REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June 1997	Tech Report

4. TITLE AND SUBTITLE
Approximation Probability of Detection in the Janus Model

5. FUNDING NUMBERS

6. AUTHOR(S)

Mickey A. Sanzotta
E. Todd Sherrill

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
OPERATIONS RESEARCH CENTER
USMA
WEST POINT, NY 10996

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

12b. DISTRIBUTION CODE

19980804 063

13. ABSTRACT *(Maximum 200 words)*
This report documents our approach at developing a satisfactory detection model for use in computing information gain in the Janus Wargame. Information gain measures the Blue forces' awareness of Red's disposition over time. Within a time interval (t, t+1) the measure is a distance measure between two probability distributions Pt and Pt+1 respectively. These distributions represent the relative discrete probability, from Blue's perspective, that a Red vehicle is in a particular area of the battlefield. The sum of the discrete probability values over all areas is 1.0 with those areas of greatest likelihood having the larger values. Information is generated by the actions of Blue sensors. When any Blue sensor scans an area of the battlefield Blue gains information about the enemy disposition. The magnitude of this new information is determined in part by the efficiency of the Blue sensor. As new information about the location of a Red vehicle becomes available to Blue we update the probability distribution using theoretical formulations based upon Bayes formula [4].

Though the theory is simple, its implementation in the Janus model was very challenging. The Bayesian formulation mentioned above requires two parameters during each time stage: 1) a listing of which areas of the battlefield (cells) Blue sensors looked in and 2) the probability of detection (PD) for the sensors that did the respective scanning. Neither of these data are directly available in Janus runs, nor can they be deduced from Janus output files. For example, the Janus algorithms for line of sight computations and detection of enemy vehicles are only called when two opposing vehicles are within some threshold of proximity to each other. Since information gain credits finding where the enemy is not, we need to know at each time increment what terrain cells Blue sensors have searched regardless of the presence or absence of enemy vehicles. Likewise, we need to know what the PD would have been for each particular sensor / cell pair had there been an enemy vehicle present when the sensor searched the cell. For our purposes, a sensor is considered to have searched a cell if it has unobscured line of sight between its position and the particular terrain cell.

14. SUBJECT TERMS	15. NUMBER OF PAGES
JANUS	20
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
unclassified	unclassified	unclassified	

United States Military Academy
West Point, New York 10996

APPROXIMATING PROBABILITY OF DETECTION IN THE JANUS MODEL

OPERATIONS RESEARCH CENTER
TECHNICAL REPORT
JUNE, 1997

DTIC QUALITY INSPECTED 1

Approximating Probability of Detection in The Janus Model

MAJ Mickey A. Sanzotta
Department of Mathematical Sciences

and

MAJ E. Todd Sherrill
Operations Research Center

U.S. Military Academy
West Point, NY 10922

A Technical Report
of the
Operations Research Center
United States Military Academy

May 1997

ACKNOWLEDGMENTS

The authors expresses deep appreciation for the guidance and tutelage received from the following members of the Staff and Faculty of the United States Military Academy:

Dr. Donald R. Barr, Professor, Department of Systems Engineering
MAJ Dave Olwell, Associate Professor, Department of Mathematical Sciences
LTC Patrick Driscoll, Professor, Department of Mathematical Sciences

And to the following for his guidance and consult of the subject area:

Mr. Paul West, Department of Systems Engineering.

Table of Contents

I. PURPOSE

This report documents our approach at developing a satisfactory detection model for use in computing *information gain* in the Janus Wargame. Information gain measures the Blue forces' awareness of Red's disposition over time. Within a time interval (t, t+1) the measure is a distance measure between two probability distributions P_t and P_{t+1} respectively. These distributions represent the relative discrete probability, from Blue's perspective, that a Red vehicle is in a particular area of the battlefield. The sum of the discrete probability values over all areas is 1.0 with those areas of greatest likelihood having the larger values. Information is generated by the actions of Blue sensors. When any Blue sensor scans an area of the battlefield Blue gains information about the enemy disposition. The magnitude of this new information is determined in part by the efficiency of the Blue sensor. As new information about the location of a Red vehicle becomes available to Blue we update the probability distribution using theoretical formulations based upon Bayes formula [4].

Though the theory is simple, its implementation in the Janus model was very challenging. The Bayesian formulation mentioned above requires two parameters during each time stage: 1) a listing of which areas of the battlefield (cells) Blue sensors looked in and 2) the probability of detection (P_D) for the sensors that did the respective scanning. Neither of these data are directly available in Janus runs, nor can they be deduced from Janus output files. For example, the Janus algorithms for line of sight computations and detection of enemy vehicles are only called when two opposing vehicles are within some threshold of proximity to each other. Since information gain credits finding where the enemy is not, we need to know at each time increment what terrain cells Blue sensors have searched regardless of the presence or absence of enemy vehicles. Likewise, we need to know what the P_D would have been for each particular sensor / cell pair had there been an enemy vehicle present when the sensor searched the cell. For our purposes, a sensor is considered to have searched a cell if it has unobscured line of sight between its position and the particular terrain cell.

This report explains an approach that was used to estimate P_D values for certain combinations of sensor-target pairs, as a function of sensor-cell distance. The methodology involves fitting curves to P_D values obtained from the Janus software and is illustrated with several example cases. Also discussed are alternative approaches to selection of P_D models and methods of fitting them to the Janus detection values.

II. MODEL SPECIFICATIONS

The model must produce P_D results in the interval [0,1]. We prefer the simplest function possible that adequately models detection. We chose to use a function or a subroutine in our computations rather than a table lookup method, since this would require us to store data tables on each sensor/target pair.

III. JANUS GRAPHICAL VALIDATION AND VERIFICATION

A. General.

The Janus software contains a graphical validation and verification (V&V) section. We discovered in this section, curves that provide a representation of probability of detection data for each observer-target pair. These graphs can be defined by either primary or secondary sensors against a stationary or moving enemy target, and can be varied from simple detection to actual identification of the enemy. We decided to try to replicate the information in these curves with a closed form function mapping the range from sensor to target into P_D. An example of the type of curves displayed in the Janus V&V section is shown below.

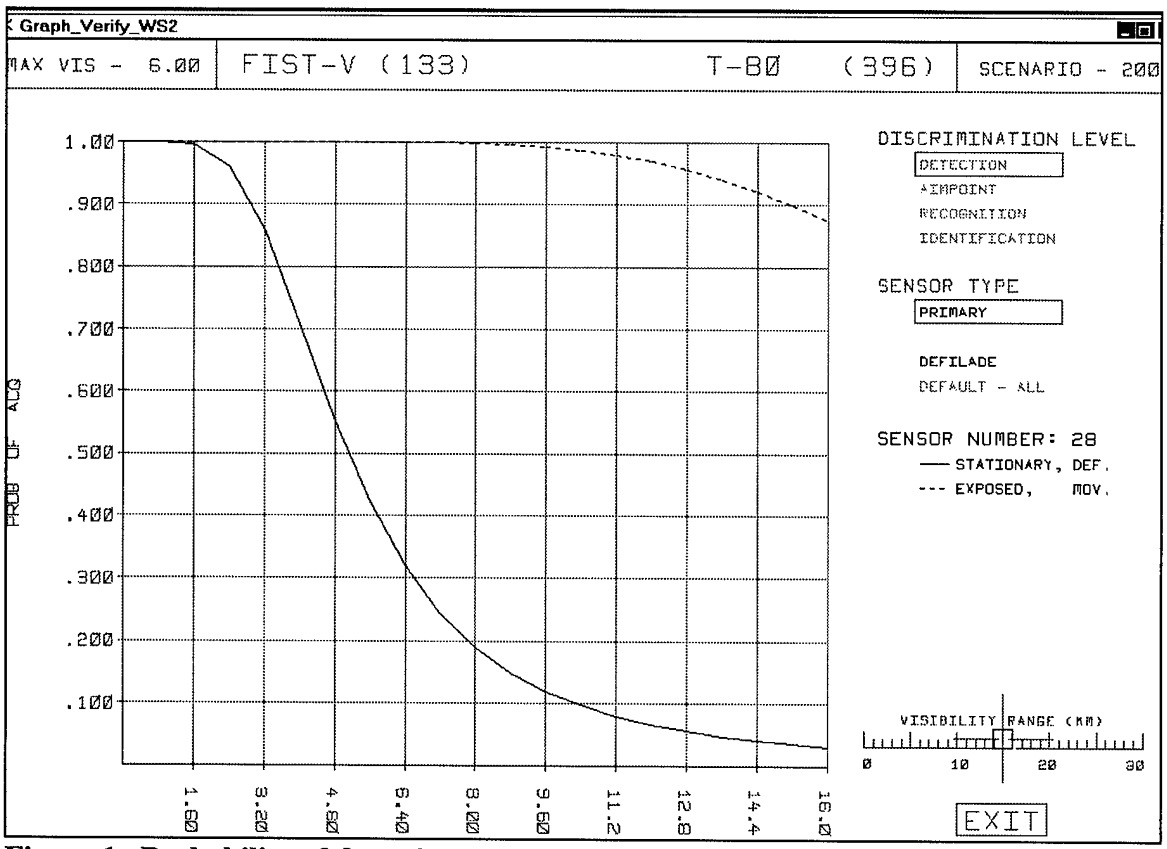

Figure 1. Probability of detection curve versus range (km) for a FistV w/ thermal sights vs. a stationary (solid line) T80.

B. Plot Analysis.

The plot of a FISTV seeking a stationary T80 with a Thermal Sight is a continuous, monotonically decreasing function that begins at one 1.0 and asymptotically approaches zero as range increases. This graph represents the probability of detection as a function of range, given friendly vehicle/sensor type and enemy vehicle. Although this is a simplification of the Janus algorithm, it still represents the general physical nature of detection probability. As range increases, the likelihood of detection decreases.

C. Replication of Janus Curves.

To replicate these curves, we entered each graph for an observer-target pair, extracted ordered pairs (range, P_D) from the graph, and fit a function to the ordered pairs. The data we extracted from the graph shown in Figure 1 is as follows:

Table 1. Data extracted from Janus V&V graphical screen, FistV with thermal sights vs. T80.

Range	P_D
.0001	.9999
.8	.9988
1.6	.99
2.4	.96
3.2	.86
4.0	.7
4.8	.55
5.7	.4
6.4	.315
8.0	.19
9.6	.13
11.2	.085
12.8	.06
14.4	.04
16.0	.02

Since this is an arduous process, we decided to limit the number of observer/target pairs we would model. These observer-target pairs are actually an ordered triple of vehicle-sensor-target, since a vehicle can have more than one sensor on board. We decided that we would model the primary and secondary sensors on both ground and aerial systems against an enemy T80 tank. The friendly systems we modeled are the FistV, M1, M2 on the ground, and the AH-64 and OH58D in the air. We wanted to use "killer" systems as well as "supporting" systems to diversify the P_D curves and compare their differences. In comparing the curves we tended to favor those that most accurately modeled the killer systems, since there are a greater number of killer systems deployed on a battlefield. The table below depicts the vehicle-sensor-target triples we modeled.

Table 2. Vehicle-Sensor-Target matrix.

Ground System vs. T80*		Aerial System vs. T80*	
FistV	Thermal	AH-64	Thermal
	Optical		Flir
M1	Thermal	OH-58D	Thermal
	Optical		Flir
M2	Thermal		
	Optical		
*T80 is in Stationary Mode			

For each of these triples, we produced a table from the corresponding Janus V&V Screen, depicting the range and P_D for each friendly vehicle-sensor against a stationary T80. The tabulated data can be found in appendix A.

IV. CHOOSING A MODEL

Since the data defines Probability of Detection (P_D) as a function of Range for each sensor, we were able to fit a curve to the data with standard mathematical techniques. The fitting procedures we attempted were polynomial curve fitting, cubic spline interpolation, fitting inverse functions to the data, conducting straight interpolation of tabled data, and performing logistics regression on the data. The method we chose was logistics regression. Appendix B briefly explains the other modeling methods. We chose logistics regression because the data are produced by a binary operation (detection). Logistics regression also allows us to perform multiple linear regression on the data set. Thus one regression equation in our algorithm will provide an estimated P_D for all ground and aerial systems. By using this method we could create a data matrix with the range and P_D, along with which type of vehicle and sensor we were modeling. An abbreviated example of the ground data matrix that we utilized in Excel®, is shown below. A similar table was used for the aerial data.

Table 3. Abbreviated multiple logistics regression data matrix

	Type Vehicle			Sensor Type		
Xrange	**FistV**	**M1**	**M2**	**Therm**	**Optic**	**Ydetect**
0.8	1	0	0	1	0	0.9988
3.2	1	0	0	1	0	0.86
6.4	1	0	0	1	0	0.315
0.8	0	1	0	1	0	0.9988
3.2	0	1	0	1	0	0.805
6.4	0	1	0	1	0	0.275
0.8	0	0	1	1	0	0.9988
3.2	0	0	1	1	0	0.86
6.4	0	0	1	1	0	0.315
1.1	1	0	0	0	1	0.9998
4.4	1	0	0	0	1	0.95
8.8	1	0	0	0	1	0.45
1.1	0	1	0	0	1	0.9998
4.4	0	1	0	0	1	0.999
8.8	0	1	0	0	1	0.75
1.1	0	0	1	0	1	0.9998
4.4	0	0	1	0	1	0.999
8.8	0	0	1	0	1	0.75

V. THE METHOD OF LOGISTICS REGRESSION

A. General.

Using a transformation function, $T(P_D)$, we transformed the original dependent data (in our case P_D, which is bounded by [0,1]) into the Real line. We conducted multiple linear regression of the transformed values upon independent variables such as Range and Range^2 which determines a linear prediction function, $\hat{T}(P_D) = g(x)$ when x is the vector of independent variables. This relation is re-transformed through T^{-1} to obtain the prediction relation in the original space [0,1], $\hat{P}_D = T^{-1}(g(x))$. From here on we will describe the dependent data, P_D, as y and the range data as x.

B. Choosing a logistics regression model.

The most difficult part of the process of fitting a model to the Janus V&V curve data was determining which logistic model to utilize. We experimented with three different types of logistic transformation functions shown as follows.

The first is named the logit transformation [8], which is defined as:

$$T_1(y) = \ln\left(\frac{y}{1-y}\right), \text{ with the associated inverse transformation; } \pi_1(g(x)) = \frac{e^{(g(x))}}{1+e^{(g(x))}}.$$

Where $g(x) = \alpha + \beta_1 x + \beta_2 x + \cdots + \beta_6 x$ from our tabled data above, and:

α is the y-intercept in the transformed space.
β_1 is the coefficient multiplied by the data in the Xrange Column.
β_2 is the coefficient multiplied by the 1 or 0 in the FistV Column.
$\vdots$
β_6 is the coefficient multiplied by the 1 or 0 in the Optic Column.

The second was the loglog transformation [1], defined as:

$$T_2(y) = \ln(-\ln(1-y)), \text{ with the associated inverse transformation; } \pi_2(g(x)) = 1 - e^{-e^{(g(x))}}.$$

And finally, the probit model [1], which utilizes the standard normal probability function, defined by:

$$T_3(y) = \Phi^{-1}\left[\frac{(y-\mu)}{\sigma}\right], \text{ and associated inverse transformation: } \pi_3(g(x)) = \Phi\left[\frac{g(x)-\mu}{\sigma}\right].$$

Again $g(x) = \alpha + \beta_1 x + \beta_2 x + \cdots + \beta_6 x$ with corresponding coefficients for the loglog model.

A successful fit of one of these models would allow us to employ an algorithm with two lines of code that only requires an identified vehicle-sensor(ground and aerial), and a specified range to produce an estimated P_D. Each of these transformations were modeled in Excel® and compared against one another.

C. Comparison of the Models.

To compare the three models, we conducted 4^{th} order fits on all vehicle/sensor pairs against the T80. This means that we continued with the same data set as shown above, but added the variables of $Xrange^2$, $Xrange^3$ and $Xrange^4$ to the model. The goal was to find one model that adequately fit the ground based sensors and one that adequately fit the aerial based sensor.

The multiple logistics regression provided the square of the correlation coefficient, R^2 value for evaluation of how well the model fit in the transformed space, but not the original data space (1,0). Therefore we manually computed an R^2 value associated with the original data and the final predicted P_D, from the logistics models using the formula:

$$R^2 = \frac{\sum (y-\bar{y})^2 - \sum (y-\hat{y})^2}{\sum (y-\bar{y})^2};$$

where y is the original P_D, $\bar{y}$ is the mean of the original P_D, and $\hat{y}$ is the Predicted P_D. This value of R^2 was a discriminating factor in which of the three models we chose for each data set (ground and aerial). The R^2 values associated with these 4th order models are depicted in the table below, with the most favorable R^2 value for each model and vehicle sensor pair in bold print.

Table 4. R^2 values for 4th order multiple logistic regression models, ground systems.

R^2 VALUE; GROUND SYSTEMS VS. T80					R^2 VALUE; GROUND SYSTEMS VS. T80			
	Thermal					Optical		
	FistV	M1	M2			FistV	M1	M2
Logit 4th	.9726	0.9427	0.9359		Logit 4th	0.4687	**0.8122**	**0.7965**
Loglog 4th	**.9844**	**0.9786**	**0.9759**		Loglog 4th	0.9246	0.5866	-0.508
Probit 4th	0.9842	0.9772	0.9745		Probit 4th	**0.9271**	0.5773	-0.556

Table 5. R^2 values for 4th order multiple logistic regression models, aerial systems.

R^2 VALUE; AERIAL SYSTEMS VS. T80				R^2 VALUE; AERIAL SYSTEMS VS. T80		
Thermal				FLIR		
	AH-64	OH-58D			AH-64	OH-58D
Logit 4th	*0.98252*	*0.97267*		Logit 4th	*0.99839*	0.91197
Loglog 4th	0.94115	0.90427		Loglog 4th	0.92712	**0.9314**
Probit 4th	**0.98921**	**0.99023**		Probit 4th	**0.9987**	0.9254

During this process we discovered two factors affecting which model we would select.

1) The first was the characteristics of the normal distribution function. Although the probit model would work best for the aerial sensors, coding the normal distribution function and its inverse into an algorithm would require another approximation function, since no closed form solution exists for the normal distribution. The probit model was therefore eliminated as an option for our model. Thus we had to replace the probit with the logit model (R^2 values italicized in table).

2) The second factor of concern was that all of the models adequately fit the primary (Thermal) sensors, for each ground and aerial vehicle, but not the secondary sensors (Optical and FLIR, respectively) especially for the ground optical sensor. The lack of fit of these logistic regression models suggested that we needed to utilize a separate model for the secondary ground sensors. However, further research into the Janus modeling system revealed that the search algorithm in Janus switches every twenty (20) seconds from the primary to secondary scanning sensors, until a detection occurs [5]. It would be difficult to determine from the Janus data base which sensor, primary or secondary, made the initial detection of an enemy. Therefore, we favored utilizing the model which best fit the primary sensor. Further evaluation of the tabled data shows the best fit for the Ground vehicle's Primary (Thermal) Sensor is the Loglog fit and the Secondary (Optical) is either the Probit, which we eliminated, or the Logit models. The best fit for the Aerial Primary (Thermal) Sensors was the Probit model, again eliminated, and the Probit and Loglog for the Secondary (FLIR) Sensors. Since the Probit model was eliminated for the AH-64 Secondary Sensor, we defer to the next higher R^2 value, which is the Logit model. Also, since the Logit model fit the AH-64, and the Loglog fit the OH-58D best, we deferred to the AH-64 fit since that vehicle is a killer system, and the OH-58D is a supporting system.

Based on the R^2 tables and the discussion above, we chose to utilize the Loglog model for the ground sensors and the Logit model for the aerial sensor.

D. Limiting the complexity of the models.

We needed to determine if we could use a less complex model, i.e. a 3rd order or even 2nd order model. We desire to use the least computationally expensive algorithm possible. We wanted to reduce the complexity of our algorithm without losing an excessive amount of accuracy. We took the R^2 values of the 4th order models and compared them to the R^2 values of the 3rd order model, and likewise, 3rd to 2nd order. From that evaluation we determined the percent gain of using a more complex model. If the percent gain did not appear significant, say less than one percent, we decided not to use the more complex model. The comparisons of the R^2 values are shown below.

Table 6. Comparison of R^2 values for loglog multiple logistic regression model, ground.

	R^2 Values for LogLog Model, Ground Vehicle Sensors (Thermal)						
	2nd Order	3rd Order	Delta2 - 3	% Gain	4th Order	Delta3 - 4	% Gain
FistV	0.984421	0.98508	0.000659	0.067%	0.98442	-0.00066	-0.067%
M1	0.97867	0.97824	-0.00043	-0.044%	0.97867	0.00043	0.044%
M2	0.97596	0.97867	0.00271	0.278%	0.97596	-0.00271	-0.277%

Table 7. Comparison of R^2 values for logit multiple logistic regression model, aerial.

	R^2 Values for Logit Model, Aerial Sensors (Thermal)						
	2nd Order	3rd Order	Delta2 - 3	% Gain	4th Order	Delta3 - 4	% Gain
AH-64	0.929474	0.981245	0.051772	5.570%	0.981362	0.000117	0.012%
OH-58D	0.953021	0.965042	0.012021	1.261%	0.972673	0.007631	0.791%

These tables show that the ground vehicle sensors can be adequately modeled using a 2nd order Loglog model. This is because the greatest percent gain in accuracy we can achieve by using a 3rd order model is 0.278%, which, for our purposes, was not large enough to warrant increased complexity. Also we see that a 3rd order Logit model is accurate enough because the greatest percent gain in accuracy we can achieve by using a 4th order model is 0.791%, which again, is not large enough to increase the complexity of our algorithm. Thus the models we settled upon are the 2nd order Loglog multiple logistic regression model for the ground vehicle sensors and the 3rd order Logit logistic multiple regression model for the aerial sensors.

E. The models.

1. Ground Vehicle Sensors.

The 2nd order Loglog regression model for ground vehicle sensors:

$T(y) = \ln(-\ln(1-y))$, with the associated inverse transformation; $\pi(g(x)) = 1 - e^{-e^{(g(x))}}$,

Where $g(x) = \alpha + \beta_1 x_1 + \beta_2 x_1^2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \beta_6 x_6 + \beta_7 x_7$, and:

α is the y-intercept in the transformed space.
β_1 is the coefficient multiplied by the data in the Xrange Column.
β_2 is the coefficient multiplied by the data in the Xrange^2 Column.
β_3 is the coefficient multiplied by the 1 or 0 in the FistV Column.
β_4 is the coefficient multiplied by the 1 or 0 in the M1 Column.

β_5 is the coefficient multiplied by the 1 or 0 in M2 Column.
β_6 is the coefficient multiplied by the 1 or 0 in the Thermal Column.
β_7 is the coefficient multiplied by the 1 or 0 in Optical Column.

The final equation for $g(x)$ will look like:

$$g(x) = 16.244877 - 0.451966x_1 + 0.0066014x_1^2 - 0.313104x_3 - 0.2118893x_4 + 0x_5 \\ - 14.058842x_6 - 12.466667x_7$$

which will be coded into the detection algorithm along with $\pi(g(x)) = 1 - e^{-e^{(g(x))}}$, which will predict our probability of detection, P_D, for ground vehicle sensors.

The three graphs of our predicted curves and the original data curves for each ground sensor (FistV, M1, M2) are shown pictorially below.

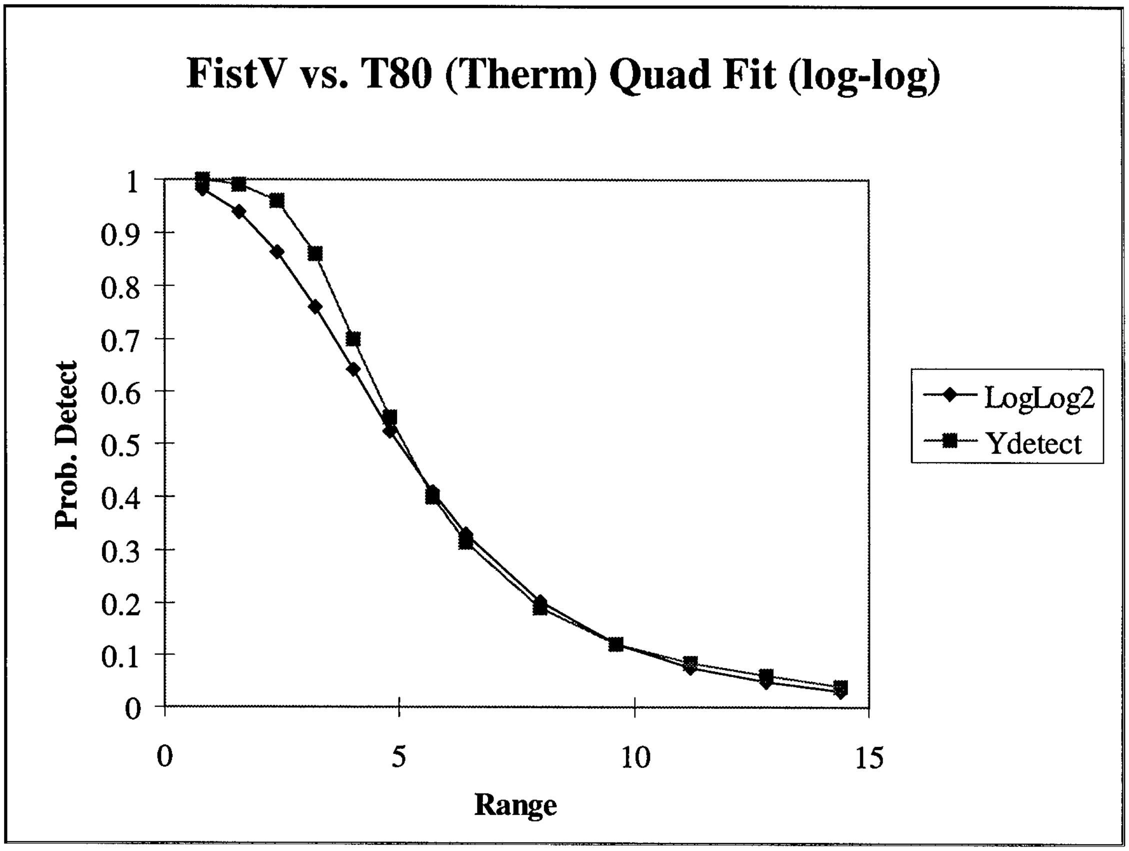

Figure 2. Graphic comparison of Loglog quadratic fit to original data, (FistV (therm) vs. T80).

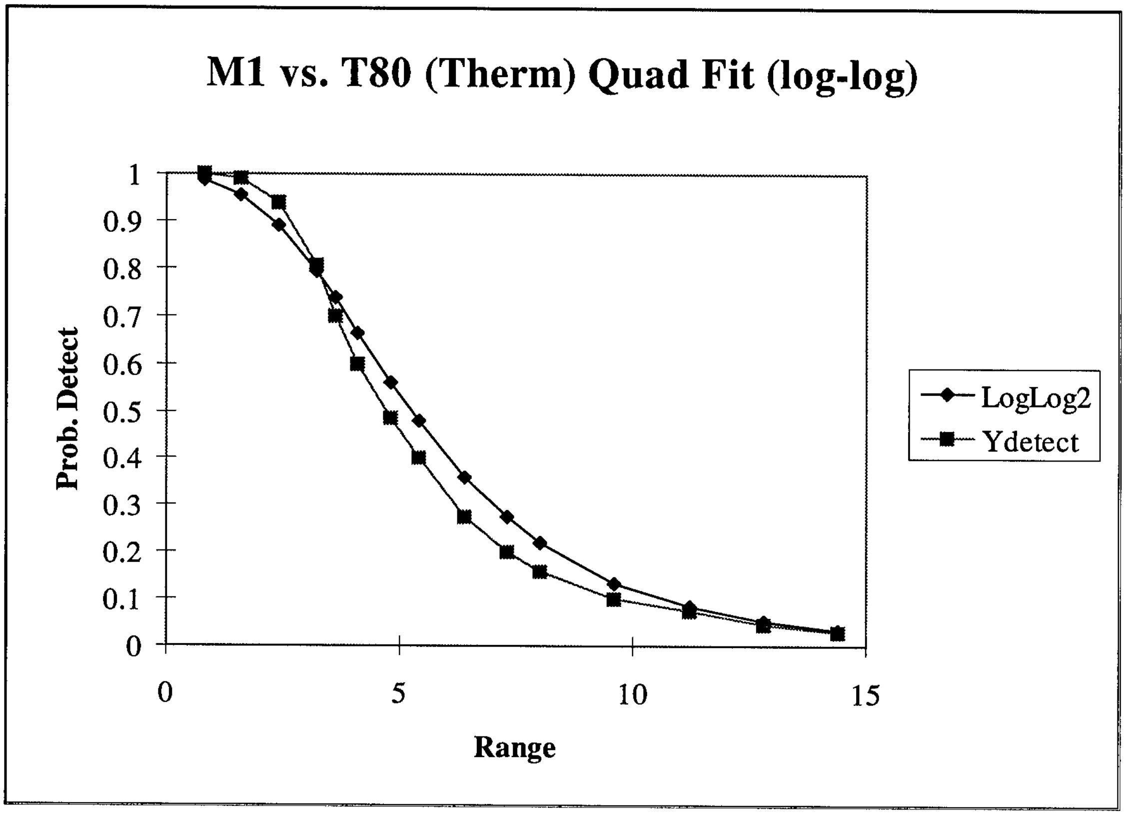

Figure 3. Graphic comparison of Loglog quadratic fit to original data, (M1 (therm) vs. T80).

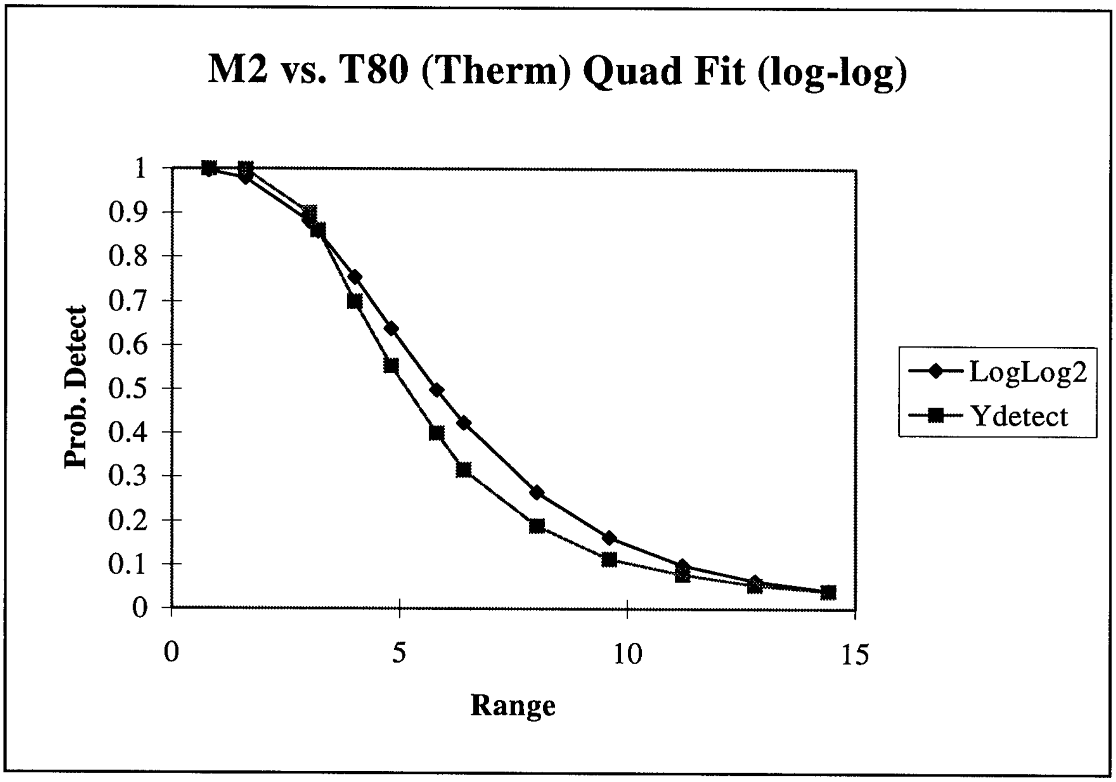

Figure 4. Graphic comparison of Loglog quadratic fit to original data, (M2 (therm) vs. T80).

2. Aerial Sensors.

The 3rd order Logit regression model for aerial sensors:

$T(y) = \ln\left(\frac{y}{1-y}\right)$, with the associated inverse transformation; $\pi(g(x)) = \frac{e^{(g(x))}}{1+e^{(g(x))}}$,

Where $g(x) = \alpha + \beta_1 x_1 + \beta_2 x_1^2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \beta_6 x_6 + \beta_7 x_7$, and:

α is the y-intercept in the transformed space.
β_1 is the coefficient multiplied by the data in the Xrange Column.
β_2 is the coefficient multiplied by the data in the Xrange^2 Column.
β_3 is the coefficient multiplied by the 1 or 0 in the Xrange^3 Column.
β_4 is the coefficient multiplied by the 1 or 0 in the AH-64 Column.
β_5 is the coefficient multiplied by the 1 or 0 in OH-58D Column
β_6 is the coefficient multiplied by the 1 or 0 in the Thermal Column.
β_7 is the coefficient multiplied by the 1 or 0 in Flir Column.

The final equation for $g(x)$ will look like:

$$g(x) = 866.981162 - 2.254311x_1 + 0.136956x_1^2 - 0.002840x_1^3 - 788.398867x_4 - 789.333333x_5 - 67.706670x_6 - 66.928571x_7$$

which will be coded into the detection algorithm along with $\pi(g(x)) = \frac{e^{(g(x))}}{1+e^{(g(x))}}$, which will predict our probability of detection, P_D, for aerial sensors.

The two graphs of our predicted curves and the original data curves for each aerial sensor (AH-64 and OH-58D) are shown pictorially below.

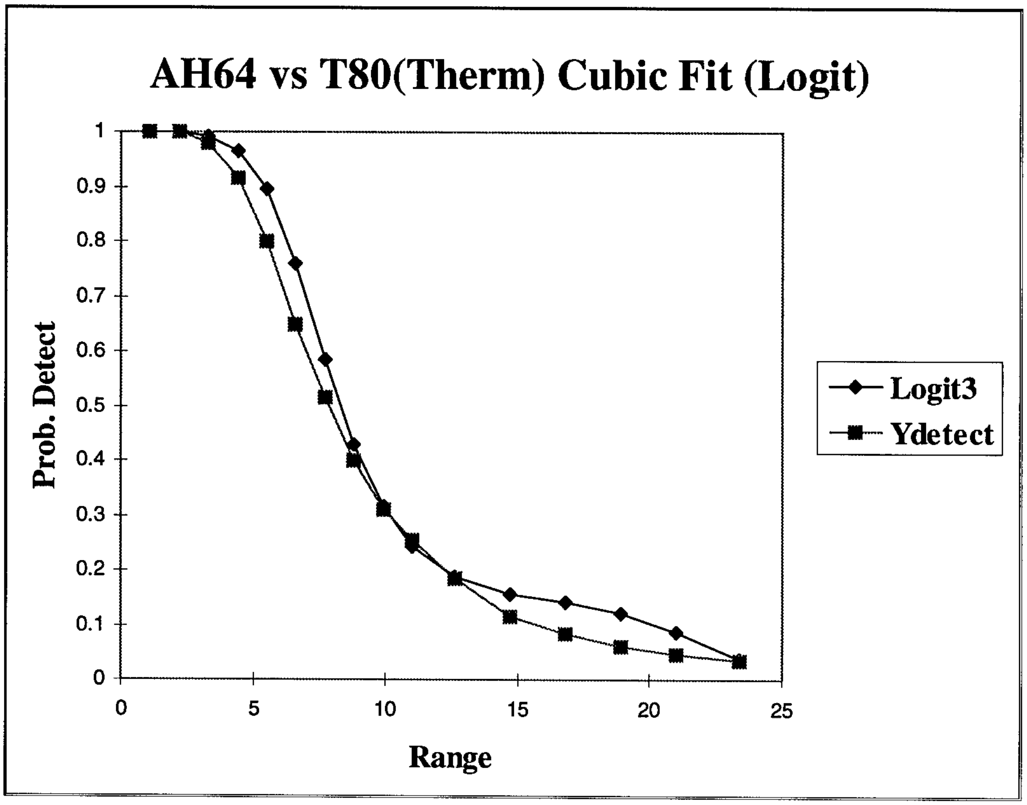

Figure 5. Graphic comparison of Logit cubic fit to original data, (AH-64 (therm) vs. T80).

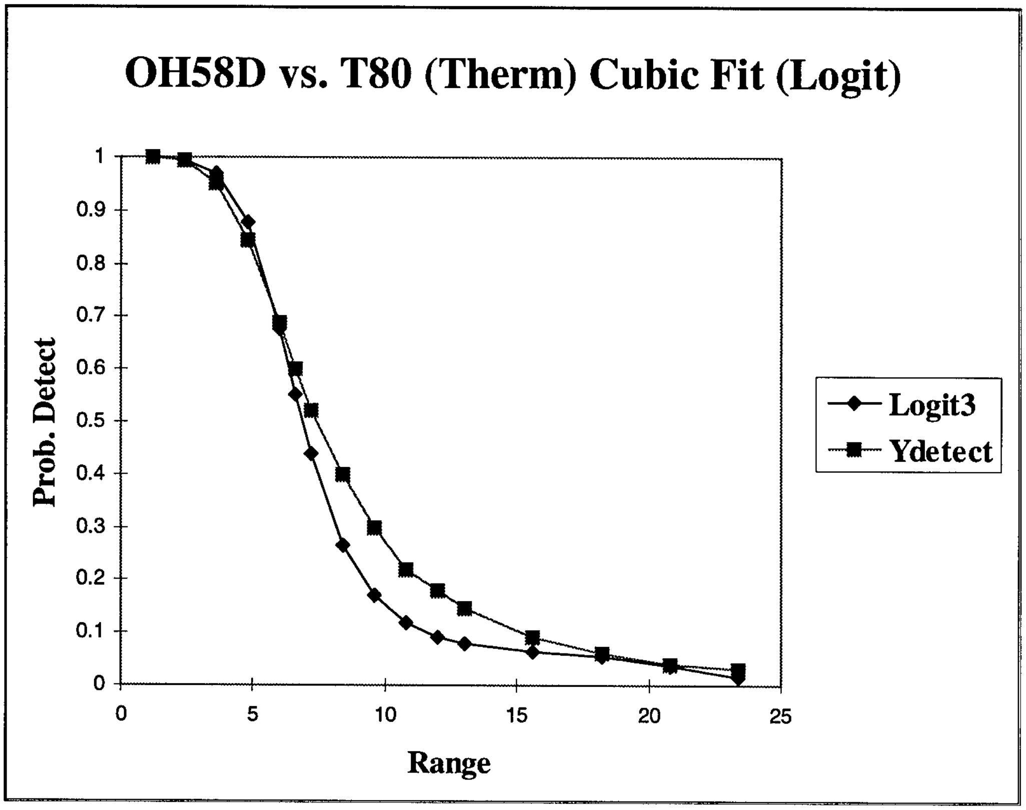

Figure 6. Graphic comparison of Logit cubic fit to original data, (OH-58D (therm) vs. T80).

VI. CONCLUSIONS

The data extracted from the Janus V&V screens is an estimate of what the screens depict. These screens simplify the actual P_D algorithm in Janus by reducing the P_D to a function of range, given a friendly sensor and an enemy target. We have simplified the model further by singularly evaluating a P_D curve of friendly vehicles against a T80 tank. We did this because we would have no idea what type of enemy the friendly vehicle would encounter, thus we assumed that all friendly forces would look for the most dangerous ground force enemy, a T80 tank.

The logistics regression models provide an accurate model of the physical nature of detection. The chosen Loglog and Logit models provide an adequate fit to the data provided by the Janus V&V curves. Multiple logistics regression is a powerful model to use in analyzing the data because 1) it provides output in the [0,1] space, and 2) we can use one model that fits the three ground systems and one model that fits the two aerial systems. This can be accomplished with reasonably conservative computation time and data storage.

APPENDIX A. DATA EXTRACTED FROM JANUS V&V CURVES.

The following tables represent the extracted data of ordered pairs (range, P_D) from the Janus Validation and Verification graphs produced for an observer-target pair for each type of sensor. See figure 1 and Table 1.

The heading consists of type of friendly vehicle with type of sensor against a stationary enemy T80. The left hand column is the sensor's range and the right hand column is the probability of detection P_D.

1. Ground Vehicle Sensor Data vs. T80.

a) Primary Sensors.

FistV vs. T80 (Therm)	
Xrange	**Ydetect**
0.8	0.9988
1.6	0.99
2.4	0.96
3.2	0.86
4.0	0.7
4.8	0.55
5.7	0.4
6.4	0.315
8.0	0.19
9.6	0.12
11.2	0.085
12.8	0.06

M1 vs. T80 (Therm)	
Xrange	**Ydetect**
0.8	0.9988
1.6	0.99
2.4	0.94
3.2	0.805
3.6	0.7
4.1	0.6
4.8	0.485
5.4	0.4
6.4	0.275
7.3	0.2
8.0	0.16
9.6	0.1
11.2	0.075
12.8	0.045
14.4	0.03

M2 vs. T80 (Therm)	
Xrange	**Ydetect**
0.8	0.9998
1.6	0.999
3.0	0.9
3.2	0.86
4.0	0.7
4.8	0.555
5.8	0.4
6.4	0.315
8.0	0.19
9.6	0.115
11.2	0.08
12.8	0.055
14.4	0.04

b) Secondary Sensors.

FistV vs. T80(Optical	
Xrange	Ydetect
1.1	0.9998
2.2	0.9997
3.3	0.999
4.4	0.95
5.5	0.93
6.6	0.8
7.7	0.62
8.8	0.45
9.9	0.31

M1 vs. T80 (Optical)	
Xrange	Ydetect
1.1	0.9998
2.2	0.9997
3.3	0.9996
4.4	0.999
5.5	0.995
6.6	0.95
7.7	0.875
8.8	0.75
9.9	0.61

M2 vs. T80 (Optical)	
Xrange	Ydetect
1.1	0.9998
2.2	0.9997
3.3	0.9996
4.4	0.9995
5.5	0.999
6.6	0.98
7.7	0.94
8.8	0.86
9.9	0.75

2. Aerial Sensor Data vs. T80.

a) Primary Sensors.

AH-64 vs. T80 (Therm)	
Xrange	Ydetect
1.1	0.9998
2.2	0.999
3.3	0.98
4.4	0.915
5.5	0.799
6.6	0.65
7.7	0.515
8.8	0.4
9.9	0.31
11	0.255
12.6	0.185
14.7	0.115
16.8	0.085
18.9	0.06
21	0.045
23.4	0.035

OH-58D vs. T80 (Therm)	
Xrange	Ydetect
1.2	0.9998
2.4	0.995
3.6	0.95
4.8	0.845
6	0.69
6.6	0.6
7.2	0.52
8.4	0.399
9.6	0.299
10.8	0.22
12	0.18
13	0.145
15.6	0.09
18.2	0.06
20.8	0.04
23.4	0.03

a) Secondary Sensors.

AH-64 vs. T80 (Flir)	
Xrange	Ydetect
0.5	0.9999
1	0.9998
1.5	0.9998
2	0.9997
2.5	0.9997
3	0.9996
3.5	0. 9996
4	0.999
4.5	0.98

OH-58D vs. T80 (Flir)	
Xrange	Ydetect
0.5	0.9998
1	0.9998
1.5	0.9997
2	0.9997
2.5	0.9996
3	0.999
3.5	0.975
4	0.85
4.1	0.8
4.3	0.7
4.5	0.595

APPENDIX B. OTHER MODELS.

This appendix briefly explains other modeling techniques that we investigated for use in modeling probability of detection. For various reasons described below, these methods were not accepted as viable solutions to our problem.

1. Polynomial Curve Fitting.

We utilized least squares regression on the data to determine predicted values of P_D at the given ranges. The data looked as if a cubic or fourth order polynomial would fit best. MATLAB® was utilized to perform the polynomial fitting. The cubic polynomial fit yielded the polynomial:

$$f_3(x) = .0003x^3 - .0004x^2 - .1275x + 1 \qquad \text{with an } R^2 = .8664$$

the fourth order polynomial fit yielded:

$$f_4(x) = -.0002x^4 + .0052x^3 - .0491x^2 - .0299x + 1 \qquad \text{with an } R^2 = .9886$$

The equation $f_4(x)$ provides an adequate fit to the data. However, there are areas of concern in using this function. The first concern is that just near zero, the value of the function $f_4(x)$ output would be slightly greater than one. Secondly, the function $f_4(x)$ is not monotonically decreasing after 10 kilometers in range. It is unrealistic for the P_D to begin to increase as range increases. Although these situations can easily be accounted for in an algorithm with 'conditional if' statements, we did not want to have to put a condition on each of the possible polynomial models used in our algorithm.

2. Cubic Spline Fitting.

The second method we used to model the P_D was cubic spline fitting. This method interpolates between data points using cubic spline fits. This turned out to be a powerful tool since the spline fit passes through each of the (*x*, *y*) data points. Yet again, the phenomena existed such that around 1.5 to 2 kilometers in range, the modeled value for P_D is greater than one. The other problem with the cubic spline model fit is that it is not a function, but a set of data points represented in a table. This type of output would require us to store a data set for each sensor / vehicle pair, and then enter into the data table with a search algorithm to determine the P_D.

3. Inverse Function Fitting.

Similar to the above methods, we attempted to fit inverse functions to the data. These functions provided S shaped curves that are bound by [0, 1] and closely resembled the data. However, these functions had to be physically manipulated in order to provide the shape of each different sensor-target pair. As an example, we fit the Probability of Detection of an M1 vs. a T80 using a thermal sight with the function:

$$f_I(x) = \frac{15}{(x^2 + 15)},$$

where the number 15 was selected off the Janus graph as a range value where P_D is near zero. This function fit the data quite well, with an $R^2 = .9244$. This method was not selected because a trial and error method is required to determine which inverse function accurately fits the data.

REFERENCES

[1] Agresti, A. Categorical Data Analysis, John Wiley & Sons, Inc., 1990.

[2] Barr, D. and Sherrill, E. T., Estimating the Operational Value of Tactical Information, US Military Academy, ORCEN Technical Report, 1995.

[3] Barr, D. and Sherrill, E. T., Exploring a Relationship Between Tactical Intelligence and Battle Results, paper presented at 63rd Military Operations Research Symposium, Annapolis, MD, 1995.

[4] Barr, D. and Sherrill, E. T., Measuring Information Gain in Tactical Operations, US Military Academy, ORCEN Technical Report, 1996.

[5] Data Base Manager's Manual, Janus 3.0, Department of the Army, Headquarters TRADOC Analysis Center, Ft. Leavenworth, KS, Revised 1993.

[6] Data Base Manager's Manual, Janus 6.0, Department of the Army, Headquarters TRADOC Analysis Center, Ft. Leavenworth, KS, 1995.

[7] Finney, R.L. and Thomas, G.B. Jr., Calculus, Addison-Wesley Publishing Company, 1992.

[8] Hosmer, D.W. Jr. and Lemeshow, S., Applied Logistic Regression, John Wiley & Sons, Inc., 1989.

[9] MatLab Reference Guide, The MathWorks, Inc., 1995.

[10] Memorandum for Record, NVEOD Detection in Janus (A), Department of the Army, TRADOC Analysis Center, White Sands Missile Range, NM, 1992.

[11] Myers, R.H. and Walpole, R.E., Probability and Statistics for Engineers and Scientists, MacMillan Publishing Co., Inc., 1978.

[12] White Paper, Target Acquisition in Janus Army, Department of the Army, TRADOC Analysis Center, White Sands Missile Range, NM, date unknown.