SIMULATION OF A MULTIPLE ELEMENT TEST ENVIRONMENT

Kenneth E. Dominiak
University of Florida
Eglin Air Force Base, Florida 32542

Ronald J. Ireland*
Honeywell Information Systems
Phoenix, Arizona 85029

Abstract

A real-time simulation of a multiple element defensive test environment is under study and development. This simulation is to be used for evaluating performance of command and control systems. The initial version of the simulation generates simultaneously up to twenty aircraft each with up to two jamming devices. The overall approach is to generate aircraft tracks and jamming returns at a large scale central digital computer. This data is then transmitted to remote radar sites and converted by on-site equipment to analog signals which are injected into the radar circuitry. A unique feature of the approach is that both real and simulated aircraft will be observed and tracked simultaneously by radar operators.

Introduction

A real-time simulation of a multiple element defensive test environment is presently under development. The rationale, design concept, and various features of the simulation are described in this paper with emphasis on the digital portion of the simulation.

This work was supported by the Range Development Division of the Air Force Armament Development and Test Center under Contract F08635-72-C-0060.

*Formerly with the University of Florida, Eglin Air Force Base, Florida.

Up to twenty aircraft, each with up to two electronic counter-measures (ECM) radar jamming devices, are generated in this initial version of the simulation. Additional numbers of aircraft will be introduced as required, after the basic simulation has been further developed. The overall approach is to generate aircraft tracks and jamming returns at a large scale central digital computer (CDC 6600). This data is then transmitted to remote radar sites and converted by on-site equipment to analog signals which are injected into the radar circuitry. A unique feature of the approach is that both real and simulated aircraft will be observed and tracked simultaneously by radar operators. The simulated aircraft will be indistinguishable from real aircraft on the radar displays. Radar operators will then interface with the remainder of the command and control system in a normal fashion.

A simplified block diagram is illustrated in Figure 1. Referring to Figure 1, the topics discussed in this paper include required inputs to the CPU, simulation activities of the CPU such as aircraft track and jamming return generation, transmission of data to the on-site buffers, and feedback information from the radars to the CPU. Not discussed here is the on-site D/A conversion equipment, the radar signal insertion equipment, and the interface between the radars and the command and control center. Both the D/A conversion and signal insertion equipments and associated concepts were developed by the Westinghouse Electric Corp., Baltimore, Md., under a separate contract.

The need for a simulation such as this arises as a result of several factors. A fundamental problem is that of validating a command and control system (CCS) by attempting to defeat it. Each such attempt requires the dedication of large numbers of aircraft, since experience has shown that the relationship between numbers of threat aircraft and utilized capacity of the CCS is highly nonlinear. That is, it is not possible to extrapolate results obtained with a few aircraft to the case of many aircraft (or saturation). As a result, it becomes highly desirable from both economic and operational considerations to generate some of the threat aircraft via simulation. On the other hand, the presence of real aircraft provides increased realism and improves the validity of the exercise. Another somewhat unrelated, but significant, advantage resulting from the use of some real aircraft is that other existing and planned simulations which utilize simulated aircraft only can be validated.

The current stage of development of the simulation can be summarized as follows:

a) Simulation requirements have been established. These are based on numbers of aircraft, maximum aircraft velocities and ranges, types of radars considered, and maximum desirable data rates between CPU and radar sites.

b) Much of the modeling is completed. In some areas, decisions concerning approaches to be taken have not yet been made, however, trade-off studies are currently underway.

c) On-site digital processing, D/A conversion, signal generation, and signal insertion equipment has been developed and demonstrated.

d) Interface requirements between the CPU and the on-site processing equipment have been established.

e) Input data requirements are only partially defined, and are currently under study.

f) Study of requirements for data

organization, storage, and retrieval is just beginning.

g) Specification of user inputs including format is not yet completed, however, it has been established that a Scenario Input Language will be developed to simplify test specification by users and to maximize control during an exercise.

h) Development of algorithms and coding is just underway. Initial coding will be in FORTRAN IV, however, assembly language will be employed if determined necessary.

Aircraft and Jammer Return Generation

Aircraft track information is provided to each radar in the form of radar cross-section (RCS) and aircraft position in range, azimuth, and elevation. This information must be supplied to the radars by the CPU at a rate sufficient to insure realistic target aircraft behavior on the radar screens, that is, behavior indistinguishable from that of a real aircraft. Three types of radars are initially being utilized - early warning (EW), track-while-scan (TWS), and conical scan (CS). The maximum number of aircraft tracks to be generated for each type of radar are listed in Table I. Aircraft dynamics for the worst case data transfer (i.e., highest data rate) can be bounded using the known maximum target velocity of Mach 3 (3000 ft./sec.) and maximum acceleration of 10g's (322 ft./sec.^{2}). These latter bounds are determined from a consideration of user requirements.

In establishing the basic simulation requirements, it was considered desirable to use existing telephone lines for transfer of data between the CPU and the radar sites. However, the maximum capacity of these lines is 2400 bits/sec. When a maximum data rate was calculated based on the numbers of targets required, the worst case target dynamics, and characteristics of the various radars, it was found to be far in excess of 2400 bits/sec. As a result, it was decided that some computation would be carried out on-site. By reducing the update rate on track parameters from the CPU and performing simple linear interpolation on-site, it was possible to reduce the maximum data rate to 2060 bits/sec.

TABLE I. RADAR TARGET REQUIREMENTS

RADAR TYPE	MAX. NO. OF TARGETS	MAX. TARGET RANGE (MI.)	MAX. JAMMERS PER AIRCRAFT
EW	20	150	2
TWS	8	60	2
CS	4	60	2

All on-site computation is performed using hardware rather than programmable software, and requires no operator to be present.

To illustrate the approach finally adopted, we consider the TWS radar. At intervals of 626 milliseconds, values of target aircraft RCS, range, azimuth, and elevation are made available at the radar site. The number of bits required for each track parameter of each target is shown in Table II. These are determined from accuracy considerations. The 96 bits for "other data" includes jamming information, timing, bookkeeping, and the like. If 8 target tracks are updated at the indicated intervals and with the indicated number of bits, a data rate of approximately 2060 bits/sec. results. However, if the TWS radar display is to appear realistic, the indicated update rate is too slow. Therefore, under the assumption of linear variation of track parameters within a 626 msec. interval, a constant increment is added to each track parameter every 62.6 msec. Thus, the update rate is increased by a factor of 10. The number of bits required for incrementing track parameters is also listed in Table II. Jammer information is provided to the radars in the form of gain factors. These are updated in the same fashion as track parameters.

TABLE II. CPU TO TWS RADAR DATA FLOW

TRACK PARAMETER	NO. OF BITS
RCS	8
RANGE (R)	14
AZIMUTH (θ)	12
ELEVATION (ϕ)	11
ΔRCS	5
ΔR	5
Δθ	5
Δϕ	5
OTHER DATA	96
TOTAL/UPDATE/TARGET	161

Various methods of generating aircraft tracks at the CPU have been examined. At maximum target ranges, it appears that point-to-point trajectories specified in terms of latitude, longitude, and altitude will provide sufficient accuracy and realism. However, at lesser ranges it appears that numerical integration of equations of motion will be required. Several integration methods which are particularly suited to the present application are being examined. Results obtained to date are summarized in the Appendix.

A variety of models and computational schemes either have been developed or are under development for specific target and jammer parameters. However, results are detailed and will not be presented here. Examples of these are the equations used to compute radar returns, models for jammer gains, and algorithms for determining which targets are in the field-of-view of which radars.

Inputs To The Central Processor

As discussed in the Introduction, input data requirements are only partially defined. Problems of data organization, storage, and retrieval will be addressed in the next phase of this continuing effort, as input data requirements become better known. In general, data inputs and file updates will be off-line since requirements for each exercise will be fixed.

Radar antenna patterns are stored at the radar sites rather than at the CPU. As a result, radar returns are computed at the CPU with the target always at mid-beam of the main lobe of the antenna pattern. The return is then modulated by the on-site processing equipment so as to properly position the target within the antenna pattern.

Three-dimensional jammer patterns are stored at the CPU. A requirement for adjacent level differences of .5 to 1.0 decibels in RF signal level has been established for jammer pattern storage. This requirement is based on the conclusion that level differences of this magnitude are essentially indiscernible when observing the resulting video on a radar display.

Each time the simulation is exercised, extensive user inputs in the form of a completely defined scenario will be required. It is important that the specification of a scenario be in terms of parameters and dimensions familiar to the user. It is impractical to expect the user to learn a computer language before he can exercise the simulation. For this reason, it has been determined that a Scenario Input Language will be developed. The language will make it possible for the user to specify aircraft positions, aircraft maneuvers, and the like, in terms familiar to him, thereby rendering the simulation more usable.

Interface With On-Site Equipment

The interface between the CPU (actually, the data link) and the on-site equipment has two facets - hardware and software. The software interface was discussed above for the TWS radar, where it was seen that a 161 bit word must be provided on-site for each target in view every

626 msec. The analysis supporting the selection of these values was also outlined. Similar analyses have been completed for the other radars. Results are summarized in Table III. The indicated word length in each case, if provided at the rate specified, will insure that all accuracy and resolution requirements are satisfied. Therefore, Table III completely defines the software interface between the CPU and the sites.

Feedback From Sites To CPU

In establishing simulation requirements, an important objective was to maximize computation performed by the CPU and minimize on-site processing. However, as discussed earlier, the conflicting objective of low data rates between CPU and sites led to a requirement for some on-site computation.

The feedback channel from the site to the CPU provides a means of reducing data rates and on-site computation. If it is known at the CPU how each radar antenna is oriented as a function of time, then targets in the field-of-view of each radar can be identified and only those targets need be transmitted to the respective site. Without feedback it becomes necessary to transmit all targets to each radar, and to then isolate targets within each field-of-view by means of on-site computation. The latter approach would involve higher data rates and require additional on-site computing capacity.

Shown in Table I is the maximum number of aircraft tracks generated for each type of radar.

TABLE III. CPU/RADAR SOFTWARE INTERFACE

RADAR TYPE	WORD LENGTH	UPDATE INTERVAL
TWS	161 bits	626 msec.
CS	171 bits	810 msec.
EW	176 bits	3 sec.

The hardware interface, illustrated in Figure 2, consists of an array of input registers at each site, one for each potential target (see Table I). The existing configuration provides one storage register per target, of word length as indicated in Table III. Thus, each target word can be stored on-site for a period of time corresponding to the update interval for that radar. This provides a tolerance on data computation and data transmission at the CPU. The existing tolerance can easily be increased by expanding the number of input registers, e.g., doubling the number of registers will double the tolerance.

If the number of aircraft within the field-of-view of the TWS or CS radars exceeds 8 or 4, respectively, then some of the targets must be eliminated from consideration. (It is assumed that the total number of targets generated never exceeds 20, so that a similar condition never arises in the case of the EW radar.) The CPU will eliminate targets as required using an appropriate algorithm, probably to be based on target range. This is an additional computation which must be performed on-site if the feedback channel is eliminated.

Summary

A digital simulation of a defensive test environment is under development. The simulation provides maneuvering target aircraft with ECM jamming for exercises designed to validate command and control systems. The simulation is an integral part of the CCS, which also includes radars, displays, operators, real aircraft with jammers, and a command and control center.

In this continuing program, simulation requirements have been established, modeling is completed or underway, significant interfaces have been defined, and on-site hardware has been developed and demonstrated.

Appendix

Four numerical integration techniques for generating aircraft tracks from equations of motion have been examined, each with a second and fourth-order integration method, and each method applied to two problems whose analytical solution is known. The four techniques selected are all basically predictor-corrector techniques. For the second order method the Nystrom midpoint formula is used as the predictor and the modified Euler formula as the corrector (Reference 1). For the fourth order method the Adams-Bashfourth formula is used as the predictor and the Adams-Moulton formula is used as the corrector (Reference 2). The four techniques employed are:

a) Predict, Correct c times (PC).

b) Predict, Correct c times, Modify (PCM).

c) Predict, Modify, Correct c times (PMC).

d) Predict, Modify, Correct c times, Modify (PMCM).

The first problem investigated was

$$\dot{y}_1 = \frac{1}{y_2} \qquad \dot{y}_2 = -\frac{1}{y_1}$$

$$\dot{y}_1(0) = 1 \qquad \dot{y}_2(0) = 1 \ .$$

The second problem was

$$\dot{y}_1 = y_2 \qquad y_1(0) = 1$$

$$\dot{y}_2 = -(9.25y_1+y_2) \qquad y_2(0) = -1/2 \ .$$

The results for the second-order method can be summarized as follows:

a) PCM is best with PC close second.

b) PMCM is, surprisingly, no better than PCM.

c) By making h, the integration step size, smaller we gain more in accuracy than any change in method produces.

d) Differences between methods are much more pronounced for larger values of h.

For the fourth-order method:

a) PC is best with PCM second.

b) Difference between techniques is not as pronounced as with second order methods.

c) PMCM does relatively better in fourth order than in second order.

d) Making h smaller does not have the impact that it does with the second order method.

In general:

a) Fourth order accuracy increases faster (relative to second order accuracy) with decreasing h.

b) Dividing h by 2 and using second order gives about same, or a slightly better, result than using fourth order with original h.

References

1. M.I. Skolnik, Introduction to Radar Systems. New York: McGraw-Hill, 1962.
2. J.V.Di Franco and W.L. Rubin, Radar Detection. Prentice-Hall: Englewood Cliffs, N.J., 1968.
3. R.W. Hamming, Numerical Methods for Scientists and Engineers. New York: McGraw-Hill, 1962.
4. L. Lapidus and J. Seinfeld, Numerical Solution of Ordinary Differential Equations. New York: Academic Press, 1971.
5. A. Ralston, A First Course in Numerical Analysis. New York: McGraw-Hill, 1965.
6. Study and Design of Signal Injection Systems Range Instrumentation Design Plan, Report No. 86-52313, Westinhgouse Electric Corp., Baltimore, Md., November 19, 1971.
7. Study and Design of Signal Injection Systems, Contract Report: F-8635-72-C-0008, Westinghouse Electric Corp., Baltimore, Md., March 1, 1972.

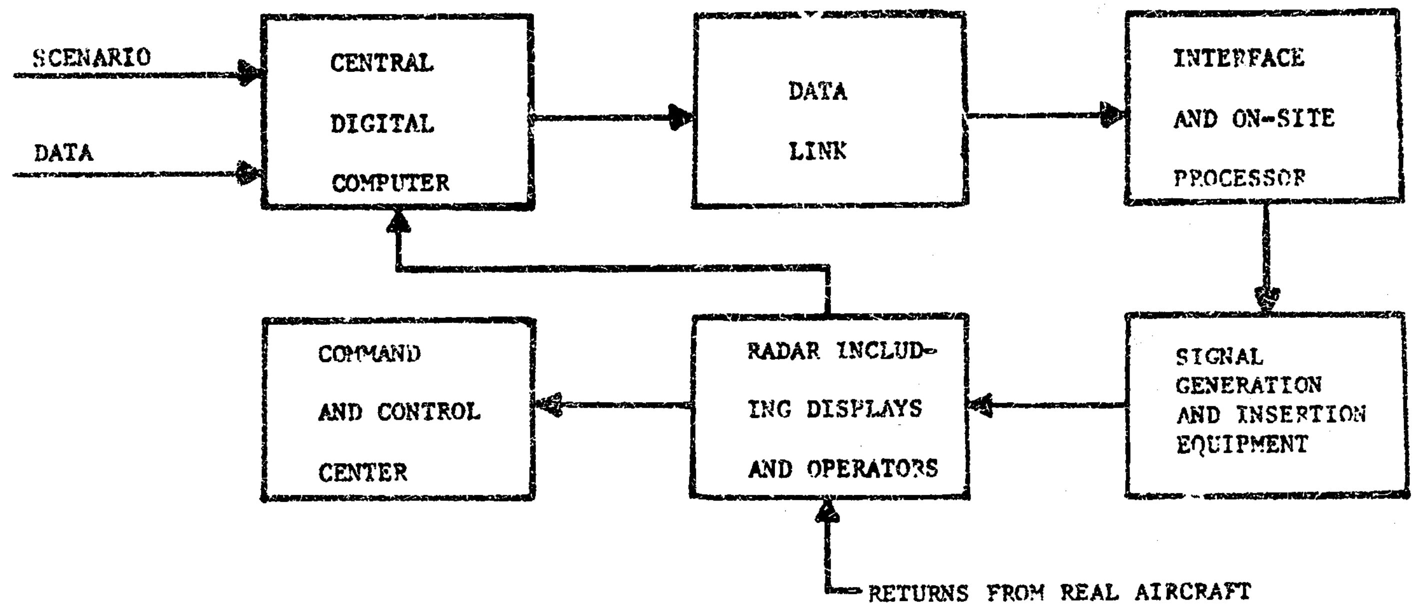

FIGURE 1. SIMPLIFIED BLOCK DIAGRAM

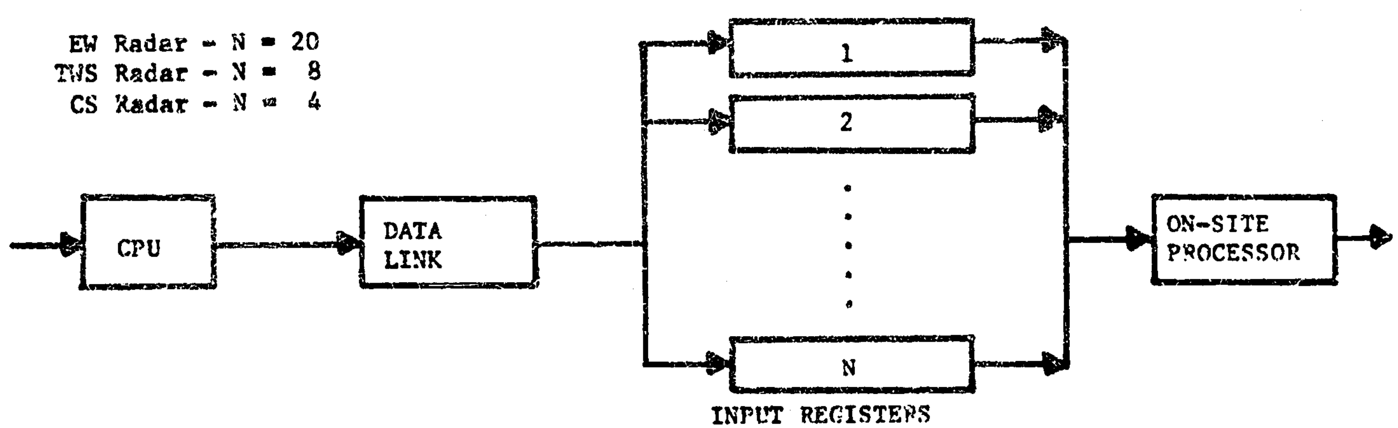

FIGURE 2. CPU/SITE HARDWARE INTERFACE