SATURN V PRELAUNCH SYSTEMS SIMULATION MODEL FOR A LAUNCH OPPORTUNITY CONTAINING MULTIPLE LAUNCH WINDOWS

Bernard J. Schroer
Aerospace Group Teleservices
The Boeing Company
Huntsville, Alabama

Abstract

This paper presents the results of the Saturn V Prelaunch Systems Simulation Model for a launch opportunity containing multiple launch windows. The simulation consisted of the events in operation during the last 26 hours of the Saturn V countdown (i.e., T-26 hours to liftoff). The measure of effectiveness, Launch Vehicle Availability (LVA), was evaluated for various launch window configurations.

INTRODUCTION

The Saturn V Prelaunch Systems Simulation Model for a launch opportunity containing multiple launch windows was developed within The Boeing Company, Huntsville, Alabama, under NASA Contract NAS8-5608, Task 3.2. Use of the model was completed in June, 1968. The requirements for the model evolved from the uncertainty associated with determining the optimum launch opportunity in terms of probability of launching the Saturn V vehicle within a launch opportunity. Although many organizations had their opinions of the optimum launch window spacings, no analytical tool existed to support these opinions. Previous studies (Reference 1 and 2) had resulted in a model of the Saturn V prelaunch for a single launch window. Therefore, the model presented in this paper was an extension of this initial model to include a launch opportunity of multiple windows. Included in this paper is a description of the simulation model, the model inputs, the selected test cases, the model outputs, and the conclusions.

MODEL DESCRIPTION

For discussion purposes, the following areas of the model will be presented:

1. Model logic.
2. Event Simulation.
3. Measure of Effectiveness.
4. Supporting Organizations.
5. Model Languages.

Model Logic

The model can be considered as being composed of two distinct sections: a main sequence and a recycle sequence. The main sequence consisted of those events in the planned countdown which operate between T-26 and T-0. The recycle sequence consisted of (1) one of various backout sequences consisting of those events required to return the countdown to some preceding point, and (2) a recycle hold consisting of those events required to sustain the vehicle status at a particular count time. A generalized flow diagram of the model is presented in Figure 1.

When a main sequence failure resulted in a scrub (i.e., accumulated delay precluded launching within the window), a backout sequence was selected. The selection was dependent on the vehicle status at the time of the scrub. Failures which caused a scrub were repaired when cryogenic detanking and purging were complete. There were eleven different backout sequences. Primarily, vehicle batteries constraints required that all backout sequences recycled the countdown to T-25.

After the backout sequence was completed, the vehicle was retained in a hold state until both of the following conditions were mutually satisfied: (1) all maintenance was completed, and (2) the remaining time to the end of the next window was (a) greater than the time remaining in the main sequence (i.e., T-25 to T-0), and (b) less than the sum of the time remaining in the main sequence and of the length of the window which was being attempted. When these conditions were mutually satis-

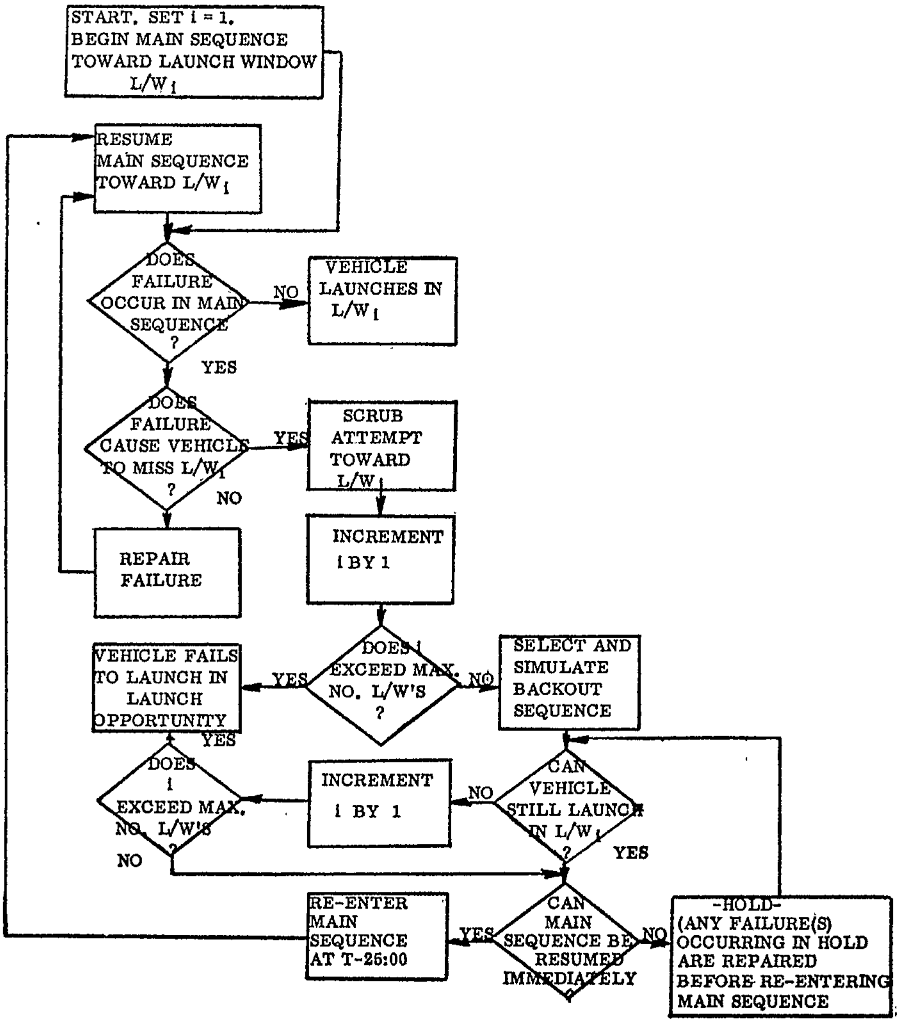


FIGURE 1. GENERALIZED MODEL LOGIC

isfied, the hold was discontinued and the main sequence was resumed at T-25 toward the next window. During the recycle hold only those events which were required to sustain the vehicle continued to operate. If equipment failed during the hold, the failures were repaired immediately.

The three classes of anomalies which were simulated by the model in an attempted launch are:

1. Failures which caused a hold but not a recycle.
2. Failures which caused a recycle but not a scrub.
3. Failures which caused a scrub and required vehicle rescheduling to attempt launching in a subsequent window.

Each of these classes is described pictorially in Figure 2.

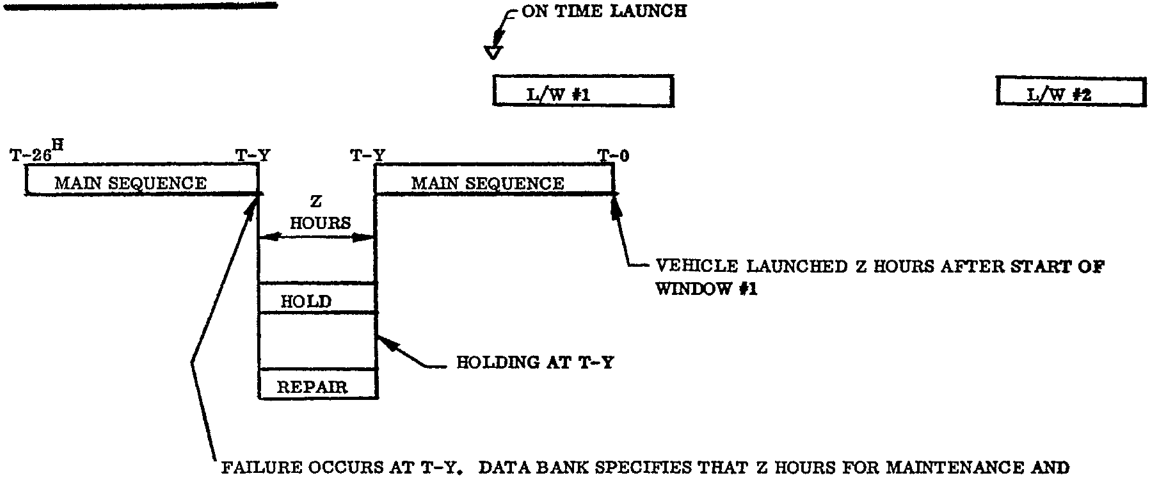
Event Simulation

The actual simulation of the main and recycle sequences was simulated by the model on the event level, where an event may consist of more than one actual item of equipment. It was assumed that all event failures follow the Poisson process. That is, the occurrence of failures was completely random and independent. Therefore, from the Poisson distribution

$$f_n(T) = \frac{(\Delta T)^n e^{-\Delta T}}{n\,!},$$

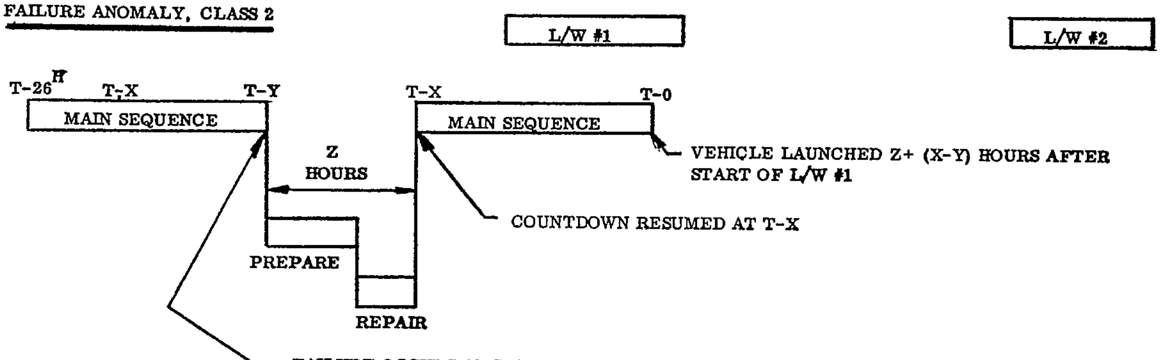
the mean time between failures can be described by the

FAILURE ANOMALY, CLASS 1



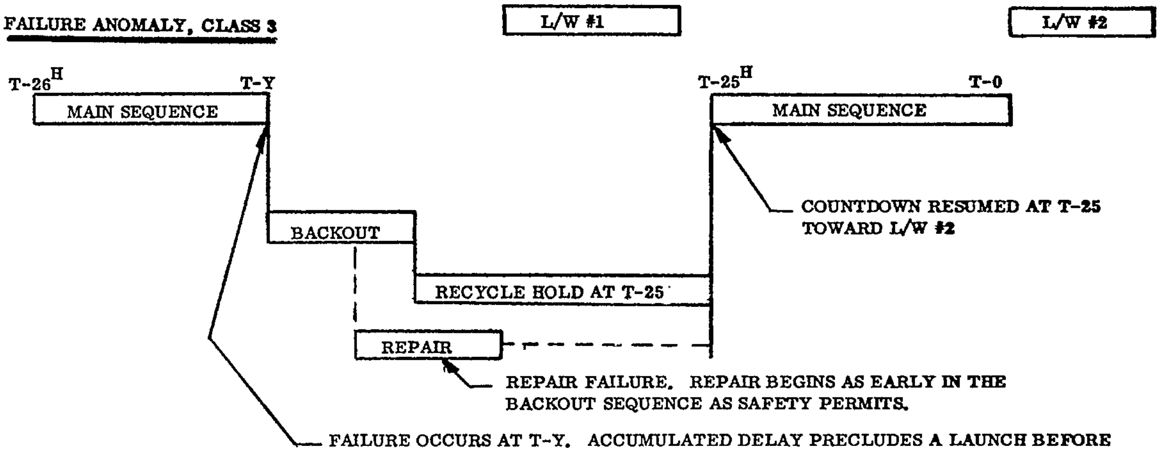
FAILURE OCCURS AT T-Y. DATA BANK SPECIFIES THAT Z HOURS FOR MAINTENANCE AND THAT RECYCLE IS NOT REQUIRED. SINCE IT IS STILL POSSIBLE TO MAKE L/W #1, A SCRUB IS NOT INITIATED.

FAILURE ANOMALY, CLASS 2



FAILURE OCCURS AT T-Y. DATA BANK SPECIFIES Z HOURS FOR MAINTENANCE AND THAT MAIN SEQUENCE MUST BE RE-ENTERED AT T-X. SINCE IT IS STILL POSSIBLE TO MAKE L/W #1, A SCRUB IS NOT CALLED.

FAILURE ANOMALY, CLASS 3



REPAIR FAILURE. REPAIR BEGINS AS EARLY IN THE BACKOUT SEQUENCE AS SAFETY PERMITS.

FAILURE OCCURS AT T-Y. ACCUMULATED DELAY PRECLUDES A LAUNCH BEFORE THE END OF L/W #1, THEREFORE, THE LAUNCH ATTEMPT TOWARD WINDOW #1 IS SCRUBBED.

MODEL LOGIC INDICATES THE VEHICLE CANNOT HOLD AT T-Y AND WAIT FOR L/W #1. THE MODEL LOGIC REQUIRES THAT THE COUNT BE RECYCLED TO T-25 IN ORDER TO TRY FOR L/W #2.

FIGURE 2. CLASSES OF ANOMALIES

negative exponential distribution

$$f(t) = \frac{1}{\Delta T} e^{-t/\Delta T} ,$$

where T is the mean number of failures. ΔT also equals $\Sigma\lambda_i t_i$, where λ_i is the failure rate of the i^{th} item of equipment and t_i is the operating time of the i^{th} item of equipment.

The probability of a failure during the event duration is determined by

$$F(t) = \int_0^t f(t)dt = 1 - e^{-t/\Delta T}$$

F(t) is a value between 0.0 and 1.0. Therefore, solving for the time to failure of the event gives

$$t = \frac{1}{\Lambda} \ln(RN),$$

where RN is a random number between 0.0 and 1.0. By using this equation for determining the mean time to failure, the model did not require the storage of a function of the time to failure for each event. Instead, only the value of Λ for each event was stored on the event data tape.

No empirical distribution was readily available for describing the mean times to repair. Therefore, using a least square curve fitting technique, a fifth order polynomial of the form

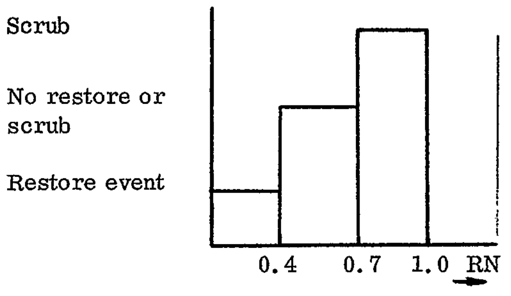
$$t = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5$$

was fitted through the sample repair time data. By initially normalizing the repair time data, a random number between 0.0 and 1.0 can be substituted for x. The repair time equation can now be rewritten as

$$t = a_0 + a_1RN + a_2RN^2 + a_3RN^3 + a_4RN^4 + a_5RN^5$$

Again the model did not require the storage of a function of the repair times for each event. Instead, only the six coefficients of the fifth order polynomial were stored for each event.

Some of the events required a mandatory recycle back to some previous completed event if a failure was simulated by the event. Failures in these events do not necessarily result in a scrub. These types of failures were identified by class 2 type of anomaly in Figure 2. The determination of the restore can best be described by the following example:



It was assumed that each event could have only one restore event. Therefore, this information was stored by restore event number and the cumulative probabilities of a recycle, no recycle, and a scrub.

Measure of Effectiveness

The measure of effectiveness used in the model was Launch Vehicle Availability (LVA). LVA was defined as the ratio of successful launches to the total number of launch attempts. Since LVA was only as good as the sample size, sampling theory related to the sampling distribution of proportions was used to determine an acceptable sample size. By trading between estimated computer run time and confidence level, a sample size of 200 launch attempts was selected which permitted a 90 percent confidence level that the LVA based on the simulation results was the true LVA.

Supporting Organizations

An overall picture of the organizations involved in the development and support of the model is presented in Figure 3. Inputs in the form of event data were provided by reliability and maintenance engineering. Inputs in the form of event sequences diagrams (i.e., main and recycle sequences) were provided by systems operations. The computer programming group maintained the data bank and also wrote the special programs in support of the data bank and the modeling group.

The systems engineering modeling group did the model development. An IBM 2250-3 graphics terminal was used for model checkout. The graphics capability allowed for compiling the model, correcting diagnostics, and recompiling the model. A description of the graphics is available in Reference 3. The systems engineering analysis group performed the engineering analysis on the outputs from the data bank and the simulation model.

Model Languages

The model was written in BSS (Boeing System Simulator). BSS is an extension of GPSS-II which was modified to satisfy specific simulation requirements. Several of the extensions of GPSS-II included the addition of the following block types: ALLOT's for

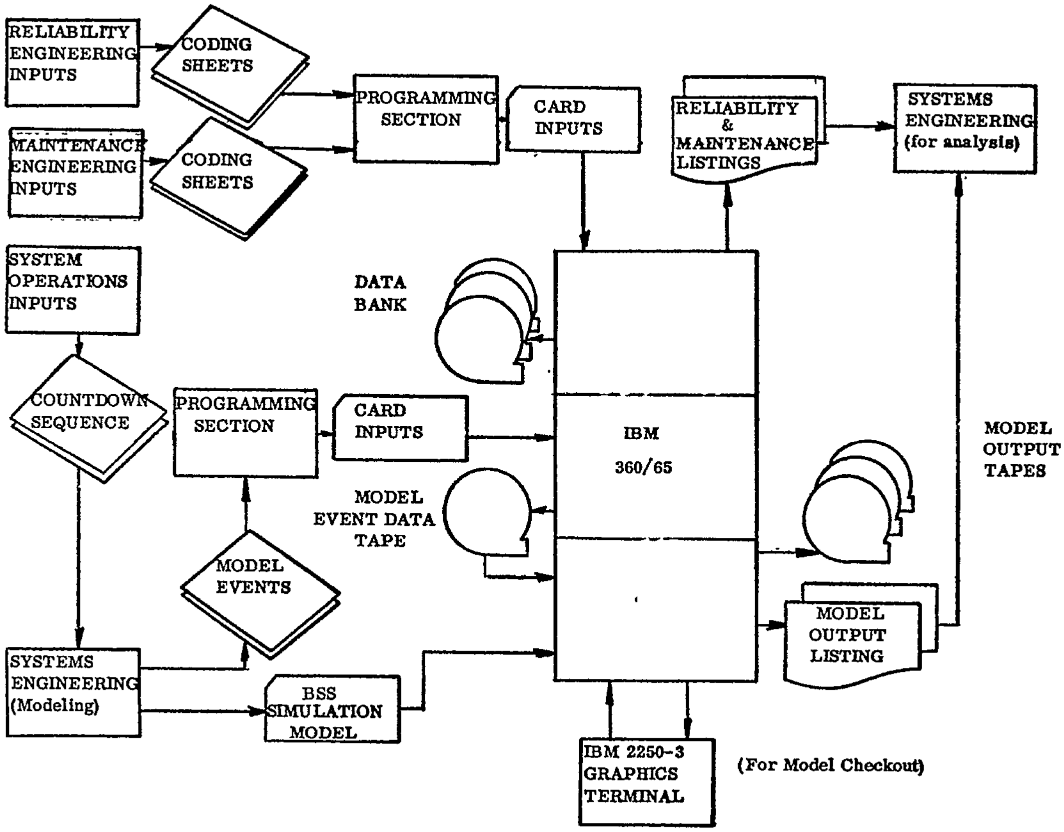


FIGURE 3. SUPPORTING ORGANIZATIONS

allocating entities, LOADX's for initializing savex locations, OUTPUT's and FORMAT's for defining specific types of outputs (used extensively during model checkout), and LINK's and UNLINK's used with user chains.

In addition to the BSS language, the model also used two HELP routines which were written in assembler language. HELP-1 was used to search the event data tape and to transfer event data, such as mean time to failure and mean time to repair, to transaction parameters within the model. HELP-2 was used to stop all support events when a failure had been simulated by the model and to selectively destroy transactions during the simulation.

The data bank supporting the model was written in COBOL and contains approximately 100,000 data elements. The data bank not only includes data on the event level, but also on all items of equipment within an event. The data bank was used extensively in generating various reports such as event and equipment rankings and operating profiles. The model used data on 260 of the events. The main sequence required data on 180 events while the backout sequences required data on 80 events.

The model consisted of 2300 BSS blocks, 100 variables, 11 macros, and 2 HELP routines. The model ran in a 250K core partition of the IBM 360/65 digital computer. For a simulation of 200 launch attempts, the execution time was 120 minutes which consists of 70 minutes of JTASK (cpu time) and 50 minutes of JWAIT (I/0 time). This run time was without the use of the LINK and UNLINK blocks.

MODEL INPUTS

Data inputs to the model were in the form of a magnetic tape (Reference Figure 3). After the model logic had been developed, a list of all events used in the model was forwarded to the programming group. This list was impacted against the data bank and an event data tape was created which contained the failure and maintenance data for each event. A typical tape record contains the event number, the failure rate λ , the coefficients of the fifth order polynomial, and the event number if a recycle but not a scrub is initiated. The data describing the launch opportunity, such as number of windows, window spacing, and window lengths, were inputted through LOADX statements.

TEST CASES

It was not feasible within the allotted time period to simulate every possible combination of window spacings. Therefore, the approach was to select a representative sample of window patterns as a baseline which could be used in evaluating similar window patterns. In selecting the test cases, it was assumed that there existed a maximum 8-day lunar launch opportunity. From this assumption, the following four general groups of launch window combinations were simulated:

1. Launch window spacing of 24 hours (a launch window spacing of xx hours is defined as xx hours from the start of one window to the start of the next window).
2. Launch window spacings of 48 hours.
3. Launch window spacings of 72 hours.
4. Launch window spacings of 24 hours between the first and second windows, 48 hours between the second and third windows, and 72 hours between the third and fourth windows.

The length of the launch windows was assumed to be either three or zero hours.

MODEL OUTPUTS

Outputs of the simulation included 15 tables relating to LVA and 3 tapes. From the 3 tapes, through small BSS programs, an additional 45 tables were generated which include failure and recycle results. These tapes were used as historical records of the model outputs and have been very useful as data sources for additional analysis. The following paragraphs present the more significant results of the model.

The cumulative LVA's for the selected window spacings are presented in Figure 4. From this figure it can be seen that LVA increased as the number of windows increased. LVA also increased as the spacing between windows increased and as the window length increased. A statistical hypothesis test was made to test for significant difference between LVA's. The null hypothesis was:

> There is no significant difference at the 10 percent level between two cumulative LVA's. That is, we are 90 percent confident that any difference is due solely to random variation.

The standardized variable $z = (P_1 - P_2) / \sigma(P_1 - P_2)$, where P_1 and P_2 are the sample proportions and $\sigma(P_1 - P_2)$ is the standard deviation of the difference of proportions, was used in the statistical test.

These statistical tests for windows of three hours indicated that there was no significant difference in LVA for:

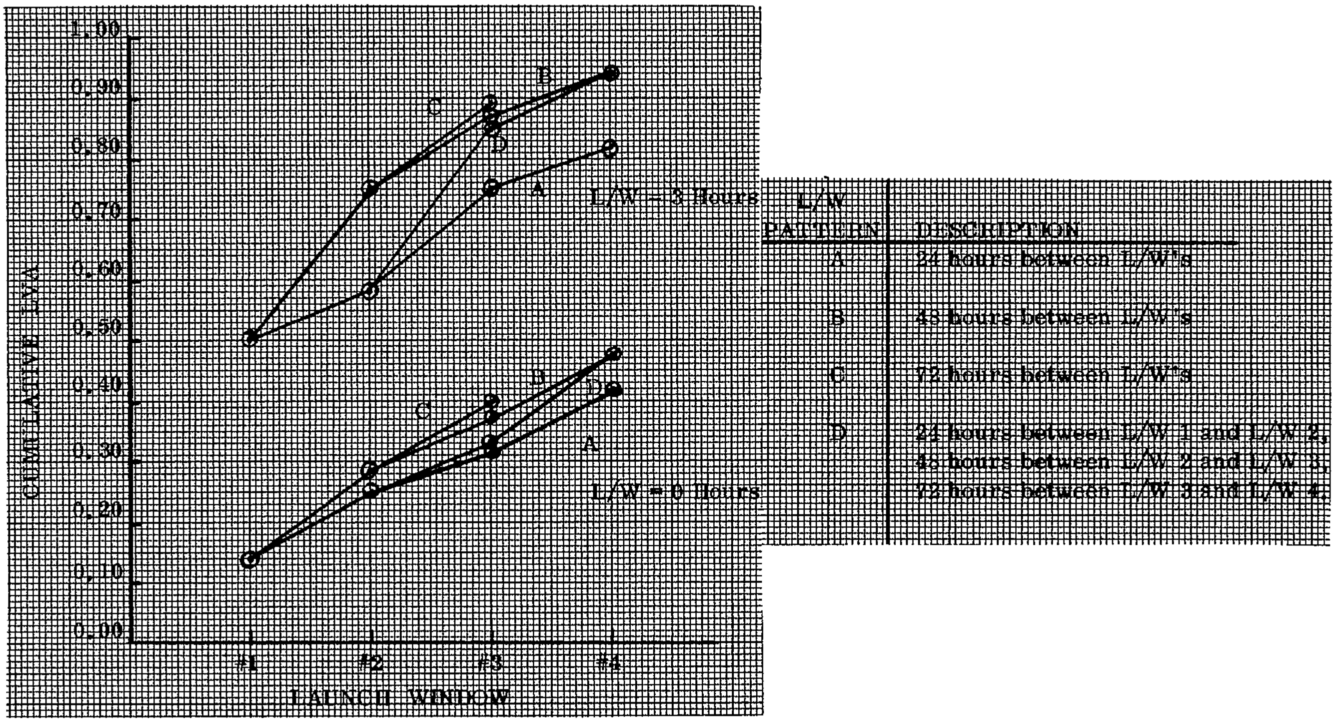
1. Two windows spaced 48 or 72 hours apart, or for three windows spaced 24 hours apart.
2. Three windows spaced 48 or 72 hours apart, or for three windows spaced 24 and 48 hours apart, respectively.
3. Four windows spaced 48 hours apart, or for four windows spaced 24, 48, and 72 hours apart, respectively.

An appreciable increase in LVA was noticed by increasing the number of windows to four; however, adding more windows after the fourth did not significantly increase LVA. Also, LVA did not significantly increase by increasing the spacing between windows beyond 72 hours.

In the previous paragraph, LVA was only compared as a function of the number of windows, window spacing, and window length. Another factor, the mean time a vehicle was on the pad before launch, can be compared with LVA. Figure 5 presents LVA for various window patterns as a function of mean pad time. Mean pad time was defined as the average elapsed time from the initial occurrence of T-26 to liftoff. As seen in Figure 4, no significant difference in LVA was noticed for two windows spaced 48 or 72 hours apart, or for three windows spaced 24 hours apart. However, if pad time were considered in selecting a window pattern, the spacing of three windows 24 hours apart or two windows 48 hours apart would minimize pad time and also not appreciably affect LVA. Similar comparisons could be made for the remaining window patterns.

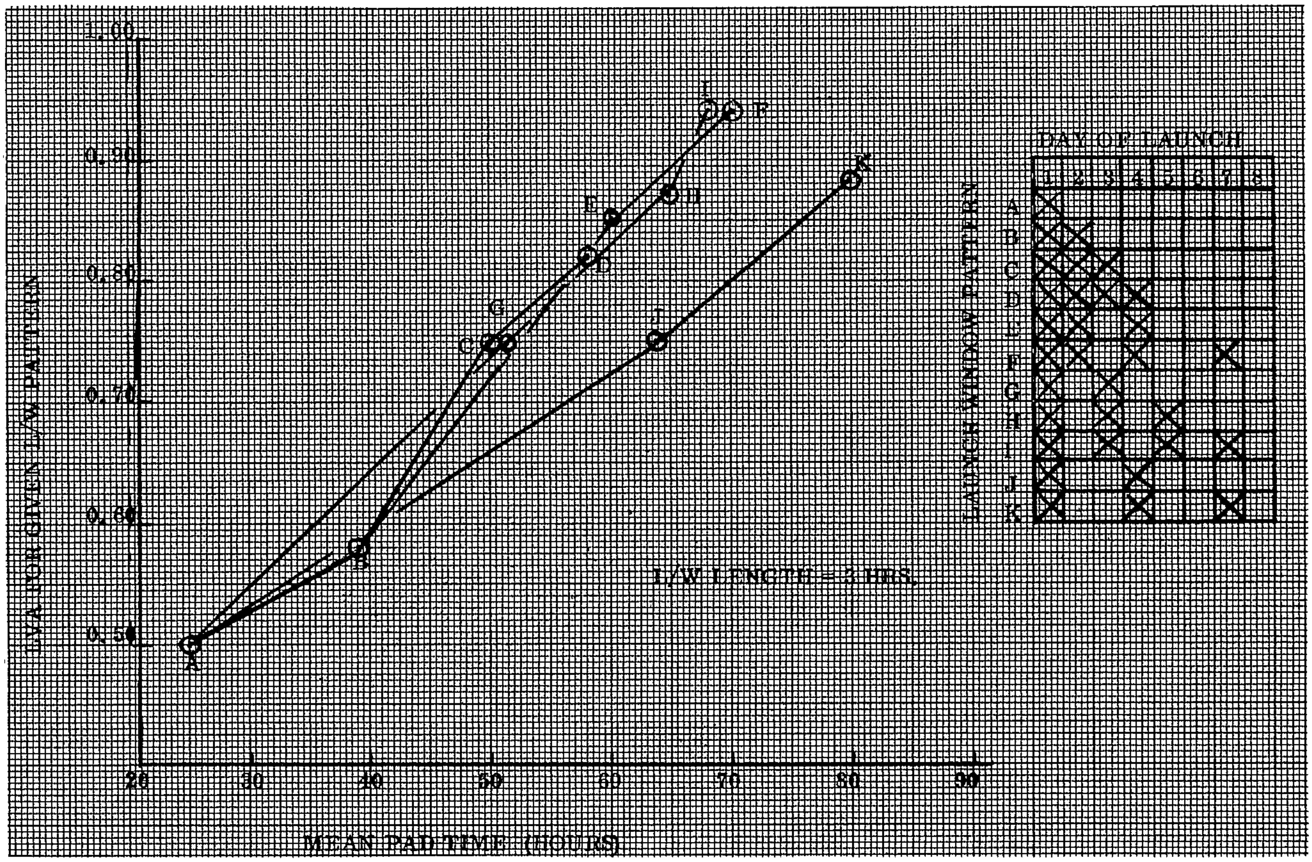
Figure 6 presents the difference between the actual and the minimum times in the backout sequences. In 76 percent of the scrubs, the difference was zero, and in 90 percent of the scrubs, the difference was less than 2 hours. This indicated a high probability of completing the backout sequence with little or no delay.

Figure 6 also presents the probability of being behind schedule when the countdown is resumed at T-25. This delay can occur because of the ground rule that the next launch window will be attempted whenever the delay time at the completion of the recycle hold is less than the window length. The relative high probability that the time behind schedule will be zero when the count is resumed at T-25 can be attributed to the high probability of being up at the completion of the recycle hold. (Reference Figure 7).



L/W PATTERN	DESCRIPTION
A	24 hours between L/W's
B	48 hours between L/W's
C	72 hours between L/W's
D	24 hours between L/W 1 and L/W 2, 48 hours between L/W 2 and L/W 3, 72 hours between L/W 3 and L/W 4.

FIGURE 4. LVA FOR VARIOUS WINDOW PATTERNS



DAY OF LAUNCH

	1	2	3	4	5	6	7	8
A	X							
B	X	X						
C	X	X	X					
D	X	X	X	X				
E	X	X		X				
F	X	X		X			X	
G	X		X					
H	X		X		X			
I	X		X		X		X	
J	X			X				
K	X			X			X	

FIGURE 5. LVA vs MEAN TIME VEHICLE ON PAD

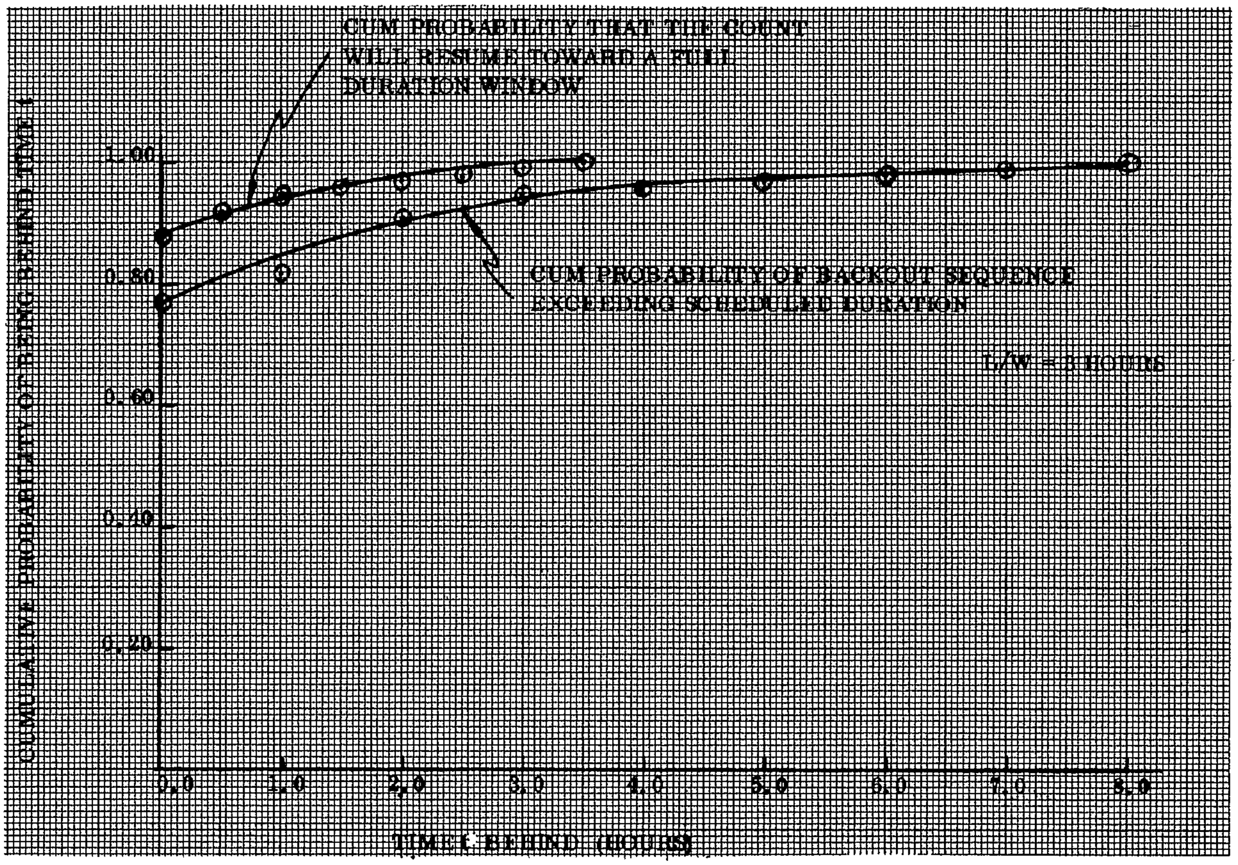


FIGURE 6. PROBABILITY OF BEING BEHIND TIME t

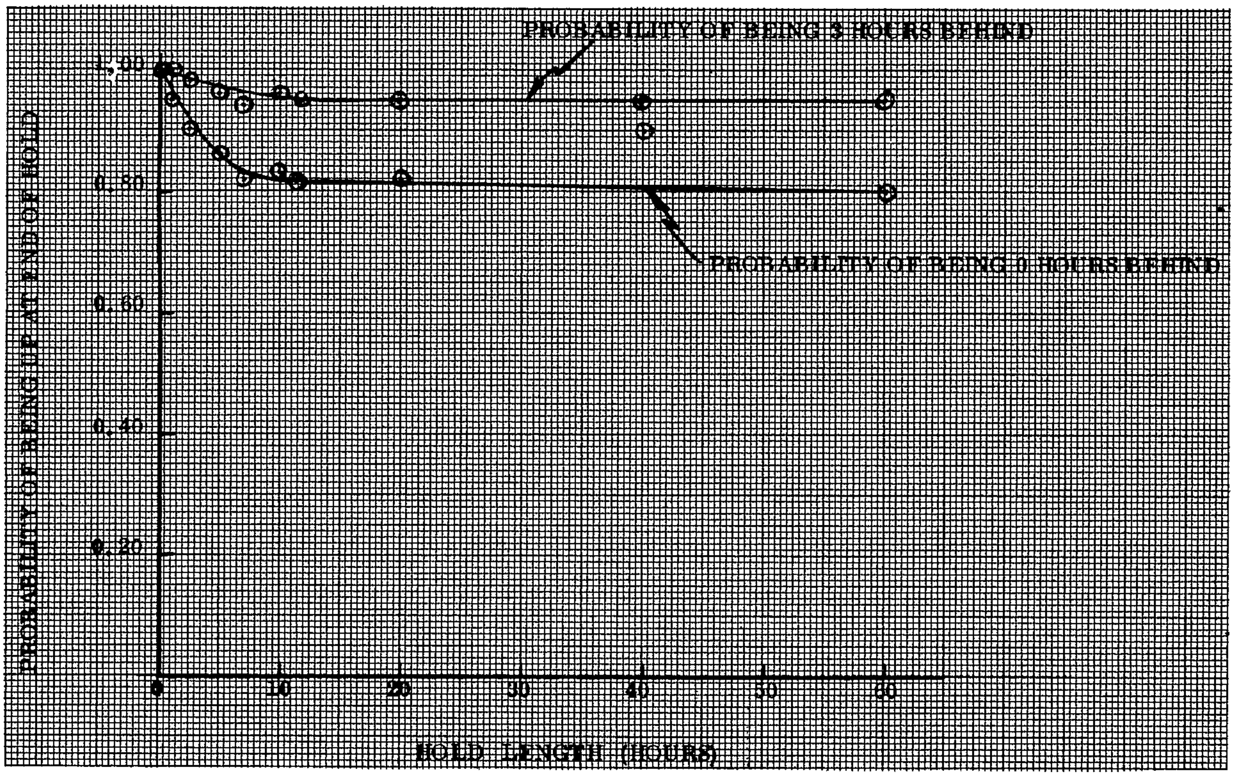


FIGURE 7. PROBABILITY OF BEING UP AT SCHEDULED END OF RECYCLE HOLD

Figure 7 shows the probability of being up (resuming the countdown without delay) at the scheduled end of the hold. This probability rapidly decreases as the hold length increases to 10 hours. However, after 10 hours this probability remains almost constant. The probability of the hold being delayed less than 3 hours for hold lengths greater than 10 hours is high, which justifies the relatively small difference in LVA between launch windows spaced 48 hours apart and windows spaced 72 hours apart (Reference Figure 4).

CONCLUSIONS

In summary, some of the significant features of the Saturn V Prelaunch Systems Simulation Model were:

1. The large data base from which an event data tape was created for the model.
2. The use of HELP routines such as for inputting event data into the model.
3. The use of an IBM 2250-3 graphics terminal for model checkout and debugging.
4. The addition of special blocks to the GPSS-II language, especially the OUTPUT and FORMAT blocks.
5. The analytical techniques of reducing the mean time to failure and the mean repair time distributions.
6. The extensive use of MACRO's to reduce coding.

Some of the conclusions which were drawn after completion of the model are:

1. GPSS was an excellent simulation language for sequence of event type simulations; however, the user is handicapped in inputting large amounts of data into GPSS blocks.
2. The separation of the data bank from the simulation model allowed for greater ease in data editing and updating, especially for models having large data inputs.
3. Reliability and maintenance engineering and systems operations were responsible for the validity of the input data; therefore, eliminating many of the problems associated with bad data inputs.
4. The use of engineers to construct the simulation model logic was very helpful in obtaining a model more nearly representing the real world environment. On the other hand, the use of COBOL and assembler language programmers to maintain the data bank and write special programs was very helpful in obtaining efficient computer programs.

REFERENCES

1. Saturn V Systems Prelaunch Simulation Model, Internal Document Number D5-6172-H-070-A, The Boeing Company, Huntsville, Alabama, 1968.
2. J. E. Snyder, E. R. Bennich and Y. H. Lindsey, "Implementation of Advanced Simulation Techniques for Predicting The Saturn V Launch Vehicle System Behavior", Proceedings of AIAA, 1966.
3. M. E. Stephenson's, "Graphics Simulation with GPSS", Digest of the Second Conference on Applications of Simulation, December 2-4, 1968.
4. B. J. Schroer, "Saturn V Prelaunch System Simulation Model for Multiple Launch Windows", Technical Memo 5-6172-H-145, The Boeing Company, Huntsville, Alabama, 1968.
5. Boeing System Simulator, Internal Document Number D5-15771, The Boeing Company, Huntsville, Alabama, 1969.

BIOGRAPHY

Bernard Schroer is a Research Engineer assigned to the Southeast Teleservices Operations of The Boeing Company. His present responsibility is to investigate new applications of the computer with emphasis on interactive computer graphics. He received an associate degree from Purdue University, a B.S. in Engineering from Western Michigan University, and a M.S. in Engineering from the University of Alabama. He is a registered professional engineer and a member of AIIE, ORSA, and SCi.