SEMI-AUTOMATED FORCE SIMULATION USING A BLACKBOARD

Mr. Kermit Gates
Mr. Frederick Frantz
PAR Government Systems Corporation
New Hartford, NY

ABSTRACT

Simulation of opposing and ancillary forces is a critical issue in training environments such as SIMNET and the Combat Training Centers. This paper addresses a general architecture for developing semi-automated force simulations, based on expert system and blackboard technology.

INTRODUCTION

Simulation of opposing and ancillary forces is a critical issue in training environments such as SIMNET and the Combat Training Centers. Advances in expert systems and artificial intelligence can be applied to developing highly representative opposing and ancillary force activities without a large complement of human players to control the forces. This paper addresses a general architecture for developing semi-automated force simulations, based on expert system and blackboard technology.

A blackboard architecture [1,2,3] provides a mechanism to utilize a number of knowledge sources (KSs) to determine appropriate responses to the simulated battlefield situation. KSs can be both rule bases, which can be developed to encode doctrinal information; and algorithms, which provide basic interpretation and processing of data representing the simulated battlefield environment.

The flexibility of a backboard approach yields a number of advantages:

1. Hierarchical adversaries can be constructed, with manual control at the highest level desired for a particular training exercise.
2. Greater sophistication in various levels of adversary skill can be implemented.
3. New algorithms, rules, databases, and KSs can be easily added.

The blackboard contains the information required to control the simulated unit. For example, the blackboard structure for a tank company contains information on:

- Environment data
- Where each weapon system is located
- Status of each weapon system
- What each weapon system is doing now
- What each weapon system intends to do in the future

KSs are then constructed for various combat states (administrative march, deliberate attack, etc.) as well as more basic functions (distance measurement).

This paper describes the components of a demonstration of semi-automated force simulation of a tank company implemented on a Sun workstation. Issues to be addressed include the blackboard architecture, the specific data maintained in the blackboard, the KSs used to control simulated force activity, use of map data, and techniques for building higher level activities of battalions and regiments on top of the company simulation.

BASIC BLACKBOARD ARCHITECTURE

We begin the technical discussion with a description of the basic blackboard structure that is used for supporting the simulation. The blackboard architecture described below is the most flexible means to provide the semi-automated force simulation capabilities. The blackboard is not monolithic. It allows flexibility by accepting a mix of heuristic and algorithmic processes necessary for complex simulations. Its hierarchical structure provides organization for large numbers of units with highly complex interactions.

Coupled with the blackboard architecture are object-oriented programming (OOP) techniques. In using OOP, economies are gained in the inheritance of characteristics passed from parent to child (brigade to battalion, for example). Also, each unit (object instance) contains its own characteristics and capabilities so that simulations written with OOP techniques are scenario independent.

Semi-Automated Force Simulation Structure

The blackboard system is modular in design and hierarchical in structure. It is the most likely format for a semi-automated gaming system. Figure 1 provides the operational concept for the semi-automated force simulation. The major analysis functions (Control, Blue Ops/Intel, Red Ops/Intel) are represented by blackboard architectures shown in Figure 2 [4]. The information needed to answer the questions in each box in Figure 1 is contained in a slot on the blackboard.

The knowledge sources ($KS_{1,2,\ldots,n}$) are independent and analyze information within the blackboard. The KSs may fuse data at one level to fill a requirement at another level. Another KS may ensure the elimination of redundant data. Other KSs may act only on information contained on the blackboard or may extract information required from algorithms ($ALG_{1,2,3}$) or knowledge-based systems ($KBS_{1,2,3}$). The

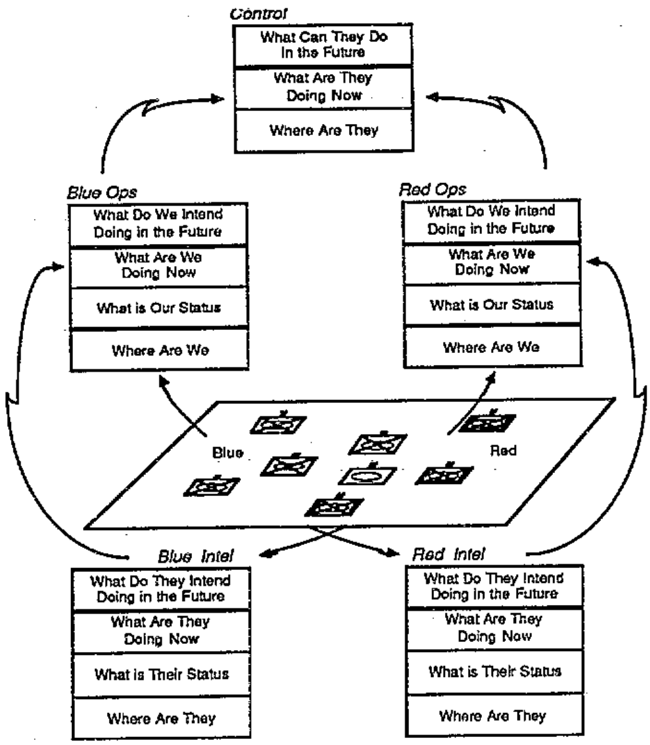

Figure 1, Semi-Automated Force Simulation Information Flow

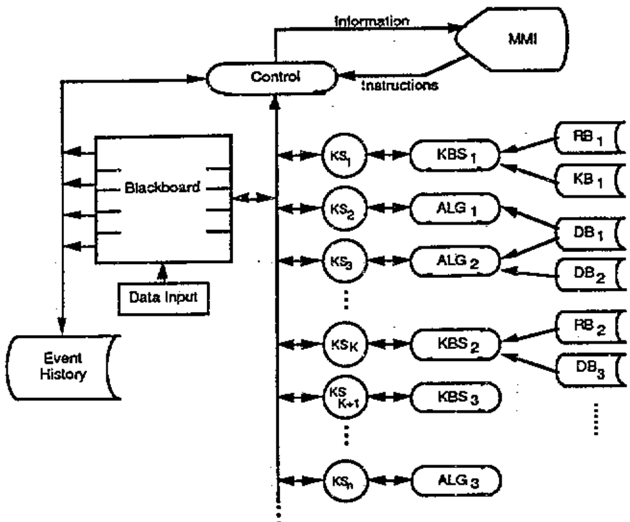

Figure 2, A Generalized Blackboard Architecture

algorithms and knowledge-based systems in turn access data or information from rule bases ($RB_{1,2}$), knowledge bases (KB_1) or databases ($DB_{1,2,3}$).

The blackboard maintains the current state of affairs. Therefore, the Event History file maintains the history of the simulation for later playback and review. The modularity of the system provides a means to replace individual elements (KS, KBS, ALG, RB, KB, or DB) without requiring revision of other system elements.

The exercise staff reviews the progress of the simulation from blackboard data via monitors (MMI) and can issue commands (Red and Blue sides) based upon incoming information.

An example using the Red Ops Blackboard illustrates the power of the system. As shown in Figure 1, the Red Ops Blackboard must address the following:

- Where are we,
- What is our status,
- What are we doing now, and
- What do we intend doing in the future?

These four questions translate into the following objective functions for each slot in the blackboard:

- Current Red unit locations are recorded at the logical level of resolution (e.g., for a platoon, individual vehicle locations would be shown; for battalion level, company centroids, with optional platoon centroids, would be shown),
- The status slot provides information concerning losses, ammunition stockages, fuel availability, etc.,
- The current state-of-affairs furnishes information concerning what the subordinates are doing as compared to the current mission, and
- All of the above information in addition to the Red perception of Blue provides options to future missions.

The first two slots, location and status, would be location of subordinates portrayed on a map and subordinate unit statistics, respectively. The last two slots require heuristics to fuse information to determine how well Red is doing and what Red should do for the next mission cycle.

Knowledge Sources (KSs) are developed to generate the information for each blackboard slot. For example, in order to answer "where are we," a KS is designed to ensure that each unit is counted only once from multiple reports concerning that unit. Another KS maintains the status of expendable stocks on board combat/combat- support vehicles. These KSs are probably stand alone and fuse incoming battlefield data. The KSs for "what are we doing now" and "future mission" use heuristics or access a knowledge-based system (KBS) which contains an expert's reasoning predicated upon the facts at hand.

Algorithms (ALG) calculate probabilities of kill or the likelihood of detection. Rule bases form networks of If–Then statements codifying an expert's problem-solving methods. Knowledge bases are RBs coupled with other data. Databases are files of information organized in a variety of formats.

Simulation Objects

The elements of the gaming plane shown in Figure 1 are represented in the computer as objects. Units, weapons systems, terrain, weather, and obscurants are developed in relation to object classes and instances. Each object consists of a file of characteristics and methods. The characteristics file describes what the object is, while the methods file describes how it reacts when activated by a message. A message is an event or action within the scenario. Gaming systems developed in this manner are scenario independent and work well in a free play environment.

Objects are organized into an inheritance hierarchy or lattice. Figure 3 is an example of such a lattice applied to Red-tracked vehicles. An object class can have multiple parents. Thus, a PT76 can inherit characteristics and methods from both the reconnaissance vehicle and tank.

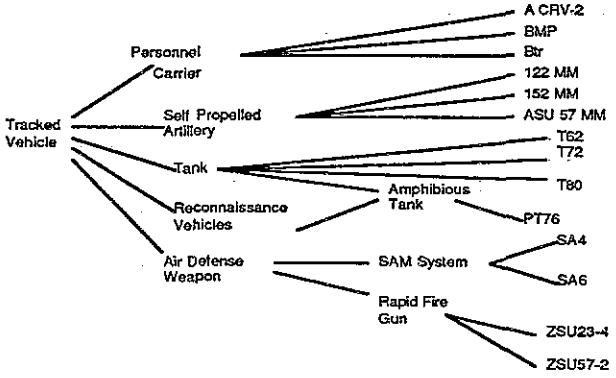

Figure 3, Example Showing Multiple Class Inheritance Lattice

Terrain characteristics can also be treated as objects. Figure 4 is a segmented piece of terrain (regular hex). Its characteristics are listed as items of important information concerning that piece of terrain. The behavior of the terrain is reflected in the behavior of the road (another object). The advantage of using a regular hex representation is that it is reasonably easy to recode map information into this format. A disadvantage is that the hex format does not lend itself to precise line-of-sight measurements.

There are several ways of displaying the terrain objects to the Red and Blue sides. One simple means is to display the hexagonal representation directly with simplified display for vegetation, towns, bridges, etc. This can be confusing. Figure 5 is more complicated than the hex display but more pleasing to individuals familiar with topographic maps. The tactical map image is displayed in one plane of the color graphics monitor, and the hex system is synchronized to the map but is transparent to the exercise staff.

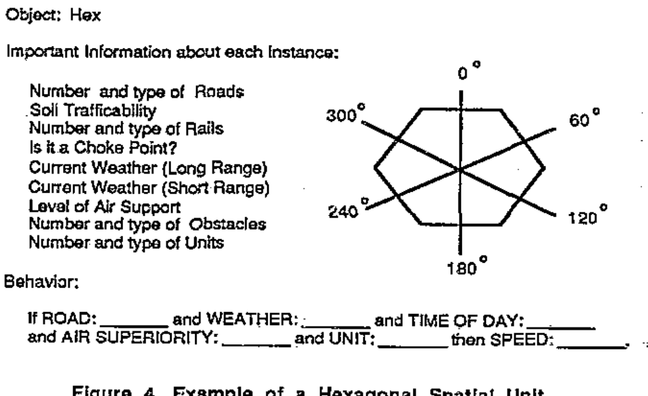

Figure 4, Example of a Hexagonal Spatial Unit Treated as as Object

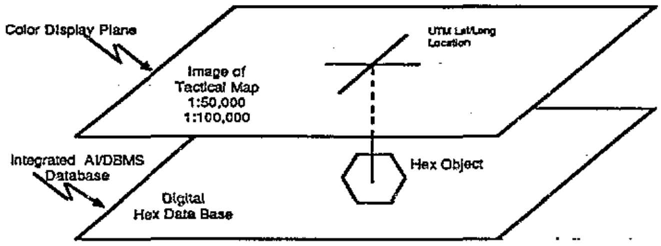

Figure 5, Correspondence Between Paper Map Image Displayed on Color Monitor and Underlaying Object Knowledge/Database

Aside from the hex system, there are other techniques that can be used to reduce mass storage needs for digitized terrain. Quadtrees decompose terrain into uniform terrain segments of varying sized rectangular areas. Another data compression technique is to call every "nth" data point. For semi automated force simulation, the customer should define the need for terrain displays, then an appropriate technique can be selected, based upon that need.

BLACKBOARD EXAMPLE--TANK COMPANY

The somewhat generic structure described in the preceding section can be applied to a variety of specific force levels and types. To illustrate a specific example, we next describe an instantiation of this structure for an opposing force (Red) tank company.

At the lowest level, Red company/battery-sized units are controlled at the weapons system level. Red weapons data and intelligence information concerning the Blue forces are the input to the Company Tank Blackboard, Figure 6. This operations blackboard maintains the status of the ten tanks in the Red company. For example, the company has been given an order consisting of:

- Mission — Movement to Contact;
- Company Boundaries — UTM Coordinates;
- Axis of Movement — Initial Point, Intermediate Points, Objective;
- Air Threat — Local Air Superiority; and
- Be Prepared Mission — Attack Opposing Forces from Route of March.

For the company, the player adjusts the position of each tank (as necessary) from a flexible template (Movement to Contact Formation), which is part of the Knowledge Source 1 — Movement to Contact. The flexible template aligns the platoons of the company at the initial point (IP) and contains the flexibility to allow the controller to fine tune the formation on the terrain. Once the formation is adjusted, each tank is a separate node on the blackboard which maintains the record of spatial relations within the company and with other units in the area (nominally within 3000 meters).

The knowledge sources (KS) shown in Figure 6 are not all inclusive. For example, the system requires combat resolution KS's and a data base with Lanchester equations or table look-ups to determine kill or no kill. Depending on the vehicle type a hull or turret kill (catastrophic, fire power, or mobility) must also be

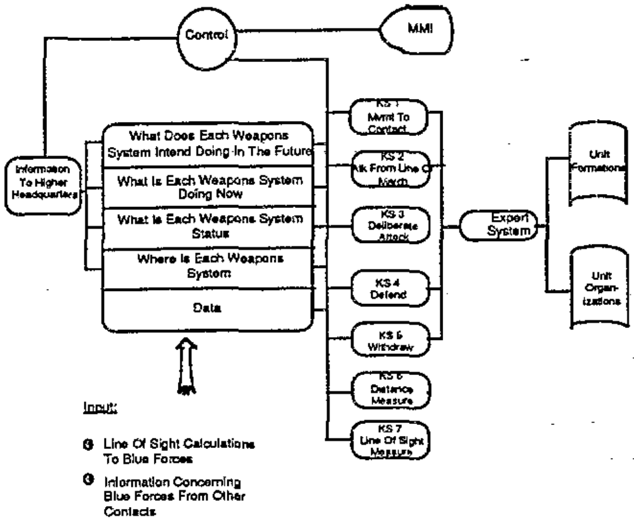

Figure 6, Company Tank (Motorized Rifle) Blackboard

determined. This KS should reside with an exercise controller or director, since battle outcome is a primary means of influencing game play.

For unit (or vehicle) operations, OOP techniques are used. Redundancy in the hierarchy of classes is *precluded by having the parent class characteristics* passed to the child class. Figure 7 is the company class hierarchy for tank and motorized rifle companies, and for artillery batteries. Of specific interest is the Tank class (Figure 8) which shows the characteristics associated to the parent (Tank class) and the child classes (Turret and Hull). From this class hierarchy, the specific instances for each of the ten tanks in the company are developed.

The Tank class provides information concerning the UTM coordinates as the tank moves toward its objective, what the tank is supposed to be doing, and how fast the tank is moving. The Turret class maintains

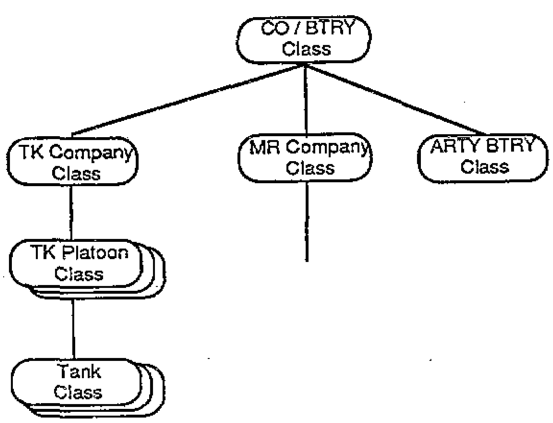

Figure 7, Company/Battery Class Hierarchy

a damage state (condition — no damage, fire power kill, catastrophic kill), amount of ammunition remaining on board (ammo — kinetic/heat/AP) and vector direction of the turret. *Similar statistics are supplied for the Hull* class: damage state (condition — no damage, mobility kill, catastrophic kill), the amount of fuel remaining

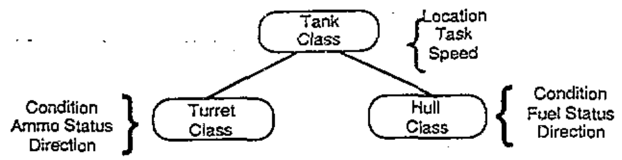

Figure 8, Company/Battery Class Hierarchy

(kilometers of travel remaining), and vector direction of the hull.

DTED/DFAD data is preprocessed in the form of terrain hexagonals as shown in Figure 9. Each hex is

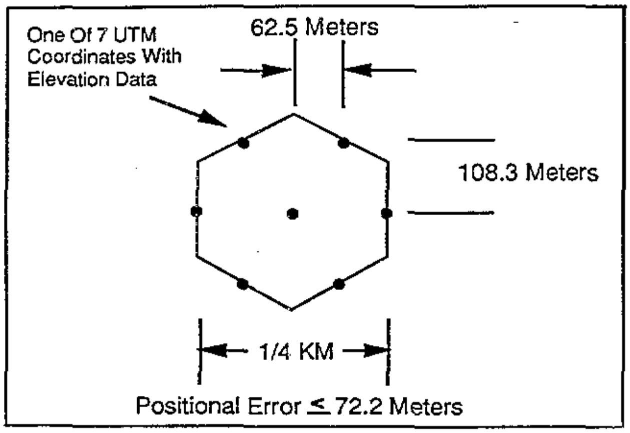

Figure 9, Hex Spatial Data

an object instance of the Terrain Hexagonal class of objects. Figure 10 provides additional detail to the terrain hex concept. Each hex has a 1/4-kilometer face-to-face spacing and contains the following information:

- roads — number/type,
- soil — trafficability,
- railroads — number/type,
- cities/towns,
- weather (short and long range),
- obstacles (mines, destroyed bridges, etc.),
- weapons systems (type/location), and
- elevation points.

There are seven elevation points: one at the center of each face and one in the hex center. These points are also synchronized to the 1/50,000 tactical map display and contain the UTM coordinates. Spacing between points is 62.5 meters horizontal and 108.3 meters vertical. Maximum positioning error is 72.2 meters. These data compare favorably with the DTED/DFAD 100-meter spacing.

The road (and rail) net is structured so that it passes through the points. Any two points can be connected to form a road segment within the hex. A tank *instance can travel either along the road or cross* country (between points) as long as the slope and soil trafficability allow movement.

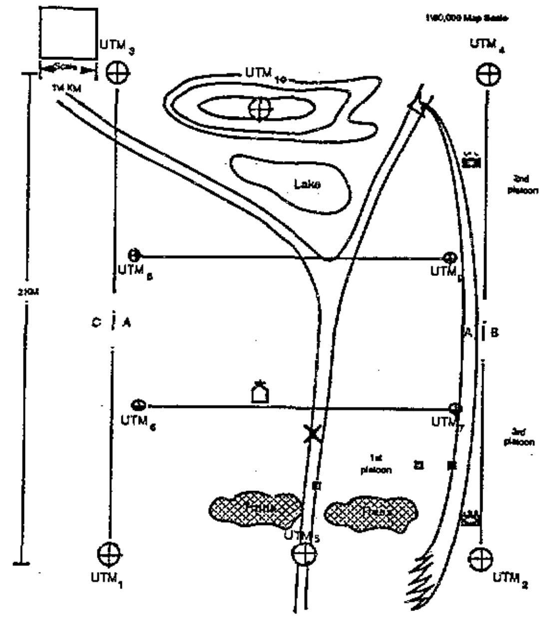

Figure 10, Attack from the Line of March (Operators Model)

A Tank Instance maintains information concerning the hull and turret direction (along with the other statistics) to be used in the packets of information for war fighting.

A detailed example of Figure 9 is presented in the follow-up Figures 11 and 12. Figure 11 is the operator's soft copy display of a 1/50,000 scale tactical map. Shown are the company boundaries (UTM_1 to UTM_3, UTM_2 to UTM_4), phase lines (UTM_6 to UTM_7, UTM_8 to UTM_9), initial point (UTM_5) and objective (UTM_{10}). The lead platoon (1st platoon) in march column emerges from the tree line road and immediately comes under fire from the high ground surrounding UTM_{10}. The company commander, accompanying the 1st platoon, determines that Blue strength is too great to continue the "Movement to Contact". After computer analysis the arrow shows the chosen path of attack for the 2^D and 3^D platoons while the remaining two tanks of the 1st platoon provide covering fires from the edge of the trees.

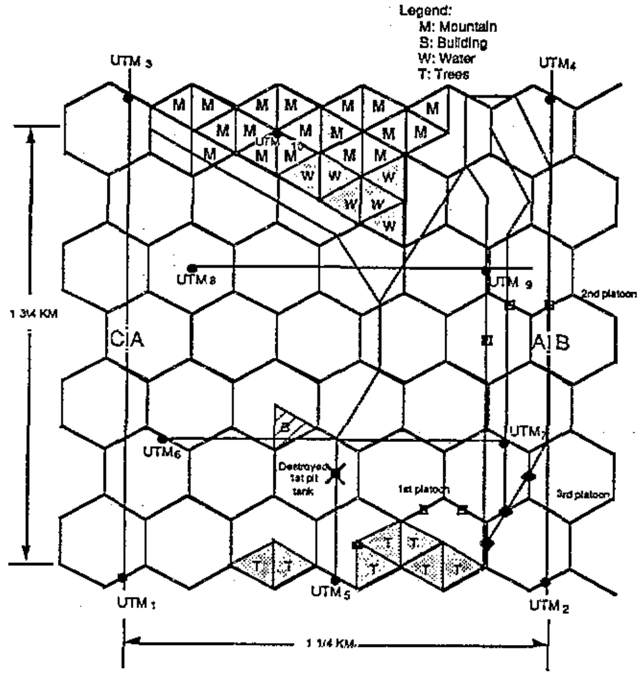

Figure 11, Attack from the line of March (Computer Model)

Figure 12 is the computer representation of the 1/50,000 scale map and the planned attack. Once the decision is made to "Attack from the Line of March" the KS2 performs the following operations.

- Searches for non-line of sight hexes within the company boundaries. The search is conducted from UTM_{10} outward eliminating those hexes that are impassible to vehicle movement,
- Establishes vehicle routes to the objective area using the least risk routes,
- Positions the 1st platoon tanks in locations to provide fire support,
- Directs the 2^D and 3^D platoons to attack along the established attack routes, and
- Requests artillery fire support on the objective at the appropriate time.

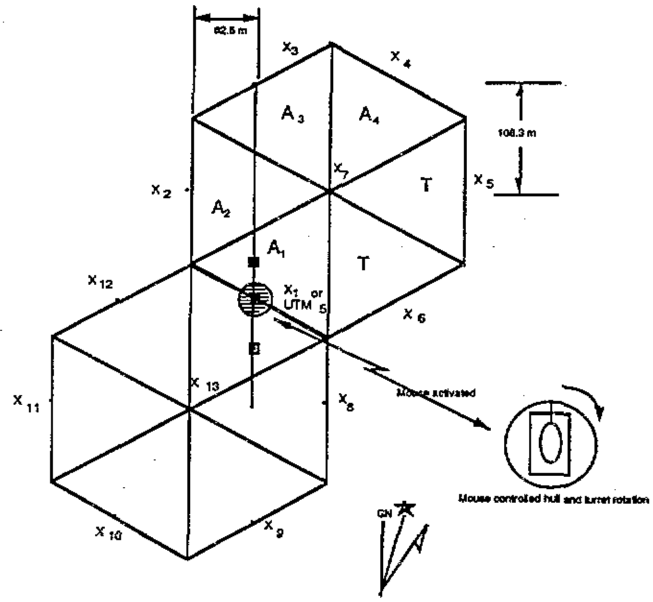

Figure 12, Hexagonal Object Class (Detail)

Figure 10 can also be expanded as shown in Figure 13. The initial position (UTM_5) is shown with the three tanks of the 1st platoon aligned on the road. The UTM positions X_{1-13} are registered on the soft copy 1/50,000 tactical map display. The tanks can travel on the road (X_1–X_3) or cross country from X_1 to any other point (X_{2-13}). The distance from X_1 to X_3 or X_8 to X_7 is

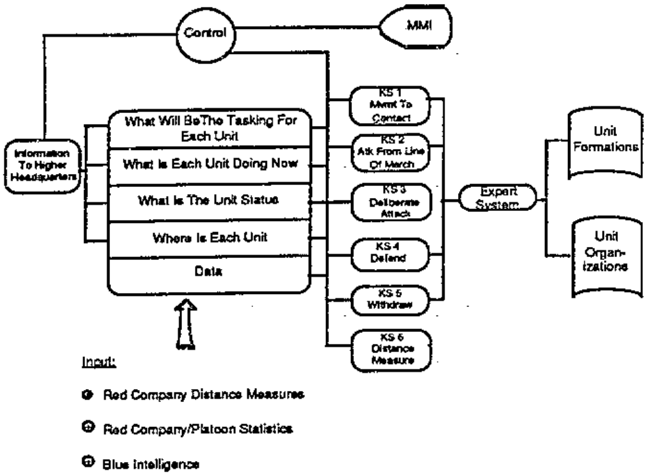

Figure 13, Battalion Tank (Motorized Rifle) Blackboard

216.5 meters. The distance from X_2 to X_3 is 125.0 meters and from X_2 to X_5, 250.0 meters. Note that to identify a tank traveling north, the horizontal components of the UTM coordinates (of two successive points) are equal. This distance is equal to \R(3) times 125 or \R(3)

$\overline{X_2 X_3} = \overline{X_1 X_3}$.

Each of the six equilateral triangles within the hex contain a mobility factor ($A_{1\text{-}4}$). These values are tagged to the UTM coordinates $X_{1\text{-}4}$. The T (tree) descriptions associated with X_5 and X_6 preclude vehicle travel within the triangles (as does an M, B, or L).

By mousing on a tank (black square) the icon for the tank appears (see Figure 13, lower right corner). By clicking within the circle (but not the turret) the hull can be rotated. By clicking on the turret, the turret can be rotated. Also, the tank can be moved to a new location using the mouse.

EXTENDING THE CONCEPT TO HIERARCHICAL CONTROL

Much of the information input to the Battalion Blackboard (Figure 14) comes from the Company Tank (or motorized infantry) Blackboard. However, some information can be generated from higher headquarters (regiment or Exercise Director) or from within the battalion itself. The battalion contains sensor assets such as radars and these systems can provide information, although they are normally used in intelligence gathering and would thereby indirectly support the Operations Blackboard.

Doctrinally, all units of the battalion execute the same tactics. That is to say, all units will attack or defend or withdraw although their manner of execution can be different. Battalions normally provide the following information to their subordinates:

- Mission,
- Company Boundaries,
- Axis of Movement,
- Air Threat, and
- Be Prepared Mission.

At this level verbal changes to a current order are common, though on a less frequent basis than company direction. Guidance is provided at least every 4 to 6 hours and is based on direct view of the ongoing operations.

The Battalion Tank Blackboard operates in a similar fashion as the Company Tank Blackboard. The major difference is that individual weapons systems are not shown in the Battalion Blackboard. The centroid symbol is displayed on the color graphics console (MMI) at company and platoon resolution. Nominally, platoon resolution is provided for the company to which the battalion commander's tank is attached. The other two companies are displayed to company resolution.

SUMMARY

Progress to date has included development of a prototype demonstrating key portions of the overall concept. A tactical map (1:50000) was copied to use as the background for the operational display. Synchronized to the map but transparent to the operator is a hexagonal grid system which contains the terrain description and elevation points. All objects operate from the hex plane but appear on the softcopy tactical map. The prototype includes a company of tanks which can be configured to an appropriate tactical formation. The object representation for the equipment allows separate but integrated representation of the hull and turret. Work is in progress on population of the knowledge sources for controlling the company activity. The initial prototype will contain knowledge sources for determining objectives, initial points, and phase lines; conducting as "movement to contact", and determining masked areas for coverage.

REFERENCES

[1] Englemore, R.S., Morgan, A.J., Nii, H.P., Blackboard Systems, Addison-Wesley, 1988.

[2] Nii, H.P., "Blackboard Systems (Part One): The Blackboard Model of Problem Solving and the Evolution Architecture," AI Magazine, Summer, 1986.

[3] Nii, H.P., "Blackboard System (Part Two): Blackboard Application Systems, Blackboard Systems from a Knowledge Engineering Perspective," AI Magazine, August, 1986.

[4] Barth, S., Barrett, S., and Gates, K., "A Blackboard Architecture for Identification of Command and Control Operations Nodes", Proceedings of the 1989 Tri-Service Data Fusion Symposium (DFS-89), May 1989.

ABOUT THE AUTHORS

Kermit (Kerry) Gates is Manager of the Expert Systems Department at PAR Government Systems Corporation. He has led the development team for a number of research and development prototype expert systems for the Army and Air Force in his 9 years at PGSC. Before joining PGSC, he served 24 years in the U.S. Army, with his final assignment the instruction of Soviet tactics and doctrine at the Army War College. He received his B.S. in General Engineering from the U.S. Military Academy, and his M.S. in Mathematics from Rensselear Polytechnic Institute.

Fred Frantz is the Technical Director for the Battle Management/Intelligence Systems Section at PAR Government Systems Corporation. He received his B.S. in Mathematics from Bucknell University, and his M.S. in Computer and Information Science from Syracuse University. He has been developing simulations and testbeds for battle management and intelligence systems with PGSC for over 12 years.