IBM Report No. 66-894-0008

ALL DIGITAL SIMULATION OF SATURN I, IB AND V BOOST VEHICLE AND GUIDANCE AND CONTROL SYSTEMS

by

W. D. Carson
R. E. Poupard
T. D. Steele
F. W. Eubank

INTERNATIONAL BUSINESS MACHINES CORPORATION

Federal Systems Division
Space Systems Center
Huntsville, Alabama
June 1966

ALL DIGITAL SIMULATION OF SATURN I, IB AND V BOOST VEHICLE AND GUIDANCE AND CONTROL SYSTEMS

by W. D. Carson, F. W. Eubank R. E. Poupard T. D. Steele

International Business Machines Corporation

Federal Systems Division Space Systems Center Huntsville, Alabama

I. INTRODUCTION

The Saturn V launch vehicle is being developed by The National Aeronautics and Space Administration's George C. Marshall Space Flight Center for Project Apollo; Saturn I and Saturn IB vehicles are providing the early testing and support for Project Apollo. The nerve center of the Saturn is its guidance and control system. An airborne digital computer provides the link which closes both the guidance and control loops, making verification of the flight computer program of vital importance. During a powered flight this onboard digital computer program can be divided into four major parts:

a) guidance, including navigation,
b) control,
c) vehicle sequencing, and
d) computer telemetry.

Each of these major computer functions must be verified and tested prior to launch, and many procedures are currently used. They include open loop tests of the flight hardware, closed loop studies (using a laboratory model of the flight computer with both analog and digital models of the Saturn vehicle), and an all-digital simulation of both the flight computer and the Saturn vehicle. Each has its own advantages, and those of the all-digital simulation are summarized briefly in the following discussion.

Simulation is defined as the analog or digital computer implementation of a set of equations which represent some usually complex portion of the physical world (system). Simulation has followed the development of computers, as it would be impossible to simulate most systems without a computer. In aerospace work the need for simulation is particularly acute since enormous expenditures are required to produce prototype or engineering models. In many cases these models are unavailable, and the first flight is the first test. Simulation provides answers similar to those obtained from exhaustive laboratory tests of an engineering model.

The all-digital simulation described here consists of a marriage between two separate simulators. The first simulator is a digital flight computer model called Simulational Interpretive Routine by Tedley,[1] since it makes the IBM 7094 Data Processing System appear to be the flight computer. It copies the flight computer in word length, instruction execution, and timing. In this case the flight computer is either the ASC-15 (Saturn I) or the Launch Vehicle Digital Computer (Saturn IB and V). The second simulator is a mathematical model of the Saturn vehicle and the remaining guidance and control hardware. It contains the six-degree-of-freedom equations of motion representing the Saturn rigid body dynamics. Hence, the name 6D is often applied to the simulator which also contains a model of the Saturn control system and a set of calculations designed to represent the inertial platform. The essential guidance and control interfaces are simulated in enough detail to permit analysis of the Saturn vehicle closed-loop guidance and control performance. The simulation requires the flight computer model to perform the flight sequencing as in actual flight, and provides the flight computer model with the appropriate sequencing command responses. Flight computer telemetry is recorded as the simulated flight progresses, permitting postflight analysis of the flight program as in actual flight. In some applications, discussed in Sections II. A and III. B, the detailed flight-computer model is not required and is replaced by a simpler model called the FORTRAN guidance model.

This all-digital simulator has advantages over other flight program tests. It is closed loop but has no hardware interface problems as it is entirely contained in one computer and in one program. Tedious programming requirements are eliminated as simulation requires no real-time operation. Studies performed on this simulator are repeatable, and can include numerous flight perturbations with minor programming effort. The simulator is readily accessible to more than one analyst at a time; user maintenance is at a minimum. While it is recognized that all-digital simulation may not be the best solution for every simulation problem, its usefulness has been established for the Saturn guidance system studies and analyses. The basic construction and use of this simulator is the subject of this note. The treatment will be general but specific enough to provide a useful insight to a complicated simulation problem.

---

[1] Tedley, R. S., Digital Computer and Control Engineering, McGraw-Hill, 1960, p. 143.

II. SIMULATOR DESCRIPTION

As mentioned in the Introduction, there are two simulators involved — the Launch Vehicle Digital Computer (LVDC) simulator, and the Saturn vehicle simulator (6D). Figure II. 1 shows a general block diagram of the combined simulation.

A. LVDC Simulator

A brief discussion of the onboard digital computer functions is required for a better understanding of the LVDC simulator. Four major functions performed by the LVDC are:

a) quidance and navigation,
b) control
c) sequencing, and
d) telemetry.

The quidance loop is closed through the LVDC. An inertial platform provides the LVDC with measured velocity changes. The computer must then add gravitational velocity changes to the measured quantities and perform the required integrations to obtain the current vector position and velocity. The guidance equations use positon, velocity, magnitude of acceleration, and time to generate steering commands, which are the desired platform gimbal angles. These desired angles are the output of the guidance equations and serve as the input to the computer control calculations.

The control loop is also closed through the LVDC. The desired gimbal angles from the guidance routine are subtracted from the measured gimbal angles obtained from the platform. These differences are transformed to body-fixed coordinates and issued at a high rate (25/sec) as attitude errors to the analog control computer, closing the control loop.

Vehicle sequencing consists of discrete signals issued by the LVDC through a stage switch selector to provide necessary switching functions to the various Saturn stages. All sequencing is performed by the LVDC. The telemetry functions require that certain words be telemetered periodically from the computer to aid in real-time evaluation of the vehicle performance and provide data for mission control decisions, and postflight evaluation.

In order to test the digital program designed for use in the LVDC, it is necessary to have a model of the flight computer which will:

a) execute the flight program instructions exactly as the hardware,
b) carry out all arithmetic operations with precisely the same accuracy as the flight computer, and
c) preserve communication and timing.

The Simulational Interpretive Routine designed to do this is called a "bit-by-bit" (BBB) simulator since its computations compare exactly, digital-bit-by-digital-bit, with the LVDC computations. The BBB model simulates the LVDC memory, initializes all locations (just as they would be initialized in flight), decodes instructions, and executes them sequentially as dictated by the flight program. Any instruction errors in the flight program (Section III. A describes some typical errors) will show the same symptoms in the simulator as in flight. Any detected instruction of data errors can be corrected in the BBB model by appropriate memory changes at the beginning of a run, providing a test of proposed changes. The input/output data paths connecting the flight computer are simulated, permitting a study of timing or data-handling problems in the communication interfaces. The four primary computer tasks outlined above (guidance, control, sequencing, and telemetry) must be verified before each flight using the BBB model. Even with this detailed simulation, the cause, or even the presence of an error, is not always obvious. The simulation remains a tool of the analyst — not a replacement for him.

Two modes of operation are possible for the BBB simulator. In preflight studies, when the flight program must be exercised with guidance and control loops closed, the LVDC output quantities are fed to the 6D, and appropriate flight inputs are determined. Figure II. 1 shows the principal communications interfaces. The BBB model requires discrete signals, gimbal angles, and velocity data as inputs. Its outputs consist of attitude-error signals, flight sequencing discretes to the 6D, and telemetry data. In postflight evaluation (the second mode of operation), the inputs are already available from flight data, so all outputs are recorded simply for comparison with flight results. The use of the BBB simulator is open loop in this mode.

A FORTRAN model of the LVDC is used for guidance and navigation studies, to determine range of variables for scaling the LVDC flight program, to evaluate failure effects studies, and for all other studies which do not require the BBB simulation of the LVDC flight program. To ensure an adequate model for preliminary flight program design and checkout, however, all essential LVDC flight program algorithms are included.

Two important advantages of the FORTRAN model over the BBB model are: a large reduction in computer time necessary to complete a simulation run, which implies a larger number of runs for a given time, and the computer language used in writing the simulator. As the name implies, the FORTRAN model is written in FORTRAN which allows the model to be changed easily and is understood by more analysts than the LVDC flight-program language.

B. 6D Simulator

The 6D simulator must take the outputs from the LVDC model (either BBB or FORTRAN) and process them to compute the inputs to the LVDC. These communications were discussed in a previous paragraph (page 2) and are shown in Figure II. 1. Proceeding around the loop in Figure II. 1, the LVDC model issues attitude error commands to the control computer. They are filtered and combined with attitude-rate commands and load-relief signals from body-mounted accelerometers to produce engine gimbal commands. These commands are transmitted by the actuator model to the vehicle simulator where rotational and translational accelerations are computed. The characteristics of the vehicle's physical environment (aerodynamics and gravitation) are calculated and their effects included in the equations of motion. The rotational and translational accelerations are integrated for use in models of the inertial platform, the vehicle-mounted accelerometers, and the rate gyros. The 6D computations of position and velocity serve as standards with which the LVDC navigation quantities may be compared.

The 6D discussion is divided into three parts: the launch vehicle and its environment, the inertial platform, and the control and actuator systems.

1. Launch Vehicle. The Saturn IB boost vehicle is shown in Figure II. 2 and consists of two stages. The first (S-IB) stage is powered by eight Rocketdyne H-1 engines which generate a total thrust of 1. 6-million pounds. The four inboard engines are clustered around the vehicle's centerline and are canted such that the thrust vector of each engine points through the approximate vehicle center of gravity at liftoff. The outer four engines are gimballed for control purposes and are also canted. The second (S-IVB) stage is powered by a single Rocketdyne J2 engine which is mounted on the vehicle's centerline and gimballed for pitch and yaw control. Roll control is achieved by reaction jets mounted on the S-IVB stage.

The launch vehicle simulation is conveniently divided into five parts:

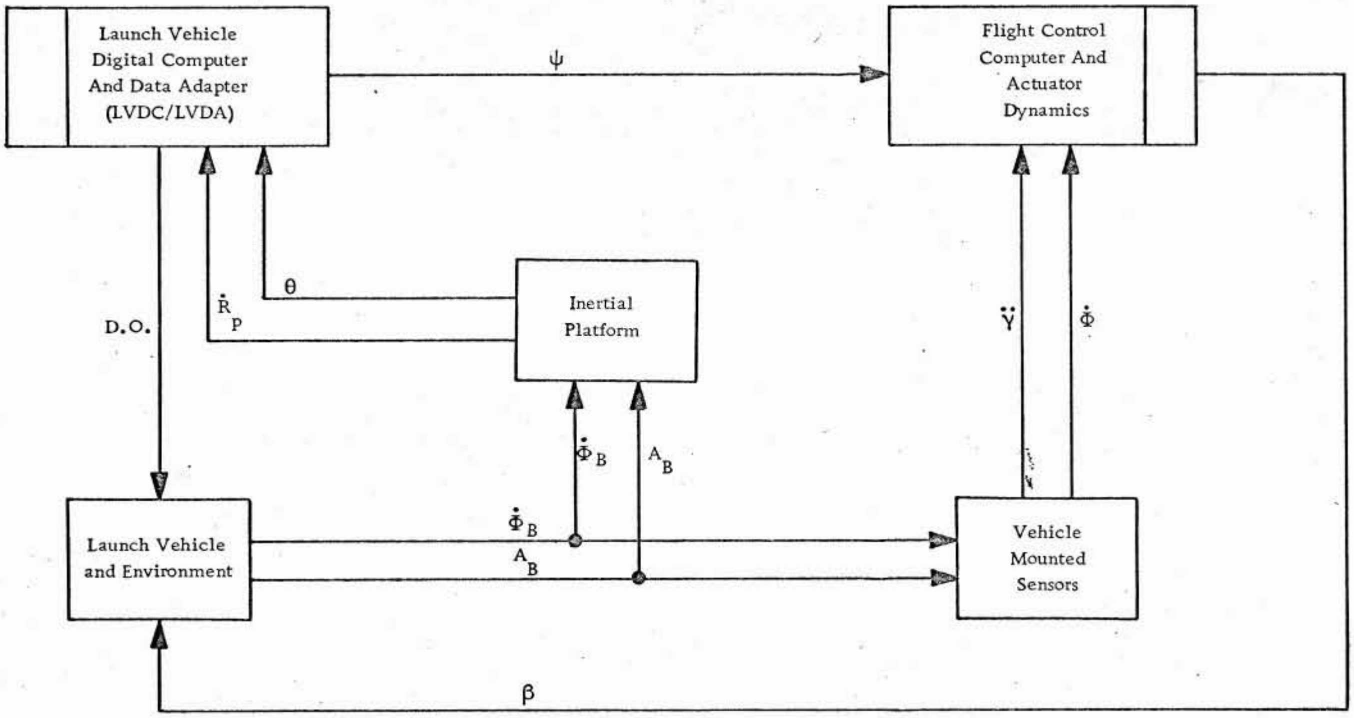
a) the rigid body equations of motion,
b) aerodynamics,
c) gravitation,
d) propulsion and mass characteristics, and
e) the vehicle-mounted sensors.

a. Equations of Motion. The vehicle is assumed to be a rigid body and, consequently, has six-degrees-of-freedom—three rotational and three translational. The equations of motion were derived from the principles of Newtonian Mechanics. The velocity of the center of gravity relative to the body is small compared to the vehicle inertial velocity and is neglected. The external forces (excepting gravitational forces) are summed with respect to a set of coordinates originating at the center of gravity, and extending along the vehicle's pitch, yaw, and roll axes. The resultant force is then divided by the total vehicle mass to obtain acceleration. This acceleration is transformed to an inertial frame where it is summed with gravitational acceleration and integrated to obtain true inertial velocity. The integration scheme is a modified form of trapezoidal integration, and double precision is used for computation of most integrals.

The rotational equations of motion are simplified by making use of the vehicle's geometric and mass symmetry about the longitudinal (roll) axis. It is assumed that the vehicle's pitch and yaw axes are aligned with the principal axes of inertia. These equations are solved in the body frame by summing the external moments, dividing by the appropriate moment of inertia, and adding coupling between axes.

b. Aerodynamics. There are aerodynamic forces acting on the vehicle as a result of its passage through the atmosphere. The vehicle is launched from a specified site located on the rotating earth, and the atmosphere is assumed to rotate with the earth. The characteristics of the atmosphere are obtained from Patrick Air Force Base Standard Atmosphere (1963) as a function of altitude. The logitudinal aerodynamic force equation can be developed from knowledge of these characteristics and the priciple of Bernoulli. An additional term is added to account for the base drag due to the vacuum created at the base of the vehicle. The linearized normal force equation is an empirical equation proportional to the aerodynamic normal force coefficient which also depends upon atmosphere characteristics, principally the Mach number. Both the longitudinal and the normal force equations use the relative velocity of the vehicle, which is the vector difference in the vehicle's inertial velocity and the atmosphere's inertial velocity (earth's rotation and wind velocity). Wind velocity may be excluded or modified by programmer option.

The vehicle's center of gravity (cg) and center of pressure (cp) are not at the same point; and, since the aerodynamic forces may be assumed to act at the center of pressure, a turning moment is created about the center of gravity. Wind tunnel measurements yield



D.O. - Discrete Outputs
ψ - Steering Commands (Attitude Errors)
A_B - Body Fixed Translational Acceleration
$\dot{\Phi}_B$ - Body Fixed Rotational Velocities
$\ddot{Y}$ - Body Fixed Accelerometer Outputs
$\dot{\Phi}$ - Body Fixed Rate Gyro Outputs
β - Engine Gimbal Angles
$\dot{R}_P$ - Platform Accelerometer Outputs
θ - Platform Gimbal (Attitude) Angles

Figure II. 1. General Block Diagram of the 6D Simulation

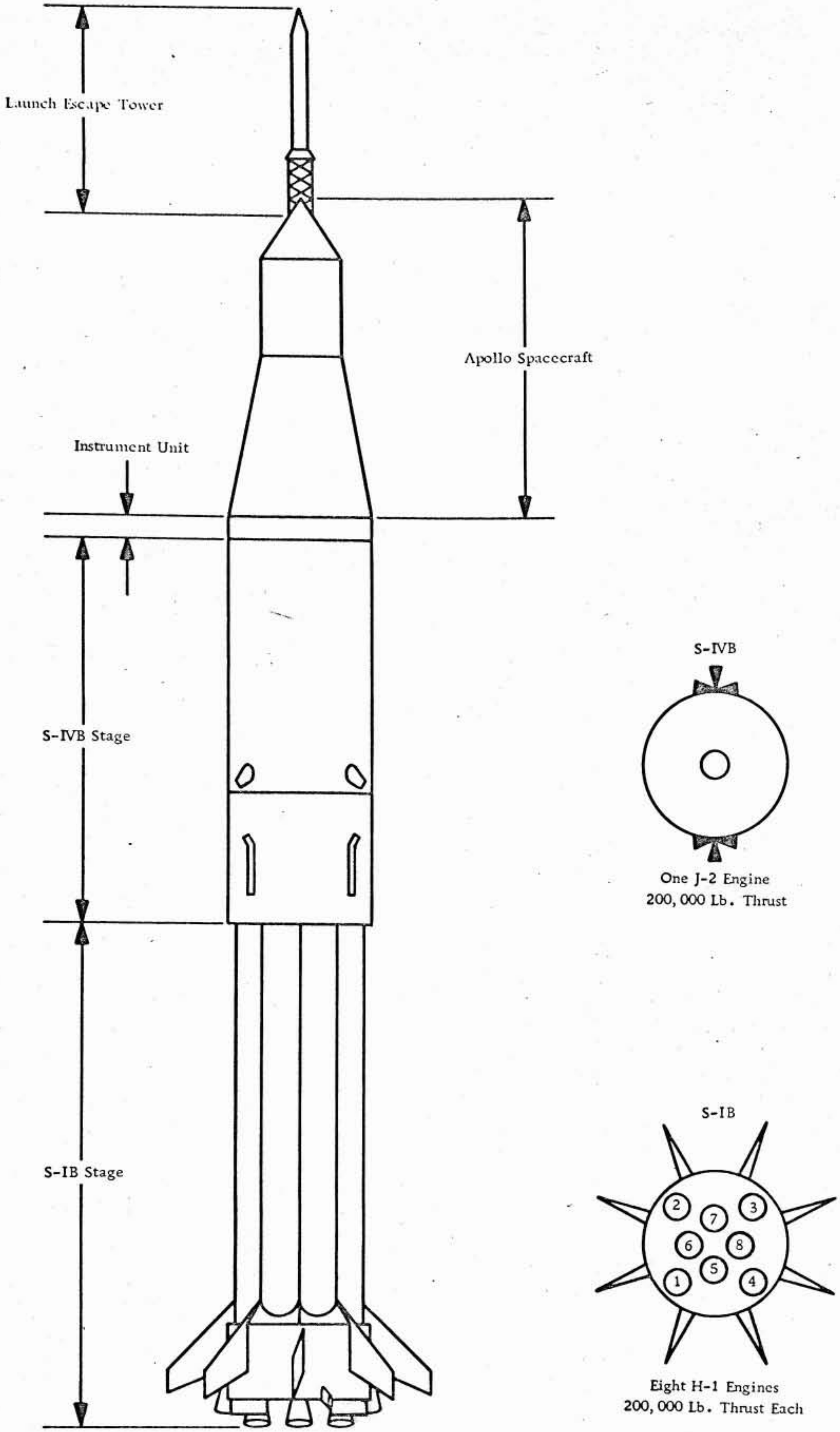


Figure II. 2. Saturn I-B Vehicle

empirical data for the center of pressure location relative to the engine gimbal plane. The moment arm is calculated from the knowledge of the center of gravity location which is fixed by the knowledge of the vehicle's mass distribution. For the calculation of the aerodynamic moments the center of pressure is assumed to be located on the vehicle's roll axis.

c. Gravitation. The gravitational acceleration is calculated in the 6D, combined with other accelerations; and the resulting acceleration is integrated to provide essentially error-free velocities and positions for comparison with LVDC navigation quantities. The method employed for gravitation calculations is similar to the method used by the Saturn flight programs, except that the equations include four terms in a Fischer ellipsoid model[2] of the earth's gravitational field instead of two.

d. Thrust-Mass Characteristics. Engine thrust and vehicle mass characteristics (i.e., cg location, moments of inertia, etc.) are obtained from the Propulsion and Vehicle Engineering Laboratory, Marshall Space Flight Center. The data are generated in a detailed simulation of the Saturn propulsion system. This simulation employs empirical equations and uses measurements taken from static-test firings. The thrust and mass data are updated for each vehicle to provide accurate results from the 6D without actually including an extensive propulsion system simulation.

The individual engine thrust vectors are resolved through the engine gimbal and cant angles and are summed to obtain a resultant force acting at the vehicle's center of gravity. Based on engine and vehicle geometry the turning moments are calculated for use in the rotational equations of motion.

e. Vehicle Sensors. Saturn's control system employs two types of vehicle-mounted sensors: rate gyros for stability (both stages) and accelerometers for wind-load relief (first stage of Saturn IB only). The outputs from the vehicle equations of motion are utilized directly to simulate the outputs of these sensors. Corrections for the signals measured by accelerometers not mounted at the vehicle's center of gravity are also added when necessary.

2. Inertial Platform. The platform simulator is used to simulate the outputs obtained from the Bendix ST-124M stabilized inertial platform during flight. The ST-124M is a three-gimbal platform having an inner-to-outer gimbal order of pitch, yaw, and roll. These gimbals provide angular measurements for attitude control of the vehicle through the LVDC and the flight control computer. These measurements are used, also, to provide a body coordinate to inertial coordinate system transformation matrix and to simulate the outputs of the integrating accelerometers. The model has provisions to include platform errors as well as gimbal angle misalignments and accelerometer failures.

3. Flight Control Computer and Actuator Dynamics. The Saturn's flight control computer and related subsystems are analog and must be represented digitally in the 6D simulation. The flight control computer combines inputs from the rate gyros, control accelerometers, and the LVDC to generate a gimbal command to the control-engine actuators. The control computer filters, amplifies, and sums these inputs. The gains and filters are changed periodically during flight by switches activated by discrete outputs from the LVDC. The pitch, yaw, and roll signals are then appropriately combined to provide inputs to the hydraulic actuator system which positions the control engines. The engine gimbal angles are limited to simulate the physical stops mounted on each control engine.

The data describing the filters and the hydraulic actuator system are usually given as linear transfer functions in terms of the Laplace variable. The use of such data assumes that linear differential equations will adequately describe the behavior of a system represented in this manner. Studies were made to show that a Z-form approximation to the inverse Laplace transform will provide adequate filter representation and actuator outputs for the range of input frequencies that are of interest (2 - 3 cps)[3,4,5]. Since the vehicle was assumed to be rigid, no bending and sloshing models are included. The Z-form theory is utilized in a separate program to obtain coefficients of difference (recursion) equations to represent the filters and the engine actuator system. Gain changes, as well as filter changes, are made whenever commanded by the LVDC.

---

[2] Fischer, I., "An Astrogeodetic World Datum from Geoidal Heights Based on the Flattening f = 1/298.3," JOURNAL OF GEOPHYSICAL RESEARCH, Vol. 65, No. 7, July 1960, pp. 2067-2076.

[3] Carson, W. D., "Digital Simulation of Analog Subsystems - A Numerical Example, "Astrionics Internal Note M-ASTR-IN-63-26, Astrionics Division, George C. Marshall Space Flight Center, Huntsville, Alabama, September 16, 1963.

[4] Tou, Julius T., Digital and Sampled-Data Control Systems, New York: McGraw-Hill Book Company, Inc., 1959.

[5] Ragazzini, J. R., and Franklin, G. F., Sampled-Data Control Systems, McGraw-Hill Book Company, Inc., New York, 1958.

III. SIMULATOR DEVELOPMENT AND USES

A. Early Application and Development

The use of an all-digital simulator for Saturn preflight evaluation started with the SA-5 flight, the first Saturn I, Block II vehicle. This vehicle was the first of the Saturns to attempt two-stage flight (S-IV second stage) and first to carry along a digital flight computer, although the digital computer operated open loop on this flight. Early in 1963 work began to combine the already existing digital simulations of the flight computer (ASC-15) and the vehicle (6D). These simulators had been developed, independently, for altogether different applications than combined simulation, although both operated on the IBM 7094. Several problems were encountered immediately. A common clock and a communication interface (internal to the IBM 7094) had to be established. Certain parts of the 6D required a fixed-time operation interval, notably the digital representation of the control system, whereas the ASC-1 gave and received outputs and inputs at varying times. Additionally, it was necessary to decide which model would lead the other; i.e., should the 6D integrate from t to $t + \Delta t$, and then the ASC-15 catch up, or vice versa.

These problems were resolved by decisions made early in the program. A convenient choice for the common clock was the ASC-15 computer time. This choice (made specifically for the ASC-15 drum-storage machine) has proven satisfactory, even with the newer core computer(LVDC), and is still in use. The ASC-15 required two types of inputs and generated two types of outputs. Discrete inputs and outputs were used for vehicle sequencing, and their occurrence times during a computation cycle were flight dependent, whereas measurements and computed commands always occurred at the same time in each computation cycle independent of the flight. The communication interface controlled the flow of both kinds of information between simulators. This was accomplished, in the case of discrete inputs and outputs, by testing appropriate registers for changes each time the interface routine was entered. In the second case, measured and computed data were transferred by clocks into the appropriate location in each simulator. Each data block was transferred only once per computation cycle. The communications interface was used to fix the integration step size for the 6D and to control the relative timing between the simulators. At the end of each ASC-15 drum revolution, the ASC-15 simulator transferred control to the communications block, where the decision was made whether the 6D should be called to catch up with the ASC-15. Thus, the 6D integration step size was fixed to be an integer multiple of the drum revolution time, and the ASC-15 was selected to lead the 6D in real time.

Implicit in the decision to communicate discrete inputs and outputs at one drum revolution intervals is the contention that no closer determination of event times than one drum revolution is required. This contention is true for all vehicle sequencing except the S-IV engine cutoff signal. In this instance, the cutoff signal is issued in a loop much shorter than one drum revolution, and this special discrete required communication between simulators on a word-time basis (1/64 of a drum revolution) near cutoff. Once cutoff was detected, the 6D simulator adjusted its step size to permit computation of the vehicle state at the cutoff time. In short, the communication block served as data manager for control of information transfer and for time keeping between the 6D and ASC-15 simulations.

While these decisions regarding timing and communications were being made and modified by experience, the combined simulation was proving its usefulness in flight program checkout for the SA-6 flight. Four types of errors were found in early checkout runs; two were flight program errors and two were simulator errors which appeared to be flight program errors. The first type of errors were coding errors made in the preparation of the ASC-15 flight program and are inevitable on tasks of this magnitude with time limitation. Their detection was the primary reason for construction of the combined simulation. Typical errors of this type included improper coefficients for guidance and navigation computations, erroneous initial conditions, and incorrect sine and cosine subroutine computations.

The second type of errors could be called conceptual errors in the flight program. Discovered by the combined simulation, they included gaps in the velocity computation which caused the vehicle to miss the desired cutoff velocity and scaling of some quantities resulting in "overflow" of their fixed-point representation under certain circumstances. For instance, in the SA-6 flight program, flight time was scaled so that if it exceeded 656 seconds, it would start over; i.e., 657 seconds would appear in the computer as 1 second. This scaling was adequate for most flight conditions since nominal flight time was 610 seconds, but it could be exceeded under some extreme — but possible — flight perturbations. When exceeded, the guidance system failed to provide accurate steering commands, and the vehicle failed to achieve the desired orbital conditions. Conceptual errors involved errors at a level above simple coding errors. They may occur when last-minute mission changes impose unforeseen operating conditions upon the flight program. Thus, they are potentially present in every flight program.

The first two types of errors, if undiscovered, could have caused severe mission degradation or even failure. At the time these errors were discovered in the SA-6 flight program, the flight program had successfully passed most of its other checkout procedures; and it is unlikely that many of the errors would have been found by other means. Therefore, although the combined digital simulation was not originally considered vital in the program checkout procedure, it soon became the most reliable and thorough flight program checkout tool in use.

The third error source uncovered in the studies was in the communications block described above. Timing problems were particularly difficult to isolate and cure. One problem cast considerable suspicion on the ASC-15 implementation of the cross-range steering equations. Errors of types one or two above were suspected; but, in truth, a one computation cycle transport lag in execution of the command was being introduced by the communication block. This transport lag caused a decrease in the system's stability margin which was causing the undesirable behavior.

A fourth type of error was uncovered when attempts were made to determine the source of "large"(200 m) navigation errors observed by comparing the separate 6D and ASC-15 values of the vehicle's position near cutoff. Once again, programming or conceptual errors were suspected, but these were eliminated in succession until such an explanation was illogical. Since the source of the error was not the flight program, the two simulators were suspected. The error was finally traced to the 6D, heretofore accepted as "perfect." Correction of the 6D decreased the navigation differences to an explainable 30 meters at cutoff. Thus, errors in the simulators themselves were the fourth type found.

Complete acceptance of both the airborne digital computer and all-digital checkout occurred after the SA-6 flight. This particular flight had an unexpected, unplanned, early engine shutdown in the first stage. The guidance implementation in the ASC-15 corrected for the perturbation and succeeded in placing the vehicle in the proper orbit. Of the system tests, the all-digital simulation alone had:

a) discovered certain scaling problems which would have prevented proper program operation in the event of an early-engine shutdown, and
b) subsequently, verified that the corrected flight program would successfully handle any engine shutdown condition.

B. Simulator Uses

The basic 6D vehicle simulator is used with and without the BBB LVDC model. When used without the BBB simulator, a FORTRAN representation of the equations solved by the LVDC is substituted. In this configuration, the simulator is used for studies such as the determination of the best form for implementation of navigation, guidance, and control equations in the LVDC. The primary uses of the over-all simulation (6D/BBB) are for verification of the flight program and for postflight evaluation of the guidance system. The BBB simulator is also used alone without the 6D vehicle portion in the postflight evaluation effort.

1. Studies and Analysis. The 6D/FORTRAN model is normally used for all studies and analyses. This version executes in approximately one-half real time on the IBM 7094 II. Examples of studies performed with the regular 6D are:

a) verification of logic used to initiate vehicle sequences,
b) navigation and guidance accuracy,
c) consumption of roll attitude control system fuel,
d) verification of backup and error path logic in the flight program,
e) determination of acceptable methods for guidance during mixture ratio shift in the J2 engine, and
f) algorithm studies.

A variation of this 6D/FORTRAN configuration is used for simulation of free fall or orbital flight. This version is used for verification of the proposed orbital navigation scheme, determination of three axis attitude control system fuel consumption, and determination of times of passage over ground stations.

These studies are performed in several phases, requiring slightly different versions of the basic simulation. In the initial studies, a simplified FORTRAN model of the guidance computer is adequate to study stability problems and basic implementation methods. Later studies require that exact algorithms be used in the FORTRAN model to study the accuracy problem, algorithm convergence, and scaling. Accuracy estimates are obtained by comparison with an ideal guidance scheme, based on calculus of variations, and an ideal vehicle.

Vehicle attitude during the orbital mission phase is maintained by a reaction jet control system. The attitude control scheme (i.e., logical decisions, computation of error commands, etc.), is implemented in the LVDC. Considerations such as control scheme, implementation, and limit cycles have a significant effect on fuel tank size. These are important considerations from a weight and volume standpoint. The vehicle simulation is used for these studies to select the best compromise between control scheme and implementation and fuel consumption required to maintain the vehicle attitude within acceptable bounds. The FORTRAN model is adequate, although algorithms must be included in the simulation.

It can be seen that the several versions of the basic simulator are used quite extensively and all are necessary to adequately define and specify the guidance computer program necessary to perform a given mission for a particular vehicle.

2. Flight Program Verification. After the flight program specifications are completely defined and the program written, a systematic procedure is necessary to verify that the finished program meets the specifications and is adequate to handle expected perturbations. There is a general agreement that, once the program specifications are defined, the flight program must not limit mission success. That is, any vehicle failures or perturbations that are sufficient to fail the flight program will already have caused a mission failure. The flight program must be written to accommodate uncertainties in vehicle parameters and certain non-critical hardware failures that do not cause a mission failure. Examples of vehicle perturbations include uncertainties in fuel load, vehicle mass, center of gravity location, mass flow rates, engine specific impulse, and thrust misalignments. In addition to these vehicle uncertainties, specifications, such as accomplishment of mission objectives with a failure of one first-stage engine after a specified time from liftoff, may exist. Mission objectives must also be met if certain discrete inputs to the LVDC are either missed by the computer or not issued by the vehicle's stages. Therefore, backups must be provided in the flight program for these discretes. The capability of the flight program to compensate for these vehicle failures and uncertainties, and meet required cutoff conditions at the same time, must be verified.

A systematic procedure for this verification has been established using the vehicle simulation combined with the BBB guidance computer simulation. In order to verify that the flight program was correctly written, it must be used in this effort. Therefore, the actual flight program tape is loaded into the BBB simulator. This verification also provides an opportunity to detect and correct any programming, scaling, or constant errors in the actual flight program. Simulator runs are made with vehicle failures and uncertainties inserted singly and in combination to simulate the worst possible conditions under which the flight program can reasonably be expected to perform. Between twenty and thirty perturbation runs are necessary with the simulator to completely verify the program operation. The ability of the vehicle to achieve stated mission objectives, such as a pre-determined orbit or specified impact area, is evaluated. In addition to verifying proper operation of the flight program, this procedure yields an estimate of guidance and control system performance that is necessary in postflight evaluation of the guidance and control system. Both the nominal behavior of each guidance and control variable and the variations are available from which a predicted flight envelope can be drawn.

3. Postflight Evaluation. Although the simulator described here has not been used for postflight analysis of the guidance and control system, an equivalent simulator was used for this purpose on the Saturn I, Block II vehicles. The two configurations used in this analysis are the full 6D/BBB simulator and the BBB simulator alone. The primary postflight use of the 6D/BBB simulator is in malfunction analysis. The computations done inflight are reconstructed to determine if the guidance computer performed correctly under the circumstances. This application was required only once on the Saturn I, Block II series when an engine failed during first-stage burn. That particular flight (SA-6) was reconstructed by two methods:

a) a trial-and-error method of thrust and mass flow rate adjustment in the remaining engines, and

b) using actual reconstructed thrust and center of gravity data from the postflight propulsion analysis.

Results from both methods agreed closely, both with each other and with telemetry reconstruction of the flight, at first stage cutoff. Table III.B.1 compares each of four sets of positions and velocities with the computer telemetered positions at first stage cutoff. The four sets of points were obtained as follows: case 1 corresponds to method 1 above. That is, the telemetered position and velocity values are compared with a 6D/BBB simulation in which thrust and mass flow rates have been adjusted. Case 2 compares telemetered quantities with a 6D/BBB run as described in method 2. Case 3 compares telemetry data with range tracking information. Case 4, the worst of the lot, makes a comparison between the telemetered quantities and the output of a 6D/BBB simulation in which thrust and mass flow rate for the failed engine alone had been modified. All differences shown in the table, except Case 4, were less than 1% of the actual position or velocity. Differences for Case 4 ranged to slightly over 2%.

This example illustrates that the simulator can be used to reconstruct flight conditions quite closely and is useful in determining whether or not the guidance computer operated correctly after the malfunction. It also demonstrates the good agreement between 6D simulation of the vehicle and the actual vehicle performance.

The second application of the simulator in postflight analysis requires the use of the BBB simulator alone. In this application the correct operation of the guidance computer hardware is determined by using actual telemetered flight computer inputs as simulator inputs. The simulated computer is then allowed to compute for one computation cycle, and the data generated by the simulator is compared to corresponding data from flight computer telemetry. If these data do not compare bit for bit, the cause is determined. If the cause is in the computer hardware, a more detailed analysis of the computer operation during that computation cycle will follow in an attempt to pinpoint the hardware failure. Only the data telemetered from the guidance computer can be compared in this manner, which implies that every operation of the guidance computer cannot be monitored using this technique. Table III.B.2 shows the amount of data examined with this procedure for two typical Saturn I, Block II vehicles. Not all telemetered data can be compared since a portion of the data are input data, and other data yield information on hardware operation that is not simulated.

This analysis tool is not a guaranteed method of locating computer faults. However, it will permit the determination of the area of possible malfunctions in the computer. In addition, it increases confidence in the proper operation of the guidance computer during a flight.

TABLE III. B. 1

COMPARISON OF TRAJECTORY RECONSTRUCTION DATA

DATA SOURCE	IN PLANE VELOCITY ERROR DIFFERENCES		IN PLANE POSITION ERROR DIFFERENCES	
	X	Y	X	Y
Case 1	0.132%	0.537%	0.176%	0.315%
Case 2	0.135%	0.293%	0.035%	0.223%
Case 3	0.057%	0.187%	0.078%	0.095%
Case 4	0.782%	1.680%	0.576%	2.040%

Case 1: Trial and error adjustment of thrust and mass flow rate.
Case 2: Thrust and mass data from postflight propulsion analysis.
Case 3: Range tracking data.
Case 4: Reduction of total thrust and mass flow rate only to account for failed engine.

Comparison at first stage cutoff.

f. IIT Research Institute (NAS8-20129), Low Thrust

Preparation of the final report is also beginning on this contract with some small consideration being given to the effect of oblateness and the three-dimensional guidance problem. Brief exercise of a spiral descent guidance scheme, based on steady-state circular velocity conditions being maintained throughout the flight to provide a velocity reference or correlated velocity, has been reported.

g. Republic Aviation (NAS8-20130), Optimization Theory and Celestial Mechanics

The bimonthly progress report for March and April on this contract indicates that the major portion of work under this contract has been completed. Dr. Morrison indicates further that the remaining time in the contract will be used in clearing up details, checking results and preparation of the final report. No difficulties in completing the contract on schedule are anticipated.

h. Vanderbilt University (NAS8-20371), Applications of COV to Trajectory Problems

The progress report for April on this contract indicates that work was continued on the multi-stage trajectory optimization problem initiated under the preceding contract NAS8-2619. Work was also initiated, through studies of steepest descent and various other direct methods, to improve computational procedures in direct methods. Dr. Boyce indicates that work will continue in the same areas during the next reporting period.

D. Optimization Theory Branch

1. In-House

Application of Optimal Control Theory for Design of Load Relief Control Systems - Efforts during this period have centered on attempts to find fixed feedback gains producing performance comparable to the optimum time response of a system subjected to deterministic disturbances. A suitable iteration technique for solving the nonlinear differential equations and two-point boundary value problems has not been found; however, first attempts at applying a quasi-linearization method for solution of such problems have given promising results.

2. Contractors

a. Northrop Schedule Order #1

Objectives: (1) To investigate load relief systems for the Saturn V/Voyager, and (2) to determine the applicability of learning systems to booster control and off-line problem solving.