Calhoun: The NPS Institutional Archive
DSpace Repository

NPS Scholarship | Publications

1978

# STAR: Simulation of Tactical Alternative Responses

Parry, Sam H.

https://hdl.handle.net/10945/38594

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

*Downloaded from NPS Archive: Calhoun*

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

# STAR: SIMULATION OF TACTICAL ALTERNATIVE RESPONSES

Samuel H.Parry

## ABSTRACT

This paper describes the development and initial results of a mid-resolution combined arms model to determine both hardware and training measures of effectiveness. One of the primary goals of the model is to achieve an acceptable level of resolution while assuring that the model inputs and interactions are understood by the military decision maker.

The structure of the model is truly hierarchical in that it is not confined to any specific unit size configuration. The parent-child set structure of SIMSCRIPT, coupled with the flexible parametric terrain model, provides the required capabilities to realize a hierarchical representation. The level of resolution is prescribed by the combat function being represented and not by inherent model structure limitations. For example, if the study is to evaluate the contribution of mobility/agility to survivability, resolution to the element level is obviously required, possibly in a battalion-level context. On the other hand, if the study is to evaluate the impact of digital communications between higher headquarters, it may suffice to play a company or battalion as the basic element on a much larger battlefield. The results of battalion-level engagements in the larger context could be evaluated by use of the lower hierarchical module.

## INTRODUCTION

A significant effort is currently under way at the Naval Postgraduate School to develop a combined arms simulation model to determine both hardware and training measures of effectiveness. One of the primary goals of STAR (Simulation of Tactical Alternative Responses) is to achieve an acceptable level of resolution while assuring that the model inputs, interactions and outputs are understood by the military decision-maker. At the present time three faculty members and nineteen U.S. Army students are involved in various facets of the modeling effort.

The structure of STAR is truly hierarchical in that it is not confined to any specific unit size configuration. The parent-child set structure of SIMSCRIPT, coupled with the flexible parametric terrain model, provides the required capabilities to realize a hierarchical representation. The The level of resolution is prescribed by the requirements. For example, if the study is to evaluate the contribution of mobility/agility to survivability, resolution to the element level is obviously required, possibly in a battalion-level context. On the other hand, if the study is to evaluate the impact of digital communications between higher headquarters, it may suffice to play a company or battalion as the basic element on a much larger battlefield. The results of battalion-level engagements in the larger context could be evaluated by use of the lower hierarchical module.

The first study application of STAR will be in support of the 105/120 mm ammunition stowed load requirements for the XM-1 tank. Initial production runs for the study will be conducted for a blue battalion vs a red regiment in December, 1978. This version of STAR will represent all appropriate ground direct fire units, two-sided artillery, minefields and smoke.

Upon completion of the Phase I battalion-level production runs (on a 10x10 km battlefield), Phase II will be initiated. The result of Phase II (April 1979) will be a brigade-level model capable of representing a blue brigade versus a red division on a battlefield approximately 50x50 km. The Phase II model will be capable of simulating a multi-echelon red regimental attack on multiple avenues of approach in both the Covering Force Area (CFA) and the Main Battle Area (MBA). Extended dynamic play of ammunition and POL resupply, as well as a significant enhancement of the tactical representation of battalion engagements, will result from the Phase II model. In addition, artillery units will be directly represented on the battlefield, allowing for specific play of counterbattery and counter air-defense fires. Finally, a dynamic air-air defense module is being developed for the Phase II model. This module will be capable of representing two-sided air-to-air engagements for both fixed wing aircraft and helicopers. All appropriate red and blue air defense systems will be played in the model.

The remainder of this paper focuses on the Phase II model, beginning with a brief description of SIMSCRIPT and the motivation for its selection for STAR.

## SIMSCRIPT

The SIMSCRIPT language, level II.5, was selected for STAR because the language was designed to model discrete event simulations. Because of the many imbedded features of the language, the programmer is given wide latitude in the construction of the event flow. The language is also very much like English with regard to the construction of commands. This feature, coupled with a very free form of syntax, allows the code to be written in an eminently readable manner. An example taken from a STAR subroutine will be shown later and will illustrate this point.

The heart of the imbedded simulation facilities is the timer, which is used with certain structural characteristics: entities, attributes, sets, and events. These facilities greatly simplify the process of writing a simulation program and debugging the code. This is further enhanced by a compiler which provides error messages and traceback routines similar to WATFOR and WATFIV in FORTRAN.

The structure of STAR begins with the concept of an entity. An entity is simply a representation or model of an item. In other simulations an entity might represent people waiting in line for tickets, customers in a bank, or cars waiting to be serviced in a gas station. In STAR the basic entity is a weapon system representing tanks, TOW's, artillery pieces, etc. Any of these entities may be brought into existence by a simple phrase which includes the name of the entity. For example, the phrase CREATE A TANK reserves a place in core for the entity which the programmer has chosen to call TANK. Associated with the word TANK is a pointer variable which points to the location in core where TANK is stored.

It is desirable to associate certain characteristics with entities after they have been created. These characteristics are referred to as attributes and are affixed to an entity by the internal bookkeeping procedures of SIMSCRIPT (the system) or are placed on the entity by the programmer. The programmer may wish to give the TANK a name or assign a current location. Phrases such as LET NAME(TANK) = XM1 and LET X.CURRENT(TANK) = 3150 will accomplish these tasks. Attributes may be changed by the programmer as necessary to reflect changes in characteristics. Moreover, the system will change system-defined attributes as necessary.

The concept of sets in SIMSCRIPT is very useful when it is necessary to group entities based on certain common characteristics or in the construction of queues. In STAR sets have been used primarily to portray membership in organizations. The set structure mirrors organizational structure and enhances the programmer's ability to model unit tactics from a micro to macro level. An entity may belong to any number of sets and the entity acquires a membership attribute which facilitates identification of an entity's unit. For example, we might wish to indicate that a TANK belongs to the red force's first regiment. The phrase FILE TANK IN RED.1ST.RGT will accomplish this task by setting a membership attribute, M.RED.1ST.RGT(TANK), equal to an interger 1. In addition, the entity will have attributes which indicate which entities precede and succeed him in the set. The set will have attributes which indicate who the first and last members of the set are and how many members belong to the set. An extremely flexible method of filing allows entities to be ordered in a set by ranking of certain attributes or by a simple first-in first-out basis. STAR uses the latter system for most applications. Currently, blue force elements are assigned to sections, platoons, companies, and battalions. Red force elements are assigned to platoons, companies, and battalions, and regiments. The set logic of SIMSCRIPT allows this to be easily expanded to higher level organizations.

Each entity in STAR is modeled to reflect a flow of activities over time. In particular, each entity initiates, or undergoes, search, detection, target selection, firing or impact. These five events are scheduled dynamically based on the current tactical situation or by way of an appropriate probability distribution. The nature of events dictate the appropriate probability distribution. When an event is scheduled for an entity, the SIMSCRIPT timer makes a record of the time that the event is to occur (in terms of overall simulated time) and the entity for which the event has been scheduled. Other characteristics of the event may be recorded in a manner similar to the assignment of attributes. At the appropriate simulated time, the event is executed unless cancelled by some logic provided by the programmer. For example, the scheduling of a fire event by a TANK on a BMP could be accomplished by the phrase: SCHEDULE A FIRE GIVEN TANK AND BMP IN X SECONDS. This event would be filed in an event set which contained, among other things, the time that the event is to take place, the entities involved in the event, and the event location with respect to other scheduled events. When X seconds had elapsed from the current simulated time, the event would take place and the consequences of the logic written in the event routine would be executed. Event routines may in turn generate other events. FIRE, for example, causes the scheduling of an IMPACT event. IMPACT leads to the scheduling of other DETECT and TARGET.SELECT events.

Event routines are supported by a number of computational subroutines in STAR. Subroutines are written in both SIMSCRIPT and FORTRAN which has given the simulation a great deal of flexibility. The difficult line-of-sight calculations, for example, are accomplished in FORTRAN because the terrain model was originally written in FORTRAN. It was this capability to call FORTRAN subroutines that made SIMSCRIPT an even more appealing language. Existing FORTRAN routines could be used with only minor modifications. Other routines which were more closely tied to the entity structure of SIMSCRIPT were written in that language. The routine that updates the list of detected targets for each TANK is written in SIMSCRIPT to take advantage of the dynamic dimensioning capabilities of the language and the pointer variable link listing techniques available. For large target arrays, these language features are extremely efficient in reducing core requirements.

The following is an example of an event routine in STAR. This particular event initiates search and is scheduled to occur every 30 seconds for all TANK entities in the simulation unless otherwise aborted. A listing is presented followed by comments keyed to the line numbers.

SEARCH ROUTINE

```
 1  UPON STEP.TIME(A)
 2  DEFINE J, K, AND A AS INTEGER VARIABLES
 3  DEFINE LOSE AS AN INTEGER VARIABLE
 4  FOR EACH TANK IN TANKS WITH COLOR(TANK) =
    RED, DO
 5  IF X.CURRENT(TANK) LE 1500.0 SCHEDULE A
    STOP.SIMULATION NOW
 6  ALWAYS
 7  LOOP
 8  IF ALIVE.DEAD(A) EQ 1 RETURN
 9  ELSE "GO THROUGH ENTIRE LOOP"
10  IF FIP(A) EQ 1 AND STEPS(A) LE 2
11  SCHEDULE A STEP.TIME(A) IN 30 UNITS
12  LET STEPS(A) = STEPS(A) + 1
13  RETURN
14  ELSE
15  LET J = COLOR(A)
16  LET K = ABS.F(J-1)
17  LET STEPS(A) = STEPS(A) - 1
18  FOR EACH TANK IN TANKS WITH COLOR(TANK)= K, DO
19  IF ALIVE.DEAD(TANK) EQ 1 GO TO LOOP
20  ELSE CALL LOOK(A,TANK) YIELDING LOSE
21  IF LOSE EQ 1
22  CALL CARDIO(A,TANK) YIELDING DET.TIME
23  SCHEDULE A DETECT(A,TANK) IN DET.TIME UNITS
24  ALWAYS
25  'LOOP' LOOP
26  IF STEPS(A) GE 2 RETURN
27  ELSE
28  SCHEDULE A STEP.TIME(A) IN 30 UNITS
29  LET STEPS(A) = STEPS(A) + 1
30  RETURN
31  END
```

Line 1: The event has one explicit argument. In this case it is a pointer variable for a TANK entity which is stored in the variable A. At the appropriate simulated time the event will be executed for the entity A.
Lines 2 and 3: Explicit type declaration statements similar to FORTRAN are used to define the mode of variables.
Line 4: This line initiates a loop which searches through a set called TANKS causing the consequence of the loop to apply only to those TANK's with a RED color attribute.
Lines 5, 6, and 7: The consequence of the loop instructs the program to stop the simulation if any red force tank reaches an x-coordinate of 1500 or less.
Lines 8 and 9: If the entity (A) for which this event was scheduled is dead, control is returned to the event timer. Otherwise, the event routine continues.
Lines 10, 11, 12, 13 and 14: If the entity (A) currently has a firing event in progress and has fewer than three STEP.TIME events scheduled, another STEP.TIME event is scheduled for the entity (A) and control is returned to the event timer.
Lines 15 and 16: The color attribute of each TANK, indicated by the words RED or BLUE, is actually stored as an integer 1 or 0. These lines set the variable K to the opposite color of the entity for which the event is being executed.
Line 17: This causes a decrement in the STEP.TIME event schedule counter.
Line 18: This initiates a loop similar to Line 4.
Line 19: If the particular TANK in the set TANKS which is currently being accessed is dead, skip to the end of the loop.
Line 20: If the opposing TANK is alive, call the subroutine LOOK which returns the argument LOSE (line-of-sight exists) as an integer 1 or 0.
Lines 21, 22, and 23: If line-of-sight exists, call the subroutine CARDIO which calculates a time to detect for tank A on the TANK currently being accessed in the loop. A DETECT event is then scheduled for tank A on TANK in DET.TIME seconds from the current simulated time.
Lines 24 and 25: Regardless of whether line-of-sight exists, continue through the loop and check the next TANK in TANKS.
Lines 26, 27, 28, 29, 30, and 31: If two or more STEP.TIME events are scheduled for tank A, return control to the timer. Otherwise, schedule another STEP.TIME event for tank A in 30 seconds and increment the STEP.TIME event schedule counter. Control is then returned to the timer.

## Modular Functions for the STAR Model

### Terrain

A continuous macro-terrain representation is provided through the use of bivariate normal distribution functions. The benefits of this representation over classical digitized terrain are as follows:

1. Line-of-sight computations are made directly from mathematical relationships as opposed to the time-consuming iterative process required with digital terrain.

2. Mobility is truly continuous as opposed to piecewise linear techniques used for digitized terrain models.

3. Terrain can be considered as a parameter of the combined arms analysis as opposed to a given. For example, what are the specific terrain characteristics which favor a light, agile vehicle over a slower, heavier vehicle?

4. By appropriate selection of input parameters, any real section of terrain can be closely approximated by the parametric terrain model.

5. Any size terrain sector can be easily generated which forms the basis for the hierarchical model structure described in a subsequent section.

6. A dynamic smoke module has been developed and operated in the parametric terrain model.

## Target Acquisition

Acquisition of ground direct fire units is based on a time-to-detect computation. For visual detection, a detection time probability model is used based on the assumed negative exponentially distributed detection times. The detection rate function, $\lambda(t)$, is related to variables which describe the "physical conditions" for each target-observer combination during small time intervals in the simulation.

Each element in a unit may be given search responsibility for a specified sector of the terrain. The size of the search sector assigned to each element will depend on the anticipated threat direction and the unit's formation. In some cases we might even be concerned with search by an individual over a full 360 degrees. In the absence of a stimulus (such as target firing, intelligence communications, or unit tactics) the observer employs a Cardioid probability density function search procedure oriented along the observer's principal observation direction. The Cardioid distribution is divided into $k$ search sectors. An associated probability of searching in the $k^{th}$ sector, $P_k$, during a time interval (0,t) is determined. Thus, the probability of detecting a target located within the $k^{th}$ interval during (0, t) is given by

$$p^k(t) = 1 - \exp(- P_k \lambda t).$$

The determination of time-to-detect for various sensor systems (such as the tank thermal sight) involves a different computation for $\lambda$ than for the visual detection (especially in a battlefield smoke environment). Investigations are currently underway to determine appropriate computational procedures for $\lambda$ for these cases.

## Communications

A Battalion Communication Model is currently under development which will incorporate EW considerations. The model contains the following essential elements:

Realistic propagation model.

Incomplete transmission detection and retransmission.

Frequency change effects.

Play of type message traffic.

Play of red intercept, MXTR location, jamming.

Play of dead space.

The methodology of the model has the following features:

Propagation model based on Longley Rice equations.

Thresholds per radio type (I/S).

Assessment of time penalties for retransmissions.

Generation of tactical traffic over nets.

Thresholds per radio type (J/S).

Tactical placement of red EW equipment.

Full implementation of the model depends on voice message traffic time data and a specification of Soviet doctrinal EW operations and performance standards.

## Target Selection

A significant research effort is underway to develop target selection criteria based on firer-target systems, remaining ammunition by type, firer current activity, anticipated resupply, target range, aspect, speed, cover/concealment, firing activity, and sector location. This effort involves the presentation of a large number of situations to a cross-section of Army personnel. Each individual will specify a criticality index and a fire/no-fire decision for each situation. Correlation/ regression analysis will then be used to generate target selection criteria. In all cases an override criteria dictated by unit leaders is incorporated into the target selection module.

## Movement Routes

Movement routes for each element may be represented either by equations or by a series of (x,y) coordinates. Routes will be described by a series of (x, y) coordinates only when the use of equations is cumbersome or not feasible (such as individual element movement within a defensive position). The use of equations will allow truly continuous movement along non-linear routes such as serpentines. The coefficients of the parametric equations may be determined based on the tactical situation and terrain conditions.

Selection of alternative routes will be accomplished within the simulation to the greatest extent possible. Under certain conditions, however, it may be necessary to implement a "stop battle criteria" to allow a manual input of the route. For example, it may not be possible to specify the total criteria for a counterattack decision. In this case criteria for a "stop battle" event would be specified (such as the enemy has closed to within 1200 meters of the friendly force). At this point, the battle would be stopped and the current situation displayed to a decision-maker, who would then decide on the appropriate course of action and modify the data base accordingly.

## Ground Mobility

Movement of individual elements along their designated route is computed over the period of time required by the battle situation. The distance moved is determined as a function of current speed and acceleration and acceleration/deceleration capabilities dictated by terrain and unit movement tactics.

An important feature of STAR is that the element position and motion parameters do not have to be interpolated or extrapolated to satisfy a required computation. For example, many models are based on event times of fixed duration. It may well happen that a target element for a given firing event has been moved ahead in time beyond the time at which a round impact would occur. In this case the element's position and movement parameters must either be interpolated or merely use the current values. This situation can lead to some erroneous conclusions, especially for highly mobile vehicles whose posture relative to the firer can alter significantly in a short period of time.

Direct Fire

The direct fire modules utilize bias, dispersion and lethality data available from appropriate Army agencies and field tests. Because STAR determines precisely the target's location, aspect and movement parameters at both trigger pull and impact, a high resolution determination of each shot is made within the capability of the data base to support the determination.

The full range of kill types is represented, as well as the determination of kill type as a function of target cover. The model also has the capability to represent various lethality arrays as a function of the remaining rounds configuration in the target vehicle. This feature is parricularly critical to the 105/120 mm Stowed Load Study for the XM-1.

Artillery

The artillery module for the STAR Brigade Level Model gives a high resolution representation of direct support fires, counter fires and tactical fire direction. The module plays both Red and Blue Artillery. The Blue Artillery consists of a direct support 155 mm battalion, reinforcing artillery and general support artillery in support of the division's counter fire mission. Field Artillery batteries and forward observers are represented within the simulation "terrain box" and are vulnerable to enemy fires. Movement of artillery from primary to alternate positions is played. Red Artillery is represented in a similar fashion, altered as necessary to conform to hardware, organizational and doctrinal differences between U. S. and Soviet forces. Discussion in this paper is limited to Blue Artillery only.

Two major submodules extend artillery contol across the brigade area. Each battalion fire direction center submodule exercises tactical fire direction to its three batteries. The logic includes decisions concerning the massing of fires, number of volleys to fire, moving units and allocating firing units. The Division Artillery Tactical Operations Center Submodule exercises control over counter fires in the brigade area and determines what counter fires are to be conducted against both enemy artillery and enemy air defense.

The basic Field Artillery System is the battery. This building block of the module consists of three forward observers, a fire direction center, a firing battery and appropriate communications nets. At this level target acquisition, target engagement and damage assessment are represented by activities in the simulation.

Forward observers acquire targets and are located in vehicles in the vicinity of their respective maneuver company commanders. Each observer is assigned a sector of responsibility which he scans for targets. Given line-of-sight, forward observers detect enemy vehicles within visual range according to a given probability distribution, and a priority is then assigned each target or target array. The factors considered in determining this priority are the target's distance from the nearest friendly element, the number of elements in the target array and the hostile status (firing/not firing).

The forward observer requests fires on his highest priority target, given that the target is within range of the firing battery. If the target is stationary, the forward observer sends the fire direction center the target's current location. If the target is moving, the forward observer estimates where the target will be at the time that the rounds will land and sends that estimated future location to the fire direction center. Each forward observer has access to one communications net (a radio link with the fire direction center) and tries to transmit the fire mission over that net. Only one forward observer may use the communications net at any one time. Firing is done by adjustment or by fire-for-effect as required. Once a fire mission is completed, the fire direction center takes the highest priority target from a fire mission queue. This fire mission queue consists of fire missions that have been transmitted over the communications net, have not been acted upon and are still current. The times and acccuracies for target detection, communication, fire direction and firing battery procedures are modeled with probability distributions. These distributions have been derived to reflect 95% confidence in satisfying the appropriate Army Training Evaluation Program (ARTEP) criteria for a category 1 unit.

Once the firing battery fires a volley during the fire-for-effect phase of the mission, the damage assessment routine is called into play. This routine gives the damage inflicted by each round in the volley on each target. In route to this damage level, the routine models the accuracy of the target acquisition device, the mean point of impact of the volley, and the individual impact location of each round in the volley. Each round impact point is distributed bivariate normal with parameters based on the weapon type, ammunition type and range to target. All targets within the forward observer's area of responsibility are examined to determine if any lie within certain radii of the round's impact point. These radii are referred to as lethality rings. If a target lies within a certain radius, it is assessed damage at the appropriate level.

Depending on the effect level determined above, appropriate attributes of the target will be changed (i.e., if a mobility kill is assessed on an enemy tank, the tank will not be allowed to move for the rest of the battle). This routine is repeated for each round in the volley.

Air/Air Defense

The Air/Air Defense module will provide a model in the combined arms environment to simulate both fixed wing aircraft, helicopters and air defense weapon systems. These systems will be in support of combat operations in the Covering Force Area and Main Battle Area for both Red and Blue forces.

Air Weapons Systems played will concentrate primarily on anti-armor weapons such as Anti-tank Guided Missiles (ATGM's). Both the COBRA gun ship with TOW missile and the Advanced Attach Helicopter with HELLFIRE will be represented. The Soviet equivalent, the HIND with its ATGM and any postulated follow-on helicopter will be specifically modeled. The AIO with ATGM and Rockets, as well as the Soviet tactical aircraft equivalent, will be the primary fixed wing aircraft played.

Air Defense Weapon Systems will include both guns and surface-to-air missiles (SAM). Gun systems played will include the U. S. Vulcan and DIVAD Gun and the Soviet ZSU-23/4, 57/2. Missile Systems will include both Radar directed and Man Portable Air Defense Systems (MANPAD's). U. S. Missile Systems include Redeye (Stinger), Chaparral (Roland) and, to the extent that it is necessary, Improved Hawk (Patriot). The latter systems will be played in direct support of the committed brigade. The Soviet Systems will include the SA-7, SA-8, SA-9 and their postulated replacements.

Target acquisition and selection for attacking aircraft will be similar to the ground-to-ground functions. Time-to-detect will be computed as a function of relative speed, range, target geometry and physical contrast. For ground-to-air systems, target acquisition will be by either visual or electronic means, with target identification represented as a delay in the time-to-detect. Command and control will be simulated through the use of sector defenses within an integrated defense.

Aircraft movement will be accomplished by preplanned alternative routes throughout the battle area to designated battle locations. Committment of air assets may then be preplanned strikes or dynamic committment as reserves using the preplanned routes.

Air Defense Movement will be dictated by the movement scheme of the supported units. Tactical positions will be preplanned for each supported unit's tactical position.

Aircraft movement will be by fire-distance equations. Both level flight and terrain-following flight will be possible for fixed-wing aircraft. Helicopters will use Nap of the Earth (NOE), level flight or hovering maneuvers.

Both the Air-to Ground and Ground-to-Air Firing Modules will incorporate a high resolution representation of the time relationships insuring that only "valid" engagements are evaluated. Outcomes of engagements will be evaluated using probabilities of hit and probabilities of kill given a hit.

## TRAINING

Currently existing combat models were developed primarily to evaluate hardware alternatives. For this reason no satisfactory model exists which directly represents training measures of performance (MOP) as they interact with the hardware to determine training measures of effectiveness (MOE). To give a grossly oversimplified example, given a million dollars to spend for training, is it more effective to train tank crews to level A, artillery crews to level B, or attack helicopter crews to level C? This evaluation cannot be made at the training MOP level, but rather must be made in a combined arms model capable of determining appropriate MOE values.

It is hypothesized that two basic parameters characterize the performance of combat personnel: the time to perform a stated mission and the "accuracy" of mission performance. An outline of the methodology which will be used to determine training MOE's is given below. The basic concept throughout is to model both the personnel and hardware components of each battlefield system to provide a common basis for MOE evaluation.

1. Define the missions to be performed by each battlefield system, using ARTEPS, "How to Fight" books, Field Manuals, etc., as the basic source documents.

2. Partition the missions into tasks such that the performance of each task can be measured for both personnel and hardware contributions. (This step requires a careful analysis of the feasibility of collecting quantitative data for each task.)

3. Develop a total error budget model for each system. This step involves a specification of every component which contributes to system accuracy and identification of the error source (e.g., personnel, hardware, or a combination of the two).

4. Develop functional relationships between components of the error budget and the time to perform the mission task. This step is very complex in that the functional relationships must consider environmental, threat and force structure factors, as well as those of the system itself.

5. Incorporate the results of steps 3. and 4. in STAR. Note that the system MOP's developed above enable each battlefield system to be described in terms of both personnel and hardware performance.

6. Determine appropriate MOE's through execution of the model to evaluate the contribution of the

component MOP's (both personnel and hardware).

7. Develop models to determine the training and hardware development requirements and costs to attain the various levels of performance evaluated in step 6. above.

8. Develop appropriate Cost/Effectiveness Ratios (CER) to determine the optimal utilization of training and hardware procurement resources.

## SUMMARY

The model development effort described in this paper provides a new approach for evaluation of the many factors in the combined arms environment. The event and set structure of SIMSCRIPT, coupled with the computational capabilities of FORTRAN, provides a powerful tool for simulating a very complex environment. It is anticipated that the model will be used in several Army studies in 1979, in addition to the XM-1 Stowed Load Study described in this paper.