Proc. Indian Acad. Sci., Vol. C 1, No. 4, December 1978, pp. 403–422, © printed in India

Theoretical basis of a simulation package for the integrated optimisation of profile and trajectory of satellite launch vehicles

N SESHAGIRI
National Informatics Centre, C-4/41 Safdarjung Development Area, New Delhi 110 016

MS received 27 November 1977; revised 24 May 1978

Abstract. Introducing the concepts of set-theoretic sequencing logic, hierarchic optimisation and sensitivity-directed iterative simulation, this paper provides a theoretical basis for the development of a simulation package for the three-dimensional atmospheric controlled flight of satellite launch vehicles subject to specified control constraints. The approach described here enables the optimisation of the profile of the vehicle to be carried out together with its trajectory, given the orbital objectives The simulation package, SIMSPACE II, developed on this basis, is shown to have features superior to the MATS program and related packages.

Keywords. Satellite launch vehicle; trajectory simulation, design optimisation.

1. Introduction

The design of a satellite launch vehicle is best carried out as the integrated optimisation of the profile of the multi-stage rocket along with the trajectory, given the satellite characteristics along with the desired orbit. The first step towards such an integrated optimisation is the development of a simulation package which can be conveniently built into the optimisation procedure. The optimisation procedure, in turn, should be so developed as to accommodate certain specified control constraints.

A FORTRAN package for the simulation of the trajectory of the atmospheric flight, given the profile of the vehicle, was developed by Seshagiri *et al* (1969) for assisting the design efforts relating to the Indian satellite launch vehicle SLV-3. The development of this package, SIMSPACE I, paralleled the efforts of the TRW Systems Group, California, where the Mission Analysis and Trajectory Simulation (MATS) package was developed Both packages had similar capabilities though SIMSPACE I was computationally more efficient but somewhat inferior in flexibility to MATS. Both were phase-oriented with input parameter controlled simulation. However, MATS made use of the bucket concept for minimising the input storage requirements by identifying the data by phase and storing it *en masse;* as each phase is initiated the associated data are brought into active memory. On the other hand, SIMSPACE I relied on a set-theoretic sequencing logic for increasing computational efficiency When adequate details of MATS were released through the publications of Lanzano (1970a, b) some of the flexibility features of MATS were introduced into SIMSPACE I.

Both SIMSPACE I and MATS lacked the following desirable features.

(i) As the bucket concept of MATS and the sequencing logic of SIMPSAC have several complementary advantages, a framework is necessary for i grating the better features of both in a common package.

(ii) As the optimisation of the vehicle profile or that of the trajectory influe each other, best results are obtained through their integrated optimisati

(iii) The integrated optimisation should be carried out by more efficient te niques while preserving the main simulation framework. Both SIMSPA I and MATS, which are basically for trajectory simulation, can do this o through repeated trial-and-error simulations as was the case when SL was designed.

The above three desirable features missing in SIMSPACE I and MATS are int duced in SIMSPACE II, a new package developed by the author as an extensiv modified version of SIMSPACE I The theoretical basis of simulation and in grated optimisation in SIMSPACE II is described in the following sections.

2. Theoretical basis of simulation

The simulation of flight performance is realised in the proposed scheme predominan in the body co-ordinate system. However, certain input parameters are convenien expressed in other co-ordinate systems like the geocentric system and the top centric system. At the core of the simulation procedure is the calculation of inerti forces and moments. The natural system in which these are calculated is the bo co-ordinate system. The principal additive constituents of the force and the mome equations are thrust, gravity, jet damping, and inertial, control and aerodynam forces and moments. The schematic diagram of interrelations between the simul tion parameters and their co-ordinate transformation given in figure 1 is self-expl natory. The approach represented by figure 1 is a significant departure from tho advocated by Krause (1961), Harris (1963), Fogarty (1968) and the TRW Systen Group which developed MATS, in the manner in which the inter-coordinate tran formations are designed so as to minimise the time of simulation

2.1. *Effect of the earth's rotation*

For accurate determination of the co-ordinates of the vehicle and the radar range the earth's rotation has to be considered. The circumferential velocity of the eart at the launch point is

$$w_1 = R_e\, \omega \cos \phi_1,$$

where ω is the spin velocity of the earth, R_e is the radius of the earth and ϕ_1 is th latitude. The earth's rotational velocity is specified in the geocentric system an designated Ω. However, since the equations of motion are represented in the bod co-ordinates, the earth's rotational components in the body co-ordinates should be determined. The transformation matrix is

$$E = [e_{ij}] = [d_{ij}]\, [\tilde{d}_{ij}], \tag{1}$$

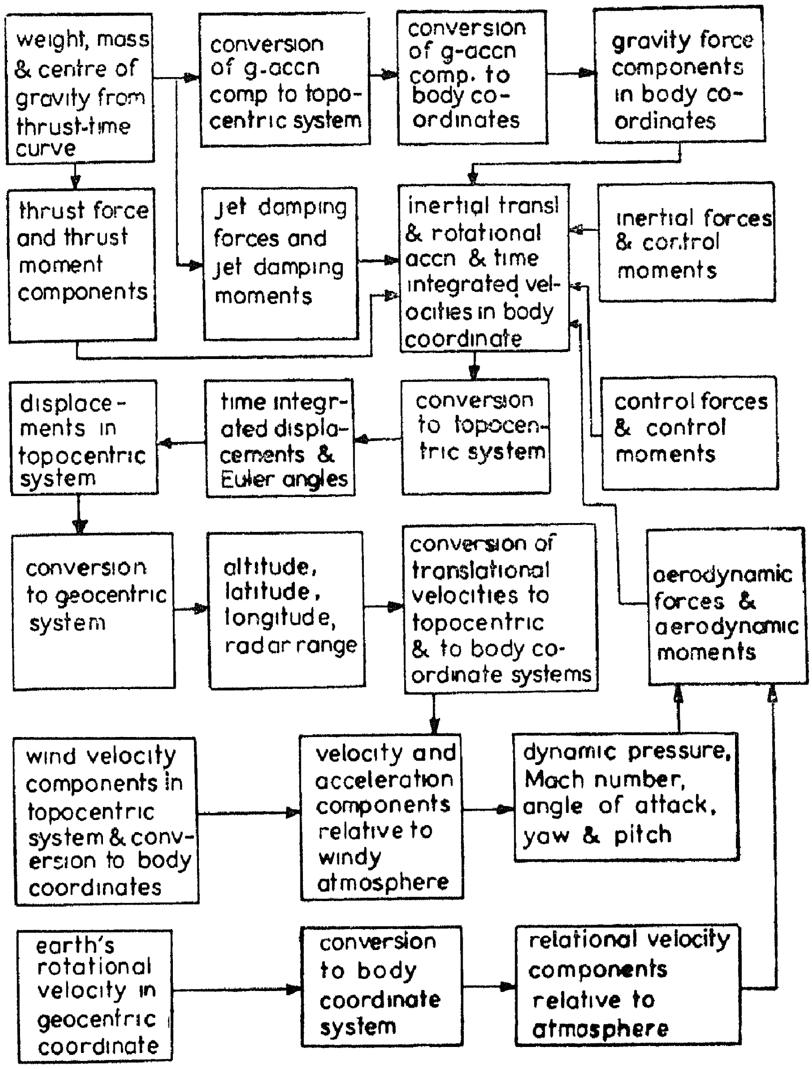

Figure 1. Schematic diagram of inter-relations between simulation parameters and their co-ordinate transformation.

where $\bar{d}_{11} = \cos\beta\cos\nu$, $\bar{d}_{22} = \sin\beta\cos\nu$, $\bar{d}_{33} = \sin\nu$ and all other $\bar{d}_{ij}$ are zero.

In (1) and throughout the rest of this section the subscripts i and j are taken to be indexes for space co-ordinates running through 1, 2 and 3

The body co-ordinate components of Ω are

$$b_i = \Omega \sum_j e_{ij}. \tag{2}$$

If the actual inertial rotational velocity components of the vehicle in body co-ordinates are q_i, then the rotational velocity components relative to the atmosphere are given by

$$\bar{q}_i = q_i - b_i. \tag{3}$$

The earth's rotation also has an influence over the translational velocity components. In topocentric co-ordinates these will be

$$v_{gi} = \Omega\ R\ \bar{d}_{i1};\ \nu = \nu_0 \text{ and } \bar{d}_{33} = 0, \tag{4}$$

where R is given in terms of the displacements X_{Gi} in geocentric co-ordinates

$$R = \left(\sum_i X_{Gi}^2\right)^{1/2}. \tag{5}$$

The contribution of the earth's rotation to the translational velocity components is given by

$$[v_i] = [d_{ij}]\ [v_{gj}]. \tag{6}$$

The calculation of translational velocity components relative to the atmosphere requires the inclusion of wind velocity components. Initially, wind velocity is specified in terms of linear velocity V_w and a rotational velocity θ_w. They are specified as functions of H, the altitude, given by $(R - R_e)$. These wind velocity components are expressed in the topocentric system by

$$v_{wi} = -V_w\Gamma_i;\ \Gamma_1 = \cos(\beta - \theta_w),\ \Gamma_2 = \sin(\beta - \theta_w),\ \Gamma_3 = 0. \tag{7}$$

Transforming (7) to body co-ordinates, we obtain

$$[v_{bi}] = [d_{ij}]\ [v_{wj}]. \tag{8}$$

Thus, if the inertial translational velocity components u_i are known, the velocity components u_{Ri} relative to the windy atmosphere can be expressed as

$$U_{Ri} = u_i - v_{bi} - v_i. \tag{9}$$

The composite relative velocity of the vehicle is

$$U_R = \left(\sum_i u_{Ri}^2\right)^{1/2}, \tag{10}$$

which corresponds to a Mach number, $M = U_R/a$, where a is the velocity of sound at an altitude H. Similarly, the dynamic pressure $\theta = \frac{1}{2}\rho U_R^2$ where ρ is the density of air at the altitude H.

2.2 *Atmospheric perturbations*

During an atmospheric flight if the angle of attack becomes very large, the aerodynamic forces produce pronounced bending moments which act on the vehicle.

To compute the angle of attack certain conventions will be followed. If the velocity vector coincides with the rocket axis, which is also the x_1 axis in body co-ordinates, the angle of attack α is assumed to be zero. The angle of attack is resolved into pitch angle of attack α_p and yaw angle of attack α_y corresponding to the resolution of U_R along the pitch and yaw axes. These are expressed as

$$\alpha_p = \tan^{-1}(u_{R3}/u_{R1}); \quad \alpha_y = \tan^{-1}(u_{22}/u_{R1}), \tag{11}$$

the angle of attack is

$$\alpha = \tan^{-1}[(u^2_{R2} + u^2_{R3})/u^2_{R1}]^{1/2}. \tag{12}$$

The aerodynamic force components are

$$F_{A1} = -C_D\, Q\, S; \; F_{A2} = -C_{N\alpha}\, \alpha_y\, Q\, S; \; F_{A3} = -C_{N\alpha}\, \alpha_p\, Q\, S], \tag{13}$$

where S is the reference area with respect to which the drag coefficient C_D and the normal force coefficient $C_{N\alpha}$ are calculated. The estimation of C_D and C_N as functions of Mach number and the angle of attack is made taking into account the shape of the vehicle.

In the computation of aerodynamic moments, the variation of the angle of attack is also included. If the time duration of an event during simulation is quantified to $\triangle t$, the variations in the relative translational velocity components can be expressed at any time t as

$$\dot{u}_{Ri} = \frac{1}{\triangle t}(u_{Ri}(t + \triangle t) - u_{Ri}(t)). \tag{14}$$

Utilising (11) and (14), the variations in α_p and α_y can be shown to be

$$\dot{\alpha}_p = A(u_{R3})\cos^2\alpha_p; \; \dot{\alpha}_y = A(u_{R2})\cos^2\alpha_y; \; A(a) = (\dot{a}\, u_{R1} - \dot{u}_{R1}a)/u^2_{R1}. \tag{15}$$

The aerodynamic moments about the three body axes are

$$M_{Ai} = (QS/2U_R)B_i, \tag{16}$$

where
$$B_1 = 2\,U_R\,C_1\,\alpha_y D + C_2\,\bar{q}_1 D^2 + C_3\,\bar{q}_3\,D^2,$$

$$B_2 = 2U_R C_{N\alpha}\,\alpha_p\,(C_g - C_p) + C_4\,\alpha_p D^2 + C_5\bar{q}_2\,D^2,$$

$$B_3 = 2U_R\,C_6\,\dot{\alpha}_y\,D^2 + C_7\,\bar{q}_3\,D^2 + C_{N\alpha}\,\alpha_y\,(C_g - C_p).$$

Here D is the reference diameter, C_g and C_p are the distance of the centre of gravity and the centre of pressure from the nozzle end; C_1, C_2 and C_3 denote the rolling moment coefficients, C_4 and C_5 the pitching moment coefficients and C_6 and C_7 the yawing moment coefficients.

2.3. *Control forces*

A change in the direction of a powered flight is determined by the forces acting normal to the instantaneous flight direction whereas the magnitude of acceleration is determined by the tangential forces. If all normal forces except the norm component of the vehicle weight are absent, the path is deflected by gravitation forces only. If one desires a different powered trajectory, one should introduce new force such that the resultant normal force controls the vehicle to correspond a pre-specified trajectory. If a thrust force normal to the instantaneous direction flight is required, the vehicle should be rotated till the X_1 axis forms an angle attack with the flight direction such that the required normal force is generated For effecting this rotation a steering force is applied. If forces due to the wind thrust misalignment or airframe deflection exist, they tend to deviate the vehicl from its prescribed trajectory. To balance the moment produced by these forces control forces are required.

Control forces and control moments can be specified as a set of conditional state ments. For example, we may have as inputs the rotational velocity components in the topocentric co-ordinates, viz., $(\dot{\phi}_c, \dot{\theta}_c, \dot{\psi}_c)$ as functions of time. The corresponding rotational velocity components in the body co-ordinate constituting the control components are

$$q_{c1} = \dot{\phi}_c - \dot{\theta}_c \sin\psi,$$

$$q_{c2} = \dot{\theta}_c \cos\psi \cos\phi + \dot{\psi}_c \sin\phi,$$

$$q_{c3} = \dot{\theta}_c \cos\psi \sin\phi + \dot{\psi}_c \cos\phi.$$

If the actual rotational velocities are q_i, the difference from the control components will be

$$\triangle q_i = q_{ci} - q_i.$$

Depending upon the type of control system used, certain conditional propositions governing the control forces F_{c2} and F_{c3} and the control moment M_{c1} may be described. The package is general enough to accommodate any practical control system through input conditions specified for F_{c2}, F_{c3} and M_{c1}

2.4. *Gravity, jet-damping and inertial components*

If no normal thrust components or aerodynamic lift forces act on the vehicle, the normal force deflecting the path of the vehicle will be that due to gravity. The advantage of gravity deflection during ascent through the denser atmosphere is that the angle of attack is kept small so that the bending moment is minimised. This will have the effect of reducing the structural weight of large vehicles. Apart from the deflection, gravitation accounts for a gravity loss which results, in general, in a loss of velocity and altitude. Thus, for considering gravity loss and gravity deflection during simulation, the three components of the gravity force should be calculated in the body coordinates.

If g_0 is the acceleration due to gravity at the surface of the earth, this acceleration of the vehicle at an altitude $H(= R - R_e)$ is given by

$$g = - g_0 (R_e/R)^2. \tag{17}$$

The gravity force due to this acceleration is governed by a variable mass due to the reduction in the weight of the propellant during the powered ascent. The change in the propellant weight can be computed from the given thrust-time curve.

At any time t let the weight of the vehicle be $W(t)$. Then, at a time $\triangle t$ later, the weight will be

$$W(t + \triangle t) = W(t) - \frac{T_{av} . \triangle t . W_p}{2 . T_I}, \tag{18}$$

where, T_{av} = average thrust of the current stage, W_p = propellant weight of the current stage and T_I = total impulse of the current stage. Correspondingly, there will be a shift in the centre of gravity so that,

$$C_g(t + \triangle t) = \frac{1}{W(t + \triangle t)} W(t + \triangle t)\, C_g(t) - \frac{T_{av} . \triangle t . W_p . C_{p0}}{2 . T_I}, \tag{19}$$

where C_{p0} is the initial value of C_p.

The mass and its time rate of change are given by

$$m(t) = W(t)/g(t) \text{ and} \tag{20}$$

$$\dot{m}(t + \triangle t) = -\, [m(t + \triangle t) - m(t)]/\triangle t. \tag{21}$$

In the topocentric co-ordinate, (X_{g1}, X_{g2}, X_{g3}), the components of the gravitational acceleration are

$$g_i = (g/R)(X_{gi} + \delta_i^3 R_e), \tag{22}$$

where $\delta_i^k = 1$ if $i = k$,

$= 0$ if $i \neq k$

Transforming these to the body co-ordinate system

$$[\bar{g}_i] = [d_{ij}]\ [g_j], \tag{23}$$

the corresponding forces are

$$F_{gi} = m\bar{g}_i. \tag{24}$$

The thrust force and moment components can be calculated from the thrust-time curve. However, we should take note of the differences between the atmospheric pressure P_a at the altitude H and the design pressure P_{des} which the thrust-time curve qualifies. The corrected thrust will be

$$T = T_{av} + (P_a - P_{des})\, A_e, \tag{25}$$

where A_e is the exit area of the nozzle.

If there are no thrust misalignment components, F_{T1}, F_{T2} and F_{T3} are all equal to T. However, in general, misalignment may be present and expressed by two angles, viz., the angle between the thrust vector and the X_1-axis (ξ) and the angle between the X_2-axis and the projection of the thrust vector on the X_2X_3-plane (η). In such cases, the corresponding forces and moments of thrust, jet-damping and inertia are along the X_2- and X_3-axis ($i = 2, 3$) respectively.

$$F_{Ti} = T \sin \xi\, (\delta_i^2 \cos \eta + \delta_i^3 \sin \eta),\ M_{Ti} = (-1)^{i-1} F_{Ti}\, C_g,$$

$$F_{Ji} = \dot{m}\, C_g\, (\delta_i^2\, q_3 + \delta_i^3\, q_2),\ M_{Ji} = -\dot{m}\, C_g^2\, q_i, \tag{26}$$

$$F_{Ii} = m(u_{i+1}\, q_{i-1} - u_{i-1}\, q_{i+1}),\ M_{Ii} = (-1)^i\, (I_2 - I_1)\, q_{i+1} q_{i-1},\ u_4 = u_1, q_4 = q_1.$$

Here I_1 and I_2 are the moments of inertia of the vehicle along the X_1- and X_2-axes respectively. These are functions of time because the mass of the vehicle and the mass distribution within the vehicle change with time during the powered ascent.

2.5. *Translation and rotation*

The forces and moments described in §§ 2.2 to 2.4 are the additive terms in the total force and moment components. Thus, the total force and moment components are respectively

$$F_i = F_{Ti} + F_{Ai} + F_{Ii} + F_{gi} + F_{ci} + F_{Ji},$$

$$M_i = M_{Ti} + M_{Ai} + M_{Ii} + M_{ci} + M_{Ji}, \tag{27}$$

$$F_{c1} = F_{J1} = M_{I1} = M_{J1} = M_{T1} = 0.$$

The inertial, translational and rotational acceleration components are respectively

$$\dot{u}_i = F_i/m,\ \dot{q}_i = M_i/I_i, \tag{28}$$

where I_i is the moment of inertia along the x_i-axis. Integrating these the corresponding inertial velocity components are realised.

$$u_i = u_{i0} + \int \dot{u}_i \cdot dt,\ q_i = q_{i0} + \int \dot{q}_i\, dt. \tag{29}$$

Since these are expressed in the body co-ordinate, a reverse transformation is required to express the displacements in the familiar topocentric and geocentric co-ordinates. In the topocentric system, they will be

$$[u_{gi}] = [d_{ij}]\, [u_j], \tag{30}$$

and $\dot{\phi} = q_1 - q_2 \cos \phi \tan \psi + q_3 \sin \phi \tan \psi,$

$$\dot{\theta} = -q_2 \frac{\cos\phi}{\tan\psi} + q_3 \frac{\sin\phi}{\cos\psi}, \tag{31}$$

$$\dot{\psi} = q_2 \sin\phi + q_3 \cos\phi.$$

The corresponding displacements and Euler angles are

$$x_{gi} = x_{gi0} + \int u_{gi}\, dt,\ \phi = \phi_0 + \int \dot{\phi}\, dt \text{ etc} \tag{32}$$

The transformation to the geocentric system is given by

$$[x_{Gi}] = [C_{ij}]\ [x_{gj} + R_e\, \delta_i^3] \tag{33}$$

The altitude can be calculated from

$$H = R - R_e,\ R = \left(\sum_i x_{G_i}^2\right)^{1/2}. \tag{34}$$

If there is a tracking system at the launch station, it will be rotating with the earth at the rotational velocity Ω In the geocentric system this will contribute a displacement x_{L_i} given by

$$x_{L_i} = R_e\ [(\delta_i^1 + \delta_i^2)\cos\nu_0 + \delta_i^3 \sin\nu_0]\ [\delta_i^3 + \delta_i^1 \cos(\lambda_0 + \Omega t) + \delta_i^2 \sin(\lambda_0 + \Omega t). \tag{35}$$

The difference $(x_{G_i} - X_{L_i})$ gives the x_{G_i}-directional distance between the launch station and the vehicle. The corresponding radar range is

$$R_L = \left[\sum_i (x_{G_i} - x_{L_i})^2\right]^{1/2}. \tag{36}$$

This corresponds to an instantaneous longitude and latitude of the vehicle

$$\nu = \sin(x_{G_3} / R),\ \lambda = \sin^{-1}(x_{G_2} / R \cos\nu) \tag{37}$$

2.6. *Set theoretic sequencing logic*

The transformation relations and the equations of motion in different co-ordinate systems are independent of the sequence of occurrence because, if we make the elemental time-step size sufficiently small, every variable occurring on the right hand side in the expansion of a variable can be substituted by the values they had taken in the previous iteration, if such have not been calculated during the present iteration. However, a problem exists during the initiation of the first iteration. In the order-independent simulation, it is necessary to specify the initial values of every variable occurring on the right hand side of the statement. This is so cumbersome that the order that minimises the time of simulation should be determined and followed in the programme.

For practical simulation, we advocate the following set-theoretic sequencing logic, which minimises the time of simulation for given schemes of integration and interpolation.

The n functions constituting the core of the simulation are represented as a family of sets

$$S = \{S_1, S_2, \ldots, S_n\}, \quad S_i \subset S, \quad S_i = \{s_{i1}, s_{i2}, \ldots, s_{in_i}\}.$$

Let $$M = \bigcup_k S_k = \{s_1, s_2, \ldots, s_m\}.$$

Let the set of variables for which initial values are given, be

$$I_0 = \{q_1, q_2, \ldots, q_w\}, \; w < m.$$

Form a residual set

$$R = M \cap \bar{I}_0$$

where $\bar{I}_0$ is the complement of the set I_0.
Find an element, $a_1 \in R$, such that under the conditions

$$(a_1 \in S_j) \Longrightarrow (d_j = 1), (a_1 \notin S_j) \Longrightarrow (d_j = 0),$$

it is ensured that

$$\sum_{j=1}^{n} d_j \rightarrow \text{maximum.} \tag{38}$$

Form a new residual set

$$R_1 = R \cap \{\bar{a}_1\}.$$

Find an element, $a_2 \in R_1$, such that under the conditions

$$(a_2 \in S_j) \Longrightarrow (d_j = 1), \quad (a_2 \notin S_j) \Longrightarrow (d_j = 0),$$

condition (38) is again ensured.
This ordering is continued till the following ordered set is realised:

$$A = \langle a_1, a_2, \ldots, a_{m-w} \rangle, \tag{39}$$

such that $\{a_1, a_2, \ldots, a_{m-w}\} \equiv R.$

A further ordering is required to be carried out as follows. Let f be a 1—1 mapping function. Find that set, $Q_1 \subset S$, for which

$$f : (Q_1 \rightarrow a_{j_1}, \; a_{j_1} \in A, I_0 \cap Q_1 = \{a_{j_1}\}, \; j_1 \rightarrow \text{minimum}).$$

Form a new residual set

$$R'_1 = S \cap \bar{Q}_1.$$

Again find a set, $Q_2 \subset R'_1$, such that

$$f : (Q_2 \to a_{j_2}, a_{j_2} \in A, a_{j2} \neq a_{j_1}, I_0 \cup \{a_j\} \cap Q_2 = \{a_{j_2}\}, j_2 \to \text{minimum}).$$

This procedure is continued till the following ordered set is realised

$$B = \langle Q_1, Q_2, .., Q_n \rangle, \tag{40}$$

such that $\{Q_1, Q_2, .., Q_n\} = \{S_1, S_2, ., S_n\}$.

A feature of MATS which gives a better computational performance as compared to SIMSPACE I is the bucket concept which minimises the storage requirements by relating sets of data with the corresponding phase, so that as each phase of simulation is initiated the associated data are brought into the active memory. This beneficial feature is included in SIMSPACE II with the modifications found necessary to integrate it with the set theoretic sequencing logic, which is a more logical and efficient phasing scheme than that in MATS. With each distinct logical step outlined in § 2.6 is considered as a distinct phase and with it is associated a storage bucket, the capacity of which is proportional to the input data intensiveness of the logic step phase The input processor then handles these buckets of different capacity. The data may be fed in as input in any order and may be superseded It is read contiguously into the bucket and organised with respect to the above described logic-step phase when the simulation is requested.

2.7. *Interpolation and integration*

Input data like thrust, aerodynamic coefficients, moments of inertia and centre of pressure are functions of Mach number and angle of attack or time. While accuracy of simulation depends on the accuracy with which these values are interpolated, the time of simulation increases exponentially with the complexity of interpolation. A compromise between accuracy and complexity of interpolation is therefore called for.

In the development of SIMSPACE II special attention was given to the flexibility in the choice of the interpolation and numerical integration methods for each phase of simulation occurring in each iteration This necessitates not only making available a minimal set of properly selected interpolation and integration algorithms but also certain objective criteria built into the package for calling the appropriate algorithm for a given phase of the simulation.

The set of algorithms available in the present version of SIMSCRIPT II for one- or two- dimensional interpolation include well tested algorithms for step, linear, general polynomial with $n = 2$ to 5 and least oscillatory techniques.

The set of algorithms for numerical integration available presently include the simple finite difference integration with constant time step, the Runge-Kutta-Gill fourth order with fixed or variable step and Cowell eighth order with fixed or variable step using Runge-Kutta as a starter. During different phases of the simulation different algorithms which are most appropriate for the phase, are called in, For example, the Cowell algorithm is called in for free flights governed by second order differential equations. Sufficient flexibility has been provided in the package for adding more interpolation and integration algorithms if required.

Experience with both SIMSPACE I and SIMSPACE II indicates that the more frequently called algorithms are the simpler ones. For interpolation, it is the constant step linear algorithm whereas for integration it is the trapezium method. Typically the latter is found suitable for expressing $u_i(t)$, $q_i(t)$, $x_{gi}(t)$ and $\phi_i(t)$, e.g.,

$$X_{gi}(t) = X_{gi}(t-\triangle t) + \tfrac{1}{2}\triangle t\ [u_{gi}(t) + u_{gi}(t-\triangle t)].$$

This is not surprising because both accuracy and time of simulation depend on $\triangle t$. Experimental simulations have shown that the reduction of $\triangle t$ gives diminishing returns after an optimum value. $\triangle t$ determines not only the finite interval of elemental integration but also the number of times the entire set of equations of motion and transformations outlined in figure 1 are iterated. For more complex interpolation or integration algorithms, this increases the time of simulation exponentially with complexity. It is for this reason that SIMSPACE I uses mainly the least complex interpolation and integration schemes, decreasing $\triangle t$ to the extent necessary to meet the accuracy called for. However, SIMSPACE II allows for a wide variety of algorithms, which are at present chosen *a priori* for each general category of the phase of simulation. The choice depends upon earlier experience with the type of computation represented by the phase under consideration. It would, of course, be of much interest to evolve objective criteria for automatic choice of the algorithms for each phase, but at present this feature is provided only for general categories of call by the main routine.

3. Theoretical basis of integrated optimisation

Powerful methods are available for the optimisation of the trajectory, given a fixed vehicle profile as well as for the optimal staging, given a fixed trajectory. The problem of the simultaneous optimisation of staging and trajectory has not received the attention it deserves, atleast in literature. The problem is complex because the trajectory description of an n-stage vehicle in a three-dimensional six degree of freedom atmospheric flight is governed by the entire set of equations described in § 2, while the optimal staging is an involved exercise even for idealised trajectories. In view of this a hierarchic optimisation method was developed for SIMSPACE II which optimises the vehicle profile and trajectory in three successive stages as outlined below.

(i) Certain approximate 'thumb-rule' optimisations are carried out for the stage weight and the structure weight of the vehicle. From these, a first approximation profile of the vehicle is realised. Certain functions governing the trajectory are parametrised and the trajectory design is related to steering coefficients and the last stage propellant fraction. Using the Newton-Raphson method, stage sizing is carried out by a discrete step steepest descent.

(ii) Using the first-approximation solutions of step (i) as the initialisation point, a more accurate sizing optimisation as well as a more accurate and versatile method of parametric trajectory optimisation is carried out.

(iii) With the second approximation profile of step (ii) as the starting point, a technique called 'sensitivity-directed iterative simulation' based on the accurate simulation routine described in § 2 is used for enabling the final integrated optimisation of the profile and the trajectory to be carried out.

A brief description of each of these three steps in the hierarchic optimisation is given in the following subsections. For convenience and as there are no cross references to the equations, the notations of § 3 are kept distinct from those of § 2.

3.1. *The low level approximation*

Minimisation of the lift-off weight of an n-stage vehicle for a given payload requires the maximisation of the growth factor G, or equivalently the minimisation of $\Pi_{i=1}^{n} L_i$ where L_i is the payload ratio of the ith stage. This minimisation is constrained by the terminal velocity specification. Equivalently, the optimality condition is

$$v_i R_i L_i = \text{constant}, \tag{41}$$

where v_i and R_i are respectively the terminal velocity and mass ratio of the ith stage. The condition to be satisfied at the optimum is

$$R = \left[\Pi_{i=1}^{n} S_i^{v_i/v_i'}\right]^{-1} \left[\Pi_{i=1}^{n}\left(1-\frac{V_n}{v_i}\frac{L_n}{(S_n+L_n)}\right)^{v_i/v'}\right], \tag{42}$$

where S_i is the structural ratio of the ith stage, v' is the average terminal velocity of all the steps and R is the effective mass ratio given by

$$R = \left[\Pi_{i=1}^{n} (S_i + L_i)^{v_i/v'}\right]^{-1}. \tag{43}$$

From (42), L_n can be determined which when used along with (41) determines L_i $(i = 1, 2, \ldots, n-1)$.

Alternatively, as the hardware is generally costlier than the propellant, it is desirable to optimise the structure weight or equivalently the system structure factor $\alpha = SG$, where S is the total structure ratio. The optimal values can be obtained as solutions of the set of equations

$$Q = -\lambda \frac{S^2 G}{R v'},\ G = \left[\Pi_{i=1}^{n} L_i\right]^{-1}$$

and
$$\frac{(S_i + L_i)}{v_i L_i}\left[S_1 + \sum_{i=2}^{n}\left(S_i \Pi_{j=1}^{i-1} L_j\right)\right] = Q, \tag{44}$$

where λ is Lagrange's multiplier.

For the computation of the trajectory, a simplified model is introduced assuming a two-dimensional earth, a constant propellant flow for each stage, constant drag and a specific impulse variation with altitude governed by the relation

$$I_{SP} = I_{vac} - (I_{vac} - I_{SL})\left(\frac{P}{P_{SL}}\right),$$

where I_{vac} is the vacuum specific impulse, I_{SL} is the sea level specific impulse, P is the ambient pressure and P_{SL} is the sea level pressure

The ideal set of equations of motion at this low level approximation is the high speed trajectory equations (HSTE) of Bingham (1964). The equations of motion for two-dimensional motion in a spherical gravitational potential of a non-rotating earth are:

$$\dot{V} = T\cos\alpha - D - mg\sin\gamma,$$

$$\dot{\gamma} = \frac{V\cos\gamma}{R} - \frac{g\cos\gamma}{V} - \frac{T\sin\alpha}{mV}, \qquad (45)$$

$$R = R_e + h, \quad g = g_0\left(\frac{R_0}{R_0 + h}\right)^2.$$

Here V is the vehicle relative velocity, T the stage thrust, D the drag, m the mass, α the angle of attack, γ the relative flight path angle, R the radius vector, h the altitude, R_e the radius of the earth, g the local gravitational acceleration and g_0 the sea-level gravitational acceleration. To approximate the earth's rotational effect, an appropriate component of the earth's rotational velocity is vectorially added to V at first stage burnout, ie.,

$$V_x = V\cos\gamma + \Omega_0 R_0 \cos i, \; V_y = V\sin\gamma, \qquad (46)$$

$$V_I = (V_x^2 + V_y^2)^{1/2}, \; \gamma_I = \tan^{-1}(V_y/V_x),$$

where V_I is the vehicle inertial velocity, γ_I is the inertial flight path angle, Ω_0 is the earth's angular rotational velocity and i is the orbital inclination.

Trajectory optimisation is carried out by parametrising the functions upon which the trajectory depends. To do this a standard proven flight profile is selected, made up of the following segments: vertical rise, initial pitch-over for a short period of time, gravity turn to first stage burnout and linear tangent steering from first-stage burnout to trajectory end conditions:

$$\tan\psi = A - Bt,$$

where ψ is the vehicle inertial altitude referred to the launch horizontal. The parameters selected to describe this profile are an angle of attack to control the initial pitch rate and the two linear tangent steering coefficients A and B.

To meet trajectory end conditions, Bingham (1964) employs a modified Newton-Raphson three-variable iteration technique. The end conditions chosen are velocity, altitude and flight path angle. The control functions used are the two linear tangent steering coefficients and the propellant fraction of the last stage P_n. The Newton-Raphson technique for multi-dimensional functions is given by the matrix equation

$$[\delta \mathbf{f}] = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right] [\delta \mathbf{x}],$$

where $\quad \mathbf{f} = \{V, h, \gamma\}; \quad \mathbf{x} = \{P_n, A, B\},$

and the square matrix is composed of unique elements which are partial derivations of the elements of $\mathbf{f}$ with respect to the elements of $\mathbf{x}$. The $\delta\mathbf{x}$ term contains the control parameter increments while the $\delta\mathbf{f}$ term describes the errors in the trajectory end conditions. One of the problems which arises in obtaining a convergence for the trajectory is the sensitivity of the end conditions to the initial pitch rate. To improve the convergence characteristics and correct for the sensitivity problem, a weighting function K is employed as follows.

$$[\delta \mathbf{x}] = K\left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right]^{-1} [\delta \mathbf{f}]. \tag{48}$$

The form of the K factor is largely determined by experience. To fully automate the trajectory simulation, it was necessary to develop analytic approximations for estimating initial values of the last stage propellant fraction λ_n and the two linear tangent steering coefficients A and B. This is done by feeding the values obtained from the approximate sizing step.

3.2. *The middle-level approximation*

The sizing and trajectory optimisations described above are very approximate and the solutions derived from them are only intended to serve as initialisation values for more accurate methods used in the middle-level approximation. Initialisation values are important as they increase convergence and decrease computational time. The line of evolution of staging optimisation concepts found suitable for this middle-level approximation is that based on the concepts of Schurmann (1957), Subotowitz (1958), Hall & Zambelli (1958), Cobb (1961) and Adkins (1965). A general formulation of staging optimisation consolidated from their work is given in the following.

The thrust F, specific impulse I and the sea-level vehicle weight W are time functions and related by $F = -\dot{W}I$. The equation of motion is taken in the direction of the velocity vector

$$\dot{W} + GW = -JS, \quad J = C_D q/(I \cos \alpha), \quad G = \frac{F \cos \alpha - C_D qS}{WI \cos \alpha}, \tag{49}$$

where F is the thrust, α the angle between thrust and velocity vector, C_D the drag coefficient, q the dynamic pressure and S the reference area for drag.

The optimal trajectory obtained in the low-level optimisation is used to determine J and G as functions of time which renders (1) into a first order differential equation in W. The solution to (1) for the jth stage is

$$W_{jb} = W_{ji} \exp(-N_j) - Q_j S_j, \tag{50}$$

where $N_j = N(t_{ji}, t_{jb}) = \int_{t_{ji}}^{t_{jb}} G(t)\, dt,$

$$Q_j = Q(t_{ji}, t_{jb}) = e^{-N_j} \int_{t_{ji}}^{t_{jb}} \exp[N(t_j, t)]\, J(t)\, dt.$$

The subscripts b and i refer to burnout and ignition times respectively. For the jth stage the thrusting duration or burning time, $\bar{t}_j = t_{jb} - t_{ji}$, must be selected so that $(1/I \cos \alpha)$ is defined for all t on this interval. The burnout value is

$$-\dot{W}_{jb} = G_j W_{ji} e^{-N_j} + (J_j - G_j \cdot Q_j) S_j, \tag{51}$$

where G_j and J_j are at burnout

Propellant and payload weights for the jth stage are related by

$$W_{jb} = P_j = (W_{ji} - P_j)(1 - \lambda_j), \tag{52}$$

where P_j is the payload, i.e., $P_j = W_{(j+1)i}$, and λ_j is the propellant mass fraction for the jth rocket motor.

From (50) and (52), the payload ratio of the jth stage μ_j is given by

$$\mu_j = \bar{\mu}_j/(1 + Q_j S_j/\lambda_j P_j),$$

$$\bar{\mu}_j = e^{-N_j}/\lambda_j - 1/R_j, \tag{53}$$

$$R_j = \lambda_j/(1 - \lambda_j).$$

The overall payload/gross weight ratio is

$$P_n/W_{1i} = \Pi_{j=1}^{n} \mu_j, \tag{54}$$

where P_n is the final payload. It is now required to determine the optimum set of velocity increments (v_j) of the jth stage which maximise P_n/W_{1i} subject to the constraint of a known total velocity increment,

$$V_T = \sum_{j=1}^{n} v_j.$$

The propellant flow rate $(-\dot{W})$ is an arbitrary function of time and can be related to W_j and $\bar{t}_j$ by the dimensionless function $\eta(t)$. At burnout for the jth stage,

$$-\dot{W}_{jb} = \eta_j (W_{ji} - W_{jb})/\bar{t}_j. \tag{55}$$

From (51) to (55),

$$t_j = \frac{\eta_j R_j (1-\mu_j)}{G_j (1+R_j \bar{\mu}_j) + (J_j/Q_j - G_j)(\bar{\mu}_j - \mu_j) R_j}. \tag{56}$$

Using this expression and well-known techniques of Lagrangian multipliers it is possible to arrive at the optimal staging values. The optimal staging values so arrived at may in turn be used to optimise the trajectory further.

The optimal trajectory values so obtained are in turn fed back to the above staging optimisation procedure. This iteration is continued until both the optimal staging and the optimal trajectory converge to a unique set of values. To facilitate this, the parametric approach adopted in § 3.1 must be refined for better accuracy and convergence. After a number of trial-and-error experiments, the following approach based on the better features developed by Powell (1969), Hestenes (1969), Karacsony & Cole (1970), Buys (1972), Brusch & Peltier (1974) and Brusch (1976) was adopted.

Following Brusch the general formulation of trajectory optimisation leads to the following non-linear programming problem.

$$\text{Minimise } I(\mathbf{x}) \text{ subjected to } \mathbf{h}(\mathbf{x}) = 0 \text{ and } \mathbf{g}(\mathbf{x}) \geqslant 0, \tag{57}$$

through the following definitions. Let the vector of all parameters subject to optimisation be

$$\mathbf{x} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_w, d_1, d_2, \ldots, d_q\}, \tag{58}$$

where $\mathbf{a}_k$ is the vector of parameters associated with the parametric control model for u_k, $u_k(t) = V_k(a_k, t)$. The design variables d are variables to be optimised which are constant with respect to time-like gross lift-off weight and propellant mass fraction. Let the number of parameters in the x vector be N_x.

It is advantageous to express the constrained variables as equivalent free variables to enable the use of unconstrained function optimisation methods.

If $\xi \geqslant 0$ is the sth state variable inequality constraint then consider the transformation

$$T_s = \min_t \, [\xi(t)] \left[\int_{t^0}^{t^f} A \exp\{-B_{\max} [\min(\xi(t), C_{\max}), C_{\min}]\} \, dt \right], \tag{59}$$

where A, B and $C_{\max}$ are positive constants and $C_{\min}$ is a negative constant. Equation (59) is a functional that transforms ξ_s into a number T_s such that T_s is negative if the constraint is violated and positive otherwise. This transformation enables the use of unconstrained optimisation methods outlined below.

The above non-linear programming problem can be solved by the method of multipliers. If (λ_k, μ_k) and (K_k, C_k) are respectively the equality and inequality constraint multipliers and equality and inequality constraint quadratic multipliers, we can construct an augmented objective function of the form

$$J(x) = I(x) + \sum_{k=1}^{N_h} [\lambda_k h_k + K_k h_k^2] + \sum_{k=1}^{N_g} \begin{bmatrix} \mu_k g_k + C_k g_k^2, \text{ if } \mu_k + 2\, C_k g_k \leqslant 0 \\ -\mu_k^2 + 4\, C_k, \text{ if } \mu_k + 2\, C_k g_k > 0 \end{bmatrix}. \tag{60}$$

The broad outline of the computational procedure consists of the following steps.

(i) Initial values of x, C, λ and μ are obtained from the low-level approximation step. Select tolerable errors for each of the constraints and positive control tolerances. Set $j = 1$; evaluate $J_0(x)$.

(ii) Perform unconstrained function minimisation of $J_j(x)$ through iteration.

(iii) Test for solution convergence.

(iv) Update the Langrange multipliers using the approach suggested by Buys (1972) and Hestenes (1969).

(v) Change the penalty constants as required using the approach suggested by Powell (1969)

(vi) Set $j = j + 1$ and return to step 2.

The above method avoids the severe functional distortion characteristic of the penalty function approaches and do not require return to a feasible solution after each optimisation step as required by the gradient projection and related methods. Both these features tend to reduce the computational load significantly.

3.3 *The higher approximation*

One of the methods of optimisation uses repeated trial-and-error simulation. The simulation scheme of § 2 being near exact and comprehensive, optimisation as a process of repeated simulation would consume prohibitively more computer time. In spite of this, desired levels of accuracy require the use of simulation routines atleast as the last step in the optimisation. The solutions are therefore required to be brought as near the exact as possible before simulational optimisation is resorted to. The solutions of the middle-level approximation obtained from the procedure of § 3.2 is found to be close enough to appreciably reduce the total computational time for the hierarchic optimisation.

With the middle-level approximation serving as the initialisation for the simulational optimisation step, the following algorithm is employed for the final step.

(i) The solutions from the middle-level approximation step are taken to be the initial set of values.

(ii) Sensitivity of the objective function J is worked out for each design variable like azimuth, launch angle, and coasting time.

(iii) The design variables are perturbed one at a time in the sequence of the ascending order of sensitivity While a design variable is perturbed so as to minimise J, all the other variables are held constant at their previous values. For each perturbation, the simulation is carried through once. From a trend analysis the perturbation step can be estimated, which will minimise the number of simulational iterations.

(iv) The gradient can be estimated using the same technique described in (57) to (60). The only difference is that it is sensitivity-directed and the entire simulation is carried out for each iteration as against the use of simplified two-dimensional trajectory equations in the middle-level approximation.

(v) A design variable is perturbed until J attains a minimum value relative to the fixed values of the other variables and subject to the specified constraints. The set of variables are perturbed one after the other cyclically until J saturates at a minimum value within the tolerance limit specified.

The above 'sensitivity-directed iterative simulation' approach enables the integrated optimisation of profiles and trajectory. If we designate the above algorithm with respect to the objective function J, the set of constraints C and the set of design variables V as Alg (J, C, V) then the integrated optimisation will be the cyclic iteration of Alg (J', C', V') and Alg (J'', C'', V'') where the former refers to the trajectory optimisation and the latter to the profile optimisation.

4. Conclusions

The theoretical basis of the evolution of SIMSPACE II has been outlined providing a framework for including the better features of the packages SIMSPACE I and MATS. In addition, the following new features have been incorporated for increasing the scope and versatility of simulation and optimisation of satellite launch vehicles.

First, by the set-theoretic sequencing logic of SIMSPACE I is oriented to accommodate the MATS bucket concept.

Secondly, a hierarchic optimisation procedure is advocated for increasing convergence and decreasing computational time of the integrated staging-trajectory optimisation.

Lastly, a 'sensitivity-directed simulational iteration' approach to the integrated optimisation is suggested to include not only the finer features of the profile of various stages but also the three-dimensional 6-degree of freedom atmospheric trajectory including refined computations responding to drag, wind effects, control schemes etc.

An important feature of SIMSPACE II is that it has a general main routine structure which is independent of particular sets of interpolation, integration and optimisation subroutines. Facility is provided for including any set of hierarchic optimisation methods, a bank of interpolation and integration procedures along with the governing criteria for automatic call from the main routine. This enables a constant improvement of the package as better interpolation, integration and optimisation algorithms become available.

SIMSPACE II also has a number of other features which are not described here as they are not related to the main theme of this paper, e.g., the methods for including range safety constraints. As these have independent theoretical basis, they will be described elsewhere.

References

Adkins C N 1965 *J. Spacecr Rockets* **2** 249

Bingham R H 1964 Tech Note TN-64-BP-04 (General Dynamics Inc, California, Convair Division Aeroballistics)

Brusch R G and Peltier J P 1974 *Acta Astronaut* **1** 1453

Brusch R G 1976 *IEEE Conf. Decision and Control* (California: Clear Watch Beach)

Buys J D 1972 Ph.D. dissertation (Netherlands. University of Leiden)

Cobb E R 1961 *Am. Rocket Soc. J.* **31** 342

Fogarty L E 1968 *Simulation* **11** 187

Hall H H and Zambelli E D 1968 *Jet Propul* **28** 463

Harris R J 1963 NASA Rep. TN-D-183
Hestenes M R 1969 *J Optim. Theory Appl.* **4** 303
Karasony P J and Cole C E 1970 AIAA Paper p. 70
Krause H G L 1961 *Am. Rocket Soc.* Rep. p. 2073
Lanzano B C (ed) 1970a *MATS programmer's handbook* (Calif TRW Systems Group)
Lanzano B C 1970b *J. Spacecr. Rockets* **7** 1132
Powell M J D 1969 *Optimisation* ed. R Fletcher (London Academic Press)
Schurmann E E H 1967 *Jet Propul.* **27** 863
Seshagiri N, Janardana Rao Y and Rajendra Prasad M 1969 TIFR Spl Rep. (also Rep. No. SSTC-ARD-TR-5-70)
Subotowicz M 1958 *Jet Propul* **28** 460