A GENERAL ALGORITHM FOR THE OPTIMAL COORDINATION OF A SPACE-BORNE COMPUTER-TRANSMITTER COUPLING

N. Seshagiri and P. Sadanandan

Computer Group, Tata Institute of Fundamental Research, Colaba, Bombay, India

An analysis is made of a situation that may arise in space to earth communication, viz., the establishment of the priority of transmission of messages that flow out of a space station taking into consideration factors like the noise level of the channel, the length of the message, and the importance as well as the effective content of the message relative to other messages received at the earth station.

A coupling of the transmitter with a special purpose computer is suggested, giving an algorithmic schemata for their optimal coordination. Requirements of a computer simulation of the coupling utilizing the SIMSCRIPT language are outlined.

1. INTRODUCTION

Scheduling of messages from a space station to an earth station is subject to numerous constraints [1, 2]. The more important constraints are the noise level of the channel which affects the clarity of deciphering the message at the earth station, the length of the message, the importance of the message and the effective information content of the message relative to other messages received earlier at the earth station. Thus when several messages are received simultaneously, there exists the problem of assigning priorities for the transmission of the messages. Such an assignment becomes especially important when control commands transmitted to the space station from the earth station are based on the messages received. Such scheduling problems do not have simple solutions. Consequently, a need is felt for a special purpose computer which can execute the scheduling algorithm.

The coupled system, comprised of the transmitter, the computer and the associated equipment, is given as a block schematic in Figure 1.

Messages are assumed to come from n distinct sources. The computer receives these and stores them in the memory. Each source is associated with a set of descriptors guiding the criteria of priority. Their magnitudes are controlled by the earth station where the earlier messages are analyzed and new magnitudes governing the priority are transmitted back to the scheduling computer.

An important aspect of scheduling, in practical cases, is the interdependence of the messages. In other words, the priority of each message is relative to the

Contributed to the Proceedings of the Symposium on Computer Processing in Communications, Polytechnic Institute of Brooklyn, April 8–10, 1969.

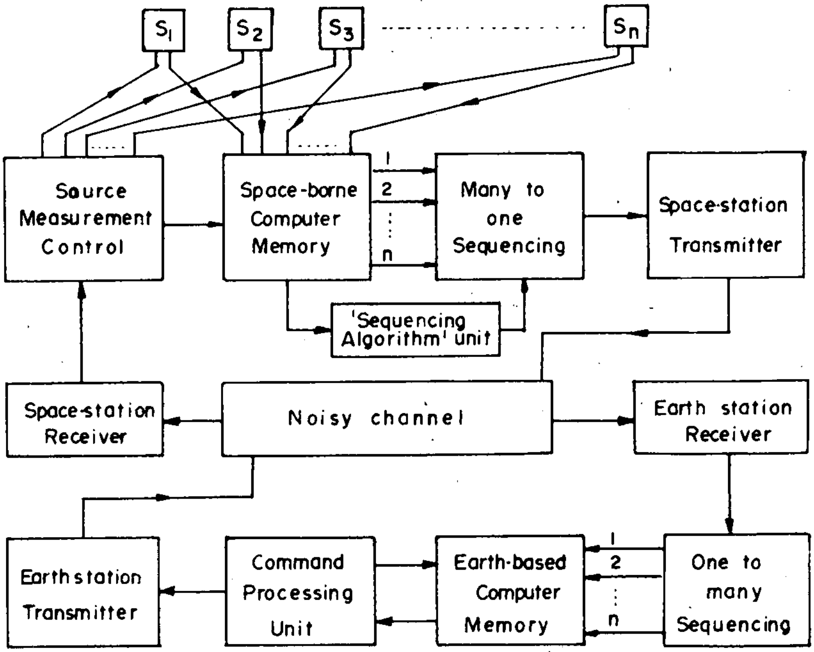


Fig. 1. Schematic of the computer-transmitter coupling.

sequence of messages already received at the earth station. The scheduling algorithm becomes nontrivial for the practical case wherein the feed-back time lag in the feed-back loop of Fig. 1 is large compared to the average rate of arrival of messages.

In what follows, the criteria of priority is discussed in detail. A subsequent section describes the scheduling requirements in its generality. This is followed by an algorithm for the optimal scheduling of the messages. In the last section, requirements of a computer simulation utilizing SIMSCRIPT are outlined.

2. CRITERIA FOR DECIDING PRIORITY

As mentioned earlier, the assignment of priority to messages will be subject to numerous constraints. The inclusion of all these would call for a complicated scheduling algorithm which, in turn, requires a more sophisticated computer. One of the criteria of space technology being the minimization of the payload, it becomes necessary to simplify the scheduling algorithm to the maximum extent possible. This would require a proper choice of the constraints with a view to minimize their number. Four of these affect the assignment of priorities to an extent large enough for most practical purposes. These are described below.

A. Importance of the Message

Every message transmitted from the space station has some utility at the earth station and some at the space station. The former is not important for the assignment of priority, as it is seldom governed by limitations of time. On the other hand, the latter is of much significance, as the earth station assesses the information concerning the functioning of the space station from the received messages. Depending upon the current list of messages received, adequacy or otherwise of information pertaining to different aspects of the functioning of the space station will be analyzed. If the picture is inadequate regarding an aspect which is crucial, a control command will be transmitted back to the space station thereby reassigning the priorities to the types of messages sent by the latter. Thus importance of a message needs quantification.

The following typical example serves as an illustration. A measurement on the ionizing power of cosmic rays is proposed to be studied at a region in space with inadequate knowledge of the orders of magnitude involved. Different techniques of measurement are proposed to be utilized in different conditions. The exact nature of the conditions involved can be ascertained only by a detailed analysis of the measurements obtained. It may prove too costly to realize this at the space station itself. Therefore the measurements are transmitted to earth where they are processed. A control command from the earth station decides the particular technique to be employed at a given time. As the region at which the measurement is required to be carried out is well-delimited, if the space station is moving around an orbit, a time limit is set in making the decision. Thus the earth station has to specify a high priority for this source of information earlier to the space station entering this region.

It can be seen from the foregoing that quantification of importance depends on the time limits involved. In the above example, they are the time lag in the feedback loop and the time of transit in the region by the space station. Since importance of source of information is relative to that of other sources of information, a cost should be associated with each source. In general terms, importance of a message can be quantified as given below.

The importance of a message should not bring in the concept of "meaning" of message. This is because a classification of the messages based on meaning would involve a prohibitive amount of retrieval and comparing operations in the space-borne computer itself. This tends to increase the complexity of this computer apart from the disadvantages arising out of increased time of computation. Therefore we assume that the importance of a message is tantamount to the importance of the source as a function of time.

The derivations to follow are based on the following quantities:

S_i—The i^{th} source of information

t—The time at which the priority list is required to be established

$t_{\max,i}$—The time at which the decision about the average rate of transmission of the messages from S_i is required at the space station

t_{es}—The time of transit of the message from the earth station to the scheduling computer

t_{se}—The time of transit of the message from the space station to the control and command generator of the earth station

T—The time duration for which the current scheduling is applicable. Obviously, this is given by $\underset{i}{\text{Max}} \{t_{\max,i}\} - t$

T_c—The upper bound for the time of computation for arriving at the optimal schedule

T_i—Total time of computation required for processing the N_i messages at the earth station

τ_i—Time of receiving the final command from the earth station in excess of $t_{\max,i}$

$\tau_{\max,i}$—Upper limit of the excess time τ_i

r_i—The average rate of transmission of messages from S_i

$\overline{r_i}$—The average rate of arrival of messages at the space station

N_i—The number of messages received at the earth station from source S_i

$N_{\min,i}$—The minimum number of messages from S_i required for drawing decisive inferences

$\overline{N}_i(t)$—The number of messages from S_i accumulated in the space-borne computer till time t

$N_i'(t)$—The number of messages from S_i accumulated at earth station computer till time t

$P_i(\tau_i)$—The penalty function associated with source S_i as a function of τ_i.

If all messages could reach the earth station at such times that the final command from the earth could reach the space station before $t_{\max,i}$, for each i; the priority problem will not exist. However, in practice, this need not be so. Situations do arise where it is impossible to meet the time limits specified for every S_i. A compromise solution is therefore called for. Such a compromise solution requires the definition of an objective function which could be optimized. Since τ_i would reflect itself as a penalty cost, one can define a penalty function for S_i, viz., $P_i(\tau_i)$. Clearly then, the sum total of all penalty functions over all the sources should be minimized with respect to τ_i. Therefore, the objective function is

$$F = \sum_{i=1}^{n} P_i(\tau_i) \quad . \tag{2.1}$$

This minimization of the objective function is constrained because of the restrictions imposed by the considerations of time. The first constraint is that

$$0 \le \tau_i \le \tau_{\max,i} \qquad (\text{for all } i) \quad . \tag{2.2}$$

This is so because it is immaterial whether a command comes earlier than $t_{\max,i}$ for source S_i as long as the execution of this command at the space station is not possible before $t_{\max,i}$.

Another constraint is imposed by the difference in the number of messages arriving at the space station and at the earth station. The former is given by

$N_{1i} = \overline{N}_i(t) + (T + \tau_j)\,\overline{r_i}$, where S_j is the source for which $t_{\max,i}$ is maximum; the latter is given by

$N_{2i} = (t'_{\max,i} + \tau_i)\, r_i$, where $t'_{\max,i}$ is the net time duration available for transmitting the messages from the i^{th} source and is given by

$$t'_{\max,i} = t_{\max,i} - t - t_{se} - t_{es} - T_c - T_i \quad .$$

Thus the inequality constraint is

$$(t'_{\max,i} + \tau_i)\, r_i \leq \overline{N}_i(t) + (T + \tau_j)\, \overline{r_i}, \qquad \text{where } r_i \text{ is a variable} \quad . \tag{2.3}$$

As there exists a minimum number of messages from each source required for drawing decisive inferences, it is necessary to assure that the number of messages received at the earth station exceeds this minimum number. However, at an arbitrary time, t, a subset of these messages numbering $N_i'(t)$ would already be present at the earth station. Therefore, the inequality to be satisfied is

$$N_{\min,i} - N_i'(t) \leq N_{2i} \quad .$$

In terms of the rate of transmission and time duration the above inequality can be written as

$$N_{\min,i} - N_i'(t) \leq (t'_{\max,i} + \tau_i)\, r_i \quad . \tag{2.4}$$

The average length of a message from source S_i also influences the rate of transmission of the messages from this source. For purposes of this communication, a single channel transmission is assumed because the concepts and derivations made here can easily be extended to multi-channel transmission.

Let $T_{av}(\ell_i)$ be the time of transmission of a message of average length ℓ_i from S_i. The time taken to transmit all the messages required at the earth station is given by

$$T_t = \sum_{i=1}^{n} [r_i(T + \tau_j)\, T_{av}(\ell_i)] \quad .$$

It follows that a constraint

$$T_t \leq T + \tau_j \quad ,$$

should also be satisfied. Cancelling $(T + \tau_j)$, the inequality becomes

$$\sum_{i=1}^{n} [T_{av}(\ell_i)\, r_i] \leq 1 \quad . \tag{2.5}$$

B. Effective Content of Messages

The possibility of the existence of redundancy in the messages received at the earth station, may effectively reduce N_i and hence r_i. If the redundancy α is expressed as a ratio, the effective rate of transmission is given by

$$r_i' = (1 - \alpha)\, r_i; \qquad 0 \leq \alpha \leq 1 \quad . \tag{2.6}$$

In Eqs. (2.3), (2.4) and (2.5), r_i should be replaced by $(1 - \alpha)\, r_i$.

C. Noise in the Channel

Similar to the above, the noise in the channel may also reduce the number of useful messages received at the earth station. The reduction in the number of messages or in the rate of transmission β, can be expressed as a ratio and may be determined from the ratio of the number of messages that could be deciphered to the total number. The effective rate of transmission is given by

$$r_i'' = (1 - \beta)\, r_i' = (1 - \alpha)(1 - \beta)\, r_i \quad . \tag{2.7}$$

Therefore, in Eqs. (2.3), (2.4) and (2.5), r_i should be replaced by $(1 - \alpha)(1 - \beta)\, r_i$.

D. Length of the Message

The average length of the message ℓ_i from source S_i, influences the ratio β. An increase in the value of ℓ_i increases the value of β. From a detailed knowledge of the characteristics of the channel and the source, it is possible to obtain, *a priori*, a function $f(\ell_i)$ which gives the effective value of β, viz.

$$\beta' = f(\ell_i)\beta \quad . \tag{2.8}$$

This calls for a replacement of β in Eq. (2.7) by $f(\ell_i)\beta$.

The optimization problem formulated in the foregoing is a typical nonlinear programming problem with quadratic constraints. It can be summarized as:

$$\text{Maximize} \qquad \sum_{i=1}^{n} P_i(\tau_i) \quad ,$$

subject to

(1) $0 \le r_i$

(2) $0 \le \tau_i \le \tau_{\max;i}$

(3) $\mu_i\, r_i(\tau_i + t'_{\max,i}) \ge N_{\min,i} - N_i'(t)$

(4) $\mu_i\, r_i(\tau_i + t'_{\max,i}) \le \overline{N}_i(t) + \overline{r}_i(T + \tau_j)$

(5) $\sum_{i=1}^{n} [T_{av}(\ell_i)\, r_i] \le 1 \quad ,$

where $\mu_i = (1 - \alpha)\,[1 - f(\ell_i)\beta]$ and j is the index of the source with the maximum $t'_{\max,i}$.

3. A SIMPLIFIED ALGORITHM

In Fig. 1, though computers are assumed to exist in both the stations, reference is made only to the conjoint function of both throughout the formulation of the nonlinear programming problem. No mention has been made regarding the precise role of the individual computers. This is because the formulation is dependent only upon the conjoint function of the computers at both the stations. The situation will be very different when one considers the solution of the mathematical programming problem. Two cases of interest suggest themselves:

Case 1. At the earth station, messages are analyzed and priorities are established by solving the mathematical programming problem. The priority list is transmitted to the space station for execution.

Case 2. At the earth station, messages are analyzed and the values of α, $f(\ell_i)\beta$, etc., are transmitted periodically. The computer at the space station solves the mathematical programming problem, establishes the priority list, and executes it.

The former is more viable if an investment in reliable microelectronic computers is not planned for. The solution of the nonlinear programming problem can be carried out utilizing any of the available techniques taking the least time of computation, irrespective of how complicated the algorithm is. The priority list so realized should be transmitted several times in order to compensate for the loss of information due to noise in the channel. This requires noise filters along with a decoding unit at the space station. A computer is still needed at the space station to store the incoming messages as well as for the decoding operations.

The latter case on the other hand, will become more and more viable as microelectronic computers become smaller and cheaper. The turning point for adapting Case 2 would be when the sum of the costs of losing or misinterpreting the priority list sent from the earth and the investment in assuring reliability of its transmission exceeds the cost of launching the additional hardware required for executing the scheduling algorithm. In Case 2 the scheduling algorithm is required to be as simple as possible. For this case the objective function can be linear in practice, though the constraints are quadratic. Even though algorithms like that of Box [3] exist for realizing optimal solutions in such a case, they are not only time consuming but complicated too. Complications in algorithms necessarily call for increased sophistication of the space-borne computer and hence increased weight. As this is an undesirable situation, a barter between accuracy and simplicity was worked out. From this a simplified algorithm was developed which is described in the steps below.

1. The case for which all τ_i are zero occurs when the minimum number of messages required at the earth station is satisfied. The rate for the limiting condition is

$$r_{i1} = \frac{N_{\min,i} - N_i'(t)}{\mu_i\, t'_{\max,i}} .$$

2. This rate is substituted into the left hand side of Inequality (2.5). Thus

$$M_1 = \sum_i T_{\rm av}(\ell_i)\, r_{i1} .$$

If $M_1 \leq 1$ as implied by Inequality (2.5) and if Inequality (2.4) is satisfied, then

$$r_i = r_{i1}, \quad \text{and} \quad \tau_i = 0 .$$

3. If Inequality (2.4) is satisfied but $M > 1$, this indicates that the initial assumption of zero τ_i is not valid. If the linear objective function is given by

$$F = \sum_{i=1}^{n} k_i \tau_i ,$$

the value of r_{i1} should be modified such that

(a) The value of r_{i1} is increased if k_i is greater than the average of $\{k_1, k_2, \cdots, k_n\}$. This is required, because, if $k_j > k_i$ then for a minimization of F more weightage is given for decreasing τ_j than τ_i.

(b) The limiting case of inequality (2.5) should be satisfied because it gives the nearest acceptable value for M_1, viz., unity.

To a first approximation, the above two conditions are satisfied by the solutions

$$r_i = \frac{r_{i2}}{M_2} ,$$

where

$$M_2 = \sum T_{av}(\ell_i)\, r_{i2} ,$$

$$r_{i2} = \frac{n k_i}{\sum_i k_i} \frac{r_{i1}}{M_1} ,$$

and

$$\tau_i = \frac{N_{\min,i} - N_i'(t) - \mu_i r_i t'_{\max,i}}{\mu_i r_i} .$$

4. If either Ineq. (4) is not satisfied or $\tau_i > \tau_{\max,i}$ or

$$\overline{N}_i(t) + \overline{r}_i(T + \tau_j) < N_{\min,i} - N_i'(t) ,$$

then the demand of information from the earth station exceeds the supply from the space station. The only solution for this case is to wait for the maximum possible time, viz., $(t'_{\max,i} + \tau_i)$ and process the command from insufficient data. For this case the solutions are

$$\tau_i = \tau_{\max,i}, \qquad r_i = \frac{\overline{N}_i(t) + \overline{r}_i(T + \tau_j)}{\mu_i\,(t'_{\max,i} + \tau_{\max,i})} .$$

4. SIMULATION OF THE COUPLING USING SIMSCRIPT

The computer-transmitter coupling can be designed conveniently by simulating the entire system on a digital computer. The requirements of such a simulation utilizing the SIMSCRIPT language on CDC-3600 is briefly outlined in the following ten steps.

1. Information Source Simulation

The entire design of the coupling is based on the physical requirements and limitations of the information sources like for example the range, sensitivity, and accuracy of measurements. For the purpose of simulation it is assumed that an information source, in general, can be represented as a sequence of sets of numbers. Each set is assumed to be describable by the following structure.

SOURCE CODE	NUMBER OF TYPES	MEASURE-MENT CODE	BIASED RANDOM NUMBER	MEASURE-MENT CODE	BIASED RANDOM NUMBER	•••	MEASURE-MENT CODE	BIASED RANDOM NUMBER

Each measurement-code corresponds to a distinct type of measurement of a particular source. Each source is assumed to be comprised of a finite prespecified number of types of measurements. A measured magnitude is associated with each measurement code. It is assumed that the range of variation as well as the distribution of the magnitude and the time interval between successive messages from each source are known *a priori*. In this simulation experiment random numbers are generated within the specified limit and biased to correspond to the specified distribution. The types of measurements considered during a schedule period are obtained from the command sent from the earth station. Since the number of these types considered during each schedule-period may be different, after the source code, the number of types included is also given. This number decides the length of the message $T_{av}(\ell_i)$ for source S_i.

For illustrating the variable length of the message as well as parallel occurrence of messages from different sources the following SIMSCRIPT routine is provided in three distinct parts.

The initial condition deck and the definition forms are not included here.

Part 1. *Exogenous Event DEFIN.* This is called at the start of each schedule to define the message length of each source and the corresponding measurement codes. The data for this should be supplied on Exogenous Event tape. The SIMSCRIPT routine and the corresponding description of entities, events and attributes are given in Table I.

Part 2. *Subroutine SCHDLE.* Scheduling the parallel transmission of the messages from the sources between the time interval TIME (current time) and TMAX (upper limit) is done by this subroutine. The SIMSCRIPT routine and the corresponding description of entities, events and attributes are given in Table II.

Part 3. *Endogenous Event SEND.* This event routine sends the message from a particular source as it is scheduled by the scheduling routine SCHDLE. The biased random numbers are generated and stored in the message block. A variable length structure of the message is taken into consideration. The SIMSCRIPT routine and the corresponding description of entities, events and attributes are given in Table III.

TABLE I. SIMSCRIPT Routine.

SIMSCRIPT ROUTINE

```
  EXOGENOUS EVENT DEFIN
  DO TO 1 FOR EACH SORCE I
  READ SCODE (I), NTYP
  FORMAT (2I8)
  IF (NTYP) EQ (NTYPE (I)), RETURN
  IF (NTYP) LS (NTYPE (I)), GOTO 3
  LET N = NTYP - NTYPE (I)
  DO TO 2, FOR J = (1) (N)
  CREATE WORDS
  READ MCODE (WORDS)
  FORMAT (I8)
  FILE WORDS IN LSET (I)
2 LOOP
  GOTO 5
3 LET N = NTYPE (I) - NTYP
  DO TO 4, FOR J = (1) (N)
  REMOVE WORDS FROM LSET (I)
  DESTROY WORDS
4 LOOP
5 LET NTYPE (I) = NTYP
1 LOOP
  RETURN
  END
```

DESCRIPTION

(1) DEFIN—Definition of message length and measurement codes for each schedule.
(2) SORCE—Permanent entity.
(3) SCODE (SORCE)—Source code. An attribute of source.
(4) NTYPE (SORCE)—Number of types of message—Attribute of source initially set to zero.
(5) WORDS—Temporary entity having attributes:
   a) MCODE—Measurement code—Integer type
   b) BRAN—Biased random number—Floating.
(6) LSET (I)—Single subscripted set for each source. The members of LSET (I) are the WORDS belonging to the I^{th} SORCE. The number of members in LSET (I) is given by NTYPE (I) LSET (I) is LIFO set (last-in-first-out).

NOTE:

This routine must be called at the beginning of each schedule to define the length of the message and to allocate the storage for measurements. The source codes, number of types and the particular measurement codes are read in from Exogenous Event tape.

TABLE II. SIMSCRIPT Routine.

SIMSCRIPT ROUTINE

```
  SUBROUTINE SCHDLE (TMAX)
  LET TO = TIME
5 LET T = TO + TINTR
  IF T GR TMAX, RETURN
  DO TO 1, FOR EACH SORCE I
  LET DT = TO
2 LET DT = DT + RAN (I)
  IF DT GR T, GO TO 1
  CREATE SFLAG
  STORE I IN SCD (SFLAG)
  STORE DT IN TOO (SFLAG)
  FILE SFLAG IN SCHS
  GO TO 2
1 LOOP
3 IF SCHS IS EMPTY, GO TO 4
  REMOVE FIRST SFLAG FROM
  SCHS
  CREATE SEND
```

DESCRIPTION

(1) SCHDLE—Scheduling of messages sent from the sources within the period (TIME to TMAX).
(2) RAN (I)—Permanent attribute, single dimensioned, whose value is defined by random look-up tables.
(3) SCHS—Schedule set (Ranked) whose members are the SFLAG entities containing information about the occurrence of a source at a particular time. This is lowest ranked with respect to attribute TOO (time of occurrence).
(4) SEND—Event notice which sends the message from a particular source. This has the following two attributes:
   a) SFLG—Source flag which specifies the particular source and the time of sending the message

TABLE II (*Continued*)

SIMSCRIPT ROUTINE

```
  STORE SFLAG IN SFLG (SEND)
  CAUSE SEND AT TOO (SFLAG)
  GO TO 3
4 LET TO = T
  GO TO 5
  END
```

DESCRIPTION

b) And the time of occurrence (which is automatically defined by SIMSCRIPT).

(5) TMAX—Upper limit for simulation time.

(6) SFLAG—Temporary entity having attributes:

a) Source code (SCD)

b) Time of occurrence (TOO). This entity is a member of a set SCHS which is FIFO. This is introduced to make the scheduling easier.

(7) TINTR—System attribute specifying the time interval. The simulated time is divided into these intervals for scheduling.

(8) SCD—Code for the source$_i$—Type integer.

(9) TOO—Time of occurrence of sending the message from source$_i$.

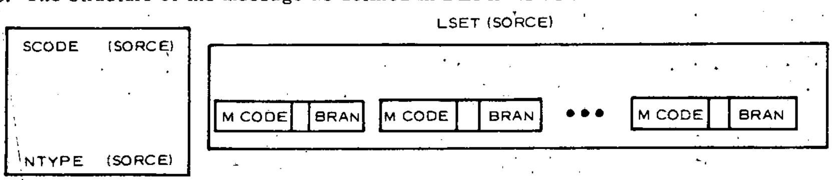
TABLE III. SIMSCRIPT Routine.

SIMSCRIPT ROUTINE

```
  ENDOGENOUS EVENT SEND
  STORE SFLG (SEND) IN SFLAG
  DESTROY SEND
  STORE SCD (SFLAG) IN ISORS
  DO TO 1, FOR EACH WORDS OF LSET (ISORS)
  LET BRAN (WORDS) = BRND (ISORS)
1 LOOP
  RETURN
  END
```

DESCRIPTION

1. SEND—Sending the messages from source.
2. SCD—Source code. Attribute of SFLAG. See SCHDLE routine.
3. LSET (I)—Is a LIFO set (see DEFIN/routine). For the source I whose members are the "WORDS" which constitute the message for source I.
4. BRAN—Attribute of "WORDS".
   The first attribute of "WORDS" is the MCODE—Measurement code which is read in by DEFIN. The second attribute of "WORDS" is BRAN where a biased random number is generated by SEND.
5. The structure of the message as defined in DEFIN is as follows:



6. BRND—A single subscripted (for each source) permanent variable whose value is defined by random look-up table facility.
7. SFLG—An attribute of SEND which contains the source identification.

2. Space Computer Storage Simulation

The simultaneous events described above require different storage allocations for different sources. As the length of the message varies from schedule to schedule, provision should be made for variable word length so as to affect the hardware economy. Since the command transmitted from the earth has as one of its components the values of $(N_{\min,i} - N'_i)$ for the next schedule period, the number of words for each source S_i can be taken to be equal to this number. If the storage stack for S_i is not completely filled up, a push-down structure is assumed. However when it is filled up, any new word coming into the storage will be placed by a random replacement structure. This facility is necessary because an exponential holding-time representation of past events could be made only by replacing old events randomly. During a congestion this helps to discount messages stored too far in the past, without making the probability of their representation zero. For purposes of simulation, it could be assumed that the stack length and the lengths of the messages are constants. It should be mentioned that depending upon individual requirements different storage structures could be assumed, e.g., the simple FIFO storage.

The SIMSCRIPT routine for illustrating the manner in which parallel occurrence of events as above could be represented is given below in two distinct steps.

TABLE IV. SIMSCRIPT Routine.

SIMSCRIPT ROUTINE

```
  EXOGENOUS EVENT START
  LET TO = TIME
5 LET T = TO + TINTR
  IF T GR TMAX, RETURN
  DO TO 1, FOR EACH SORCE I
  LET CT = TO
2 LET CT = CT + RAN (I)
  IF CT GR T, GO TO 1
  CREATE MFLAG
  STORE I IN SCODE (MFLAG)
  STORE CT IN TIM (MFLAG)
  FILE MFLAG IN SCHD
  GO TO 2
1 LOOP
3 IF SCHD IS EMPTY, GO TO 4
  REMOVE FIRST MFLAG FROM SCHD
  STORE SCODE (MFLAG) IN CODE
  CREATE MESGE
  STORE CODE IN SCOD (MESGE)
  STORE TIM (MFLAG) IN TME (MESGE)
  CREATE STORE
  STORE MESGE IN MESS (STORE)
  CAUSE STORE AT TME (MESGE)
  GO TO 3
4 LET TO = T
  GO TO 5
  END
```

DESCRIPTION

(1) RAN—A permanent system variable (one dimensional). The length of this array is N where N is the number of sources. The value of RAN is defined through random look up procedure. The initialization deck provides the random tables.

(2) SORCE—Permanent entity.

(3) MFLAG—Message flag-temporary entity, having attributes:
  a) Source code (SCODE)
  b) Time (TIM).

(4) SCHD—A ranked set having MFLAG's as members. The ranking is according to the value of TIM (MFLAG) (lowest ranking), i.e., the first member to be removed from the set is that having the smallest TIM.

(5) SCODE—Contains source identification.

(6) MESGE—Temporary entity having attributes:
  a) SCOD—Source code
  b) TME—Time of occurrence (Ranking attribute)
  c) SMEMO (SI)—Successor in set MEMO (SI)
  d) PMEMO (SI)—Predecessor in set MEMO (SI).

(7) TMAX—System variable-Contains upper limit for time of simulation.

(8) TINTR—System variable-Contains time interval.

STEP 1. *Exogenous Event START.* The messages sent by the sources (which are parallel) are scheduled and are stacked in the computer memory sourcewise. A fixed length message structure is assumed here. The simulation starts from the current time and goes upto a predermined time TMAX. The SIMSCRIPT routine and the corresponding description of entities, events and attributes are given in Table IV.

STEP 2. *Endogenous Event STORE.* It stacks the messages for each source as they are scheduled by the routine START. The SIMSCRIPT routine and the corresponding description of entities, events and attributes are given in Table V.

TABLE V. SIMSCRIPT Routine.

SIMSCRIPT ROUTINE

```
  ENDOGENOUS EVENT STORE
  STORE MESS (STORE) IN MESGE
  DESTROY STORE
  STORE SCOD (MESGE) IN SI
  LET CSTR (SI) = CSTR (SI) + 1
  IF (CSTR (SI)) GR (AVSTR (SI)), GOTO 1
  FILE MESGE IN MEMO (SI)
  RETURN
1 LET CSTR (SI) = AVSTR (SI)
  LET I = RANDI (1, AVSTR (SI))
  LET J = 0
  DO TO 2, FOR EACH MES OF MEMO (SI)
  STORE MES IN MESSG
  LET J = J + 1
  IF I EQ J, GO TO 3
2 LOOP
3 STORE MESSG IN MES
  LET TME (MESGE) = TME (MES)
  REMOVE MES FROM MEMO (SI)
  DESTROY MES
  FILE MESGE IN MEMO (SI)
  RETURN
  END
```

DESCRIPTION

(1) SI—Local variable containing the source index.
(2) AVSTR (SI)—Maximum available storage for source SI.
(3) CSTR (SI)—Current storage length for source SI.
(4) MEMO (SI)—Ranked set. Member entities are MESGE which have the attributes:
   a) Successor of the set
   b) Predecessor of the set
   c) Ranking attribute time, TME (lowest rank). And 2 owners First in set FMEMO. Last in set LMEMO.
(5) STORE—Event notice to stack the information coming from a particular source. This has 2 attributes:
   a) Time of occurrence
   b) MESS containing the identification number of the created MESGE.
(6) MESGE—Temporary entity as defined in exogenous routine START.
(7) RANDI (I, J)—Generates random numbers in the interval (I, J).

3. Priming Time

At the commencement of the simulation experiments the first schedule requires the values of $N_i'(t)$, $t_{\max, i}$, μ_i, etc., from the earth station. On the other hand, the processing of the commands at the earth station requires messages from the space station. In view of these conflicting requirements, it is assumed that the earth station sends a tentative "priming" message. Based on this, a schedule is prepared and the messages are transmitted according to this schedule. The REPORT GENERATOR is instructed to skip the status report of events during the "priming" time.

4. Scheduling Algorithm

Algorithms under test like those described in Section 3 should be built into the simulated system.

5. Simulation of Message Sequencing

The optimum values r_i and τ_i realized from the sequencing algorithm are taken as the basis for queueing the messages at the transmitter. The rate r_i (usually an integer) is expressed as the number of messages per block-time (t_b seconds) where the block time is a unit of time which is very small compared to the smallest value of $(t_{max,i} - t)$ but is large compared to the largest value of $T_{av}(\ell_i)$. During each distinct block-time, the sequencing requires the transmission of r_i messages from the i^{th} source for $i = 1, 2, \cdots, n$, in that order. Condition (2.5) can then be modified as

$$\sum_{i=1}^{n} [T_{av}(\ell_i)\, r_i] \leq t_b \qquad \text{(all the time variables expressed in seconds)} .$$

If $T_{av}(\ell_i)$ is normalized with respect to t_b and expressed in block time units, condition (2.5) will be unaltered. The rate r_i is expressed as number of messages per block-time. For the simulation we assume the latter relation expressing $T_{av}(\ell_i)$ in block time units.

That is

$$\sum_{i=1}^{n} T_{av}(\ell_i)\, r_i = M; \qquad M \leq 1 .$$

If $M = 1$, then, in one block time, r_i messages should be transmitted from source S_i $(i = 1, 2, \cdots, n)$. Under this condition the transmitter will never be idle. On the other hand, if $M \leq 1$, the transmitter remains idle for a fraction of a block-unit time during each block-unit. As this is undesirable, as soon as the messages from the last source S_n are transmitted, the next cycle of transmission from S_1 is begun.

6. Channelling of Messages

The output of the SIMSCRIPT routine corresponding to the above step forms the input to the transmitter at the earth. As the first word in the individual messages represents the code of the source, the entire message is sent to that computer storage location which is identified by this code.

7. Earth Station Computer Storage

As the storage design of the earth station computer is not as critical as that of the space station, to each source may be assigned a two-dimensional array of fixed dimensions.

8. Command Processing Unit

Commands can be processed by standardized algorithms which can be built into the simulated system.

9. Structure of the Earth to Space Message

Command messages are generated as soon as adequate number of messages are received from the space station. As these commands are interpreted at the space station, a general structure for the command is undesirable in order to minimize the hardware for the interpreting unit. All the data required for the next schedule period like $t'_{max,i}$, $N_{min,i}$, $N'_i(t)$, k_i, μ_i, and the next set of types of measurements on each source from part of the command along with control commands extraneous to the sources during the present schedule.

The general structure of the message which can accommodate a variable length is given below

NUMBER OF SUB-COMMANDS IN THIS MESSAGE	CODE OF SUB-COMMAND	MAGNITUDE ASSOCIATED WITH THE SUB-COMMAND	CODE OF SUB-COMMAND	MAGNITUDE ASSOCIATED WITH THE SUB-COMMAND	•••	CODE OF SUB-COMMAND	MAGNITUDE ASSOCIATED WITH THE SUB-COMMAND

Each syllable in the word forming the messages is of constant length.

10. Source Control Apparatus

One of the sub-commands from the earth will be the list of the types of measurements that should be carried out on each source during the next schedule period. This subcommand will actuate or control the apparatus for making the required measurements. Simulation of this unit is carried out by controlling the limits and distribution of the random numbers in the structure of the message given in Step 2, according to the requirements of the type of measurement.

The REPORT GENERATOR facility in SIMSCRIPT helps to print out status reports like the time phasing of various events going on in the system. Study of the alterations in the criteria of system design like the speed of operation of different units in the space-borne computer, the characteristics of the transmitter, the structure of the messages, the choice of the scheduling algorithm, etc., is permitted in a natural manner in SIMSCRIPT.

5. CONCLUSION

In the foregoing sections a coupling of transmitters and computers at the space and earth stations is described along with a discussion of the factors entering into the problem of their optimal coordination. A general scheduling algorithm is presented in the background of the requirements of simulation of the coupling using SIMSCRIPT. It is concluded that in space communication problems, like message scheduling that involve events described by statistical attributes, simulation offers a powerful design tool.

ACKNOWLEDGMENTS

This investigation is a part of a larger problem proposed by Dr. Vikram Sarabhai, (Chairman—Atomic Energy Commission, Chairman—Indian National Committee for Space Research, and Director—Space Science and Technology Centre). Valuable discussion with Prof. S. V. C. Aiya, Head of the Department of Electrical Communication Engineering, Indian Institute of Science, is acknowledged. Dr. K. V. Ramanathan, In-Charge of the Micro-Electronics Laboratory of the Tata Institute of Fundamental Research is thanked for acquainting us with the hardware problems associated with space-electronics. We are deeply indebted to Mr. K. K. Krishnan Kutty for very many illuminating suggestions on the use of SIMSCRIPT.

REFERENCES

[1] C. L. Weber, "Transmitter Optimization for Satellite Telemetry Systems," Meteorological & Communication Satellites, *XVI International Astronautical Congress, Athens* (New York: Gordon & Breach, 1965), 111–122.

[2] C. E. Anderson and F. G. Benf, "System Design of an Automated Laboratory and Associated Entry Vehicle for Unmanned Martian Exploration," Space-Craft Systems, *XVI International Astronautical Congress, Athens* (New York: Gordon & Breach, 1965), 147–201.

[3] M. J. Box, "A New Method for Constrained Optimization and a Comparison With Other Methods," *Computer Journal*, **8**, 45–52 (April, 1965).

[4] *3600/3800 Computer Systems SIMSCRIPT Reference Manual*, Control Data Corporation, Pub. No. 60134600 (Jan., 1966).