILL = DIAG(J,ISTRT-1)
       IF(IFILL .GT. DIAG(J,I))   IFILL = DIAG(J,I)
       DO 40  K=ISTRT,IMISS
       DIAG(J,K) = IFILL
       DATA(J,K) = IFILL + K + SENIST - 1
  40   CONTINUE
       IMISS = 0
  50   CONTINUE
  60   CONTINUE
C
C   IF THE NUMBER OF CONSECUTIVE MISSING RETURNS IS NOT ZERO (EITHER
C      BECAUSE THERE WERE MISSING RETURNS AT THE END OF THE AZIMUTH
C      OR BECAUSE MORE THAN MXMISS CONSECUTIVE MISSING RETURNS WERE
C      FOUND), THEN SET NALAS TO THE LAST GOOD LASER SHOT
C
       IF(IMISS .NE. 0)   NALAS = LGOOD
C
C   THETA IS THE SLOPE OF THE ROVER IN RESPECT TO THE CURRENT AZIMUTH
C   CRSPTH IS THE CROSS-PATH SLOPE IN THE CURRENT AZIMUTH
C      ALPHA IS THE HEADING ANGLE
C      SLPIN IS THE INPATH SLOPE
C      SLPCRS IS THE CROSS PATH SLOPE
C
       DELTA = ASMUTH(J) + ALPHA
CCCC
```

OF POOR QUALITY

- A4 -

```
CCCC   NOTE THAT TO IMPROVE THE SPEED OF THIS MODELER THE SMALL ANGLE
CCCC   APPROXIMATION CAN BE USED FOR ASIN( ), SIN(SLPIN), AND
CCCC   SIN(SLPCRS) SINCE THE IMPORTANT VALUES OF THESE ARE ALL LESS
CCCC   THAN 30 DEGREES.  THE SMALL ANGLE APPROXIMATION CAN NOT BE
CCCC   USED FOR DELTA SINCE DELTA COULD BE AS LARGE AS 160 DEGREES
CCCC
       SININ = SIN(SLPIN)
       SINCRS = SIN(SLPCRS)
       COSD = COS(DELTA)
       SIND = SIN(DELTA)
       THETA = ASIN(SININ*COSD - SINCRS*SIND)
       OUTSLP = THETA * CONVRT
       CRSPTH = ASIN(SININ*SIND + SINCRS*COSD) * CONVRT
C
CCC   NOW COMPUTE THE MAXIMUM SLOPES
C
C      LASTD IS THE DIAGONALIZED RETURN FROM THE LAST LASER SHOT
C      ILAST IS THE LAST LASER THAT SHOT BEFORE THE CURRENT SLOPE
C      CPMAX TELLS CALSLP TO CALCULATE THE MAXIMUM SLOPES
C      ZMAX IS THE MAXIMUM STEP THE ROVER CAN NEGOTIATE
C
       LASTD = 0
       ILAST = -2
       CPMAX = .TRUE.
       ZTEST = ZMAX
C
C   IF THE CROSS-PATH FOR THIS AZIMUTH IS GREATER THAN 30 DEGREES
C   THEN FLAG THE AZIMUTH AS BEING HAZARDOUS
C
       ILAS = 1
       ISEN1 = DATA(J,1)
       IF(ABS(CRSPTH) .GE. 30.0) GO TO 75
       DO 70  I=1,NALAS
C
C  -- PERFORM THE CROSS-PATH ANALYSIS --
C    ON ANY AZIMUTH, IF THERE IS A STEP DIFFERENCE BETWEEN IT AND
C    THE NEXT AZIMUTH GREATER THAN ZMAX, THEN FLAG BOTH THE AZIMUTH
C    AND THE NEXT AZIMUTH AS HAZARDOUS
C
C                      - OR -
C    IF THE CROSS-PATH SLOPE OF THE ROVER ON THAT AZIMUTH
C    WOULD BE GREATER THAN 20 DEGREES, THEN FLAG ANY POSITIVE
C    OBSTACLES AS BEING HAZARDS
C    IF THE CROSS-PATH SLOPE OF THE ROVER ON THAT AZIMUTH
C    WOULD BE LESS THAN MINUS 20 DEGREES, THEN FLAG ANY NEGATIVE
C    OBSTACLES AS BEING HAZARDS
C
       IF(J .LE. 1)  GO TO 64
       ISEN1 = DATA(J,I)
       ISEN2 = DATA(J-1,I)
       IF(ISEN2 .LE. 0)  GO TO 64
       ILAS = I
       IF(CRSPTH .GE. 20.0)  GO TO 61
       IF(CRSPTH .LE. -20.0)  GO TO 62
       IF(ABS(POS(I,ISEN1,1)-POS(I,ISEN2,1)) .GT. ZMAX) GO TO 75
```

```
       GO TO 64
 61    CONTINUE
       IF(ISEN2 .GT. ISEN1)  GO TO 75
       GO TO 64
 62    CONTINUE
       IF(ISEN2 .LT. ISEN1)  GO TO 75
 64    CONTINUE
C
C  IF THE DIAGONALIZED RETURN DID NOT CHANGE, AND WE ARE NOT ON THE
C     LAST LASER SHOT, THEN GO TO THE NEXT LASER SHOT
C
       IF(DIAG(J,I) .EQ. LASTD .AND. I .NE. NALAS)  GO TO 70
       LASTD = DIAG(J,I)
C
C  ON THE LAST LASER SHOT FORCE CALSLP TO CALCULATE THE SLOPE BY
C  SETTING THE MAXIMUM STEP EQUAL TO A NEGATIVE NUMBER
C  AND IF NO SLOPES WERE CALCULATED FOR THIS AZIMUTH, THEN CALCULATE
C  THE AVERAGE SLOPE FOR THE ENTIRE AZIMUTH
C
       IF(I .NE. NALAS)  GO TO  65
       ZTEST = -1.0
       IF(ILAST .LT. -1)  ILAST = 0
 65    CONTINUE
C
C  IF ILAST = -1, THEN THIS IS THE START OF A NEW SLOPE, SO SET ILAST
C  EQUAL TO THE LAST LASER SHOT BEFORE THIS SLOPE
C
       IF(ILAST .LT. 0)  ILAST = I-1
       CALL CALSLP(I,ILAST,J,ZTEST,OPMAX,SLOPE,DIST)
C
C  IF A SLOPE WAS CALCULATED, THEN ILAST WILL BE SET TO -1
C  IF NO SLOPE WAS CALCULATED, THEN THE CHANGE IN HEIGHT WAS LESS THAN
C  ZMAX, SO FIND THE NEXT JUMP IN THIS SLOPE
C
       IF(ILAST .GE. 0)  GO TO 70
C
C  IF A SLOPE WAS CALCULATED, THEN TRANSFORM IT INTO THE PLANET FRAME
C     AND INTO DEGREES(FROM RADIANS)
C  AND TEST WHETHER IT IS WITHIN THE ROVER'S CLIMING ABILITY
C
       SLOPE = (SLOPE+THETA)*CONVRT
       IF(INTMOD .EQ. 1)  WRITE(LIST,900)  J,OUTSLP,SLOPE
 900 FORMAT('     FOR THE ',I2,' AZIMUTH THE SLOPE OF THE TERRAIN IS ',
     1 F7.2,' DEGREES AND THE MAX SLOPE IS ',F7.2,' DEGREES')
       IF(SLOPE .LT. UPMAXD .AND. SLOPE .GT. DNMAXD)  GO TO 70
C
C  IF THE SLOPE IS HAZARDOUS, THEN INDICATE THAT IT IS
C  AND END COMPUTATIONS ON THIS AZIMUTH IF IT IS TOO CLOSE
C
       IF(DIST .GT. RNGMIN)  GO TO 80
       HAZARD(J) = 3
       RANGE(J) = DIST
       GO TO 100
 70    CONTINUE
       GO TO 100
```

```
C
C   IF A CROSS-PATH HAZARD WAS DETECTED, THEN MARK THE TWO
C   AZIMUTHS BEING CHECKED AND CONTINUE TO THE NEXT AZIMUTH
C
 75     CONTINUE
        HAZARD(J) = 6
        HAZARD(J-1) = 6
        RANGE(J) = POS(ILAS,ISEN1,2)
        RANGE(J-1) = RANGE(J)
        GO TO 100
C
C   OTHERWISE, IF THE HAZARD IS NOT TOO CLOSE, COMPUTE THE MINIMUM SLOPES
C
 80     CONTINUE
C
CCC   NOW COMPUTE THE MINIMUM SLOPES
C
C       LASTD IS THE DIAGONALIZED RETURN FROM THE LAST LASER SHOT
C       ILAST IS THE LAST LASER THAT SHOT BEFORE THE CURRENT SLOPE
C       OPMAX TELLS CALSLP TO CALCULATE THE MINIMUM SLOPES
C
        LASTD = 0
        ILAST = -1
        OPMAX = .FALSE.
        DO 90  I=1,NALAS
C
C    IF THE DIAGONALIZED RETURN DID NOT CHANGE THEN GO TO THE NEXT LASER
C SHOT
C
        IF(DIAG(J,I) .EQ. LASTD)  GO TO 90
        LASTD = DIAG(J,I)
C
C   IF ILAST = -1, THEN THIS IS THE START OF A NEW SLOPE, SO SET ILAST
C   EQUAL TO THE LAST LASER SHOT BEFORE THIS SLOPE
C
        IF(ILAST .LT. 0)  ILAST = I-1
C
C   NOW CALCULATE THE MINIMUM SLOPE
C
        CALL CALSLP(I,ILAST,J,ZMAX,OPMAX,SLOPE,DIST)
C
C   IF A SLOPE WAS CALCULATED, THEN ILAST WILL BE SET TO -1
C   IF NO SLOPE WAS CALCULATED, THEN THE CHANGE IN HEIGHT WAS LESS THAN
C   ZMAX, SO FIND THE NEXT JUMP IN THIS SLOPE
C
        IF(ILAST .GE. 0)  GO TO 90
C
C   IF A SLOPE WAS CALCULATED, THEN TRANSFORM IT INTO THE PLANET FRAME
C       AND INTO DEGREES(FROM RADIANS)
C   AND TEST WHETHER IT IS WITHIN THE ROVER'S CLIMING ABILITY
C       IF IT IS, THEN FIND THE NEXT MINIMUM SLOPE
C
        SLOPE = (SLOPE+THETA) * CONVRT
        IF(INTMOD .EQ. 1)  WRITE(LIST,910)  J,OUTSLP,SLOPE
 910 FORMAT('    FOR THE ',I2,' AZIMUTH THE SLOPE OF THE TERRAIN IS ',
```

- A7 -

```
      1 F7.2,' DEGREES AND THE MIN SLOPE IS ',F7.2,' DEGREES')
        IF(SLOPE .LT. UPMAXD .AND. SLOPE .GT. DNMAXD)   GO TO 90
C
C  IF THE MINIMUM SLOPE IS NOT WITHIN THE ROVER'S CLIMING ABILITY THEN
C  THE SLOPE IS DEFINITELY A HAZARD AND SHOULD BE AVOIDED
C
        HAZARD(J) = 4
        RANGE(J) = DIST
        GO TO 100
 90     CONTINUE
        IF(HAZARD(J) .EQ. 1)   HAZARD(J) = 5
 100  CONTINUE
C       OUTPUT DATA ON AZIMUTHS
        WRITE(6,1100)
 1100   FORMAT('1','AZIMUTH',46X,'TERRAIN CHARACTERIZATION',43X,'RTN')
        DO 20000 J=1,NUMAZ
        I=HAZARD(J)
        GO TO(110,120,130,140,150,160),I
 110    RTN(J)=1
        WRITE(6,12500)J
12500   FORMAT('0',3X,I2,5X,'OBSTACLES DETECTED ARE NOT HAZARDOUS.'
      1        ,'  TERRAIN IS PASSABLE.',52X,'1')
        GO TO 20000
 120    RTN(J)=0
        WRITE(6,6500)J,RANGE(J)
 6500   FORMAT('0',3X,I2,5X,'MISSING RETURNS DETECTED BEGINNING',
      1    ' AT A RANGE OF ',F3.1,' METERS.   TERRAIN IS NOT PASSABLE.',
      2      25X,'0')
        GO TO 20000
 130    RTN(J)=0
        WRITE(6,9500)J,RANGE(J)
 9500   FORMAT('0',3X,I2,5X,'OBSTACLE DETECTED AT ',F3.1,' METER RANGE'
      1      ,' WITH POSSIBLY HAZARDOUS SLOPE.   TERRAIN WILL BE AVOIDED.',
      2      18X,'0')
        GO TO 20000
 140    RTN(J)=0
        WRITE(6,10500)J,RANGE(J)
10500 FORMAT('0',3X,I2,5X,'OBSTACLE DETECTED AT ',F3.1,' METER RANGE ',
      1      'WITH DEFINITELY HAZARDOUS SLOPE.   TERRAIN IS NOT PASSABLE.',
      2      16X,'0')
        GO TO 20000
 150    RTN(J)=1
        WRITE(6,11500)J
11500   FORMAT('0',3X,I2,5X,'POSSIBLE OBSTACLE DETECTED BUT NOT CLOSE ',
      1        'ENOUGH TO NECESSITATE AVOIDANCE.',38X,'1')
        GO TO 20000
 160    CONTINUE
        RTN(J) = 0
        WRITE(LIST,12000)   J,RANGE(J)
12000 FORMAT('0',3X,I2,5X,'HAZARDOUS CROSS PATH DETECTED AT RANGE ',
      1 F3.1,' METERS, TERRAIN IS NOT PASSABLE.',36X,'0')
20000   CONTINUE
        RETURN
        END
```

```
      SUBROUTINE CALSLP(I,ILAST,IASM,ZMAX,OPMAX,SLOPE,XLAST)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                   C
C     THIS IS SUBROUTINE CALSLP WHICH CALCULATES THE MAXIMUM        C
C     OR MINIMUM SLOPES FOR MODEL3                                  C
C                                                                   C
C     IF OPMAX IS TRUE, THEN THE MAX SLOPE WILL BE CALCULATED       C
C       OTHERWISE THE MINIMUM SLOPE WILL CALCULATED                 C
C     ZMAX IS THE MAXIMUM SLOPE THE ROVER CAN HANDLE                C
C     SLOPE IS THE VALUE OF THE SLOPE RETURNED BY THE SUBROUTINE    C
C       IF A SLOPE IS CALULATED BY THIS ROUTINE, THEN ILAST IS SET  C
C       TO -1 TO INDICATED THAT THE VALID SLOPE WAS CALCULATED      C
C                                                                   C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      COMMON /SENX/
     1    HITLAS,HITSEN,NUMLAS,NUMSEN,NJMAX,INTDAT,NMDTPR,LASAGL,
     2    SENGLE,SCON,DATA,DIAG,POS,SEN1ST
      REAL * 4  POS(50,51,2),LASAGL(50),SENGLE(50)
      INTEGER * 4  DIAG(50,50),SEN1ST
      INTEGER * 2  DATA(50,50)
      LOGICAL * 1  OPMAX
C
C
C  IF ILAST IS NEGATIVE, THEN SET IT TO ZERO INCASE NO SLOPE
C     IS CALCULATED
C
      IF(ILAST .LT. 0) ILAST = 0
      YLAST = 0.0
      LSEN = 0
      XLAST = 0.0
C
C  IF ILAST IS NOT ZERO, THEN CALCULATE WHICH SENSOR SAW THE FIRST
C     VALID LASER SHOT
C
      IF(ILAST .GT. 0)  LSEN = DIAG(IASM,ILAST)
C
C  IF YOU ARE COMPUTING THE MAX SLOPE AND THE SLOPE IS GOING:
C    - UPHILL  THEN ADD 1 TO THE LAST LASER IN THE SLOPE
C    - DOWNHILL  THEN ADD 1 TO THE FIRST LASER IN THE SLOPE
C  IF YOU ARE COMPUTING THE MIN SLOPE AND THE SLOPE IS GOING:
C    - UPHILL  THEN ADD 1 TO THE FIRST LASER IN THE SLOPE
C    - DOWNHILL  THEN ADD 1 TO THE LAST LASER IN THE SLOPE
C
      N = 0
      IF(DIAG(IASM,I) .LT. LSEN) N = 1
      IF(.NOT. OPMAX) N = 1 - N
      IF(ILAST .LE. 0)  GO TO 10
      LSEN = DATA(IASM,ILAST) + N
      YLAST = POS(ILAST,LSEN,1)
 10   CONTINUE
      N = 1 - N
```

```
C
C   NOW COMPUTE THE SENSOR THAT SAW THE LAST VALID LASER SHOT
C
      NSEN = DATA(IASM,I) + N
      YNEW = POS(I,NSEN,1)
C
C   IF THE CHANGE IN HEIGHT IS NOT GREATER THAN ZMAX, THEN RETURN
C
      DELTY = YNEW - YLAST
      IF(ABS(DELTY) .LT. ZMAX)  GO TO 20
C
C   OTHERWISE COMPUTE THE NEW SLOPE
C
      IF(ILAST .GT. 0)  XLAST = POS(ILAST,LSEN,2)
      XNEW = POS(I,NSEN,2)
      DELTX = ABS(XNEW - XLAST)
      SLOPE = ATAN(DELTY/DELTX)
      ILAST = -1
 20   CONTINUE
      RETURN
      END
```