Proceedings of the 2000 Winter Simulation Conference
J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, eds.

ANTICIPATORY PLANNING SUPPORT SYSTEM

John R. Surdu

Department of Electrical Engineering
and Computer Science
United States Military Academy
West Point, NY 10996, U.S.A.

John M. D. Hill
Udo W. Pooch

Department of Computer Science
Texas A&M University
College Station, TX 77843-3112, U.S.A.

ABSTRACT

A new approach to military planning and execution has been proposed. This approach seeks to merge planning and execution, and replaces reaction to events with anticipation of events. This paper presents a methodology for building an automated system to support Anticipatory Planning. A Plan Description is developed to manage the many tree-like branches that occur in planning and execution of an operation. A Planning Executive can use the differences between the plan and the actual operation to control the activities of Planners and Execution Monitors in anticipating future branches to the plan. At the heart of the system are inference mechanisms for determining branches in the plan and simulations for predicting future states. This methodology enables the development of a prototype Anticipatory Planning Support System for evaluation of this new approach to military planning and execution. This paper concentrates on the activities of the Execution Monitors and their use of simulation to support those activities.

1 INTRODUCTION AND MOTIVATION

General (ret.) Wass de Czega has proposed a radically new approach to military planning and execution, which he calls Anticipatory Planning (Wass de Czege 1999). There are two main thrusts of the General's proposal. The first is that planning and execution should be treated as a tightly coupled, single process, rather than as distinct events. The second is that Anticipatory Planning is necessary in a dynamic and information-rich battlefield environment of the future.

In the traditional Military Decision Making Process (MDMP) various enemy courses of action (COAs) are posited by the intelligence officers, and the operations and planning officers propose various friendly COAs to counter them (U. S. Army 1997). Each of these friendly COAs are war-gamed in order to determine their viability. A COA is viable if it is suitable, feasible, and acceptable. *Suitable* means the COA accomplishes the mission and complies with the commander's guidance. *Feasible* means that constraints of available time, space, and resources are met. *Acceptable* means that the tactical or operational advantage gained justifies the cost in resources, especially casualties. Commanders often describe viability concerns in terms of desired end-state conditions at the conclusion of execution. The result of this analysis is a single, chosen COA for use in execution.

There is a well-known axiom that the plan never survives the first shot, which is another way of saying that a branch that was not considered in planning has occurred in execution. Consequently, the commander and staff are forced into a reactive planning mode. Rather than a long detailed plan stemming from comparisons of complete friendly and enemy COAs, the planners need a methodology that merges planning and execution. Such a methodology would develop and consider as many reasonable branches in the plan as possible in the initial planning process, and continuously update the plan as execution progresses. This coupling of planning and execution requires a new process.

According to Wass de Czega, it is futile to try to predict one most likely future and build a plan just for that case. Such plans have too little chance of survival. Uncertainty about the success of an operation is caused by clever, unpredictable enemy commanders who want to win as badly as do friendly commanders. Another source of uncertainty is how successful the friendly forces will be. Staffs are as often surprised by successes, which they are unable to exploit, as they are about slower than anticipated progress or higher than anticipated losses (Wass de Czege 1999).

What is needed, he argues, is to plan against as many of the enemy's options as possible, and to create a plan that addresses those most likely and most dangerous ones. The plan for the conduct of the upcoming (or currently being executed) operation must provide as many branches as planning time allows to deal with the next most likely or dangerous eventualities in priority. As a general rule, Wass de Czega argues, the initial course of action must be able to deal with several of the most likely eventualities with simple, “muscle movement” adaptations. The current generation of planning tools does not help planners generate the many-branched plans rapidly enough to stay ahead of the pace of decisions. Those that were available seemed too simplistic or attrition-paradigm oriented (Wass de Czege 1999).

The ability to develop and consider many branches in a plan necessitates an Anticipatory Planning process. Rather than choosing a single course of action and following it to conclusion, Anticipatory Planning involves maintaining as many possible friendly actions against as many enemy actions as possible. The plan is then considered to be a tree. The nodes of the tree represent states (i.e., snapshots of planned or anticipated dispositions of forces on the battlefield) and decision points in the plan. The branches represent the transition to a new state based on a particular enemy or friendly action. As new branches are developed, the Anticipatory Planning process will continue planning along those branches. In this way, Anticipatory Planning for a branch can be done well in advance and many options can be maintained as long as possible, rather than reactive planning once the branch occurs. Anticipatory Planning will increase the importance of the information collection plan to quickly confirm or deny the viability of branches.

New concepts coming out of the Information Technology Operations Center (ITOC) at West Point, NY, indicate that information operations (IO) stand on three legs, not two: offensive IO, defensive IO, and Information Efficacy (Ragsdale 1999). This research is designed to address the third.

2 SOLUTION STRUCTURE

This paper presents a methodology for an Anticipatory Planning Support System (APSS). See Figure 1 for a depiction of the methodology. The methodology will be implemented and evaluated in an Anticipatory Planning Prototype. For clarity, components of the methodology are capitalized.

Information from a World Integrator provides a World View that represents the actual status of execution. As discussed in Section 2.1, the location and/or status of some entities in the actual operation may be estimates. A Planning Executive controls the Anticipatory Planning process and the use of system resources. A Plan Description represents and manages the plan tree. Execution Monitors compare the Anticipated State of the plan derived from the Actual State of the operation with the Planned State at that Node and notify the Planning Executive if there is a potential problem.

The Planning Executive launches Planners to generate and evaluate new Branches. A Branch Generator uses inference mechanisms that consider possible friendly or enemy actions and produce new Branches. A Branch Evaluator examines a Branch to provide the Planner or the Planning Executive with viability measures and outcome confidences. The Execution Monitors and Branch Evaluators use simulations to perform their evaluations.

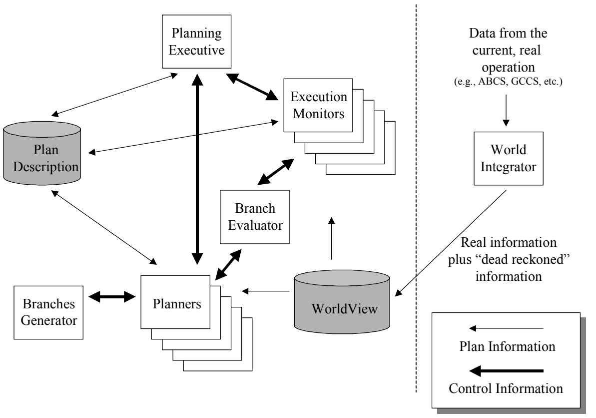


Figure 1: Anticipatory Planning Support System

The human planners will not accept or rely on the system unless they understand the system’s “logic.” If the recommendations of the system “make sense” to the human planners, or if the system provides a reasonable explanation capability, then it is more likely to be accepted and used. Regardless of how flexible and sophisticated the simulation and analysis system is, it still may not provide results that the planner will accept. Accordingly, the system provides the means for the planner to override the results with an outcome that makes more sense. This postpones the need to re-code the event resolution mechanism or the simulation.

2.1 World View and World Integrator

The methodology requires a representation of the Actual State of the operation. Surdu and Pooch describe the use of the World Integrator and a World View in providing the Actual State (Surdu and Pooch 1999).

The World View module is a representation of the real operation. In order to make the job of the Execution Monitors easier, the representation of the state of the real operation and the Plan Description should be as similar as possible. World View receives information about the state of the real operation through a series of APIs. It then transforms this information into a form that the Execution Monitors can easily interpret.

The World Integrator has the onerous task of monitoring the real operation, processing that information, and passing it to World View. In some systems, such as the Global Command and Control System (GCCS), this may involve querying a database (U. S. Army 2000). In other systems, this may require "eavesdropping" on the network. The reason for this intermediate step is that in real operations, reports on some entities may be intermittent. It is the job of the World Integrator to "dead reckon" these intermittent reports and pass them into World View. Clearly, when an entity has been "dead reckoned," this must be reflected in the information that World View gives to the Execution Monitors.

The World Integrator and World View involve issues in sensor, data, and information fusion. World Integrator must determine when an entity has been unconfirmed long enough that its actions must be dead reckoned. When some sensor reports a similar unit, World Integrator must determine whether this is merely the lost unit reappearing or a different unit. These and other issues regarding sensor, data, and information fusion are open research issues.

2.2 Planning Executive

The mission of the Planning Executive (PE) is to control the overall operation of the APSS. The PE creates and dispatches Execution Monitors (EMs) and Planners. The PE controls how many EMs and how many Planners are operating at any time, sets the maximum branching factor at any Node, and tracks the state of the (computer) system on which the APSS is running.

When an EM determines that re-planning should be conducted at a given Node, the EM gives the PE a handle to the Node in question and a certainty associated with its recommendation. The list of Nodes for which re-planning is required as well as those Nodes at which re-planning is currently being conducted is called the Planning Frontier (see Figure 2). Nodes to the right of the frontier in the figure have been nominated for re-planning by an EM, and Nodes to the left of the frontier have not been nominated.

The PE uses the confidence measures provided by EMs to determine which Nodes along the frontier will get Planners allocated to them and in what order they will be allocated. If the system is very busy, the PE may determine that it can only afford a small number of running Planners and so Planners will have to be allocated to Nodes sequentially based on the criticality of creating new Branches from the Node. If, however, the system is not busy, the PE may determine that it can afford to allocate a Planner to each Node along the frontier.

Similarly the PE determines how many EMs are running at any given time. Again, if the system resources are not heavily used, the PE might put EMs on many Nodes. On the other hand, in a resource-constrained situation, the PE might have only a few EMs that hop from Node to Node under the control of the PE.

The PE also receives inputs from the interface with the user. Through the interface, the PE allows the user to manually insert Branches or to override work being done by EMs or Planners. For instance, the commander may want to do a "what-if" analysis of some alternative action he has in mind. Through the interface and PE, this new Branch could be added to a Node and a Planner launched. The Planner will create new Branches and determine each Branch's viability. The commander might also want to manually delete a Branch, for whatever reason, and this is also done through the PE.

Finally, in a resource-constrained or very dynamic environment, it is possible that the creation of many Branches will exhaust available memory. In this case, the PE can set the maximum branching factor at Nodes to some small number (e.g., five). In this case only the five most-viable, representative Branches would be retained. This is similar to the combat simulation trajectory management research done by Gilmer, et al. (Al-Hassan, et al. 1997, Gilmer 1998, Gilmer and Sullivan 1996, Gilmer and Sullivan 1999, Gilmer, et al. 1997).

The level of autonomy of the PE is a tunable parameter. It is likely that the intuition of some commanders might be a better predictor of a Branch's viability than the decision of a Branch Evaluator. The user, therefore, might want to confirm the removal of all Branches.

By performing the actions described, the PE helps limit the scope of responsibility of the EMs and Planners as well as the search space that must be explored by them. The EMs and Planners do not need visibility of the global state of the plan or the Planning Frontier, just the Actual State of the operation. The EMs and Planners need to know only how to conduct their analysis or planning, respectively. This makes the job of designing and implementing EMs and Planners much more tractable. When dispatching a Planner, the PE must provide the Planner a handle to the Node in question, the Actual State, and the mission/objective of the operation. An EM only needs to know the Node - and its associated Planned State - that it is supposed to monitor as well as the Actual State.

2.3 Plan Description

The Plan Description is a representation of the possible ways the operation can proceed (see Figure 2 for a depiction). The Plan Description is a directed tree with the possible states of the plan held by Nodes. The Branches of the tree represent the significant changes between states caused by the actions of the friendly and enemy participants. These transitions can be the result of multiple actions by multiple entities (Normally in simulation literature, an event is that which causes a change in state. APSS is only concerned with *significant* changes in state, so Branches correspond more directly with transitions rather than with events, in the traditional simulation sense of "event.").

Note that the Plan Description is not a game tree for resolution of a minimax problem, in which each level represents a turn by the adversaries. Russell and Norvig describe the use of such a tree and the minimax algorithm (Russell and Norvig 1995). Each Node is the result of actions taken by *either* of the participants that result in a significantly new state.

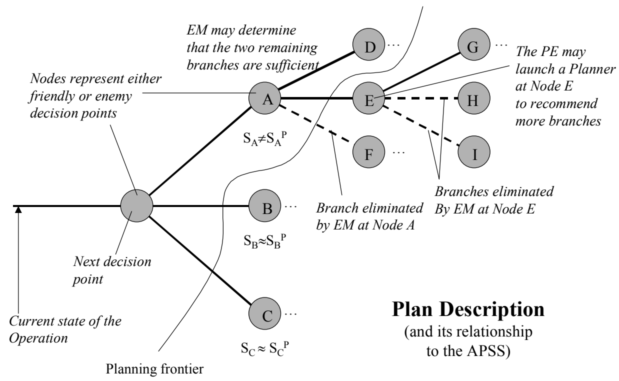


Figure 2: Plan Description

2.3.1 States

A state is the "minimal collection of information with which the system's future state can be uniquely predicted in the absence of chance events." (Pooch and Wall 1993) There are three kinds of states maintained in this system: the Actual State, the Planned State, and the Anticipated State. The Actual State comes from the World View. A Planned State is generated when a Planner initially creates a Branch in the plan, and is held in a newly created Node in the Plan Description. If an Execution Monitor is observing a Node, it periodically creates an Anticipated State by using simulations to project the Actual State forward to its observed Node.

2.3.2 Nodes

Each Node maintains a Planned State of the plan, as described above. The Nodes connect to any Branches that have been produced by Planners. The Nodes also provide an important function in communicating the viability measure associated with the Branches. Measures of viability are computed for Branches after planning or re-planning.

2.3.3 Branches

A Branch represents significant state transitions caused by actions taken by the friendly or enemy forces. This is similar to the action-based approach to planning Lansky presented in the COLLAGE system (Lansky 1994). The difference lies in the way that COLLAGE uses unsatisfied constraints to direct the execution of the system, whereas APSS incorporates a priority scheme that the Planning Executive uses to control when and how much planning is done. If the Planning Executive decides that further planning is required for a Node, a Planner is launched and given the state (Planned State or Anticipated State) of the Node. The Planner examines the outcomes of different possible transitions. The transitions have associated preconditions, viability measures, and a confidence measure. Within the constraints placed on the Planner by the PE, several of the best transitions become Branches in the Plan Description.

The commander may desire to add a decision point to the plan manually. The decision point is represented in the Plan Description as a Branch from whichever Node contains the state that holds when the decision has to be made. Then a Planner is used to assess the Branch's viability.

2.4 Execution Monitor

Execution Monitors have access to the Plan Description (PD) as well as the Actual State of the operation. The Executive can re-assign an EM to monitor another Node, but each EM is only concerned with one Node at any given time. The Actual State of the operation will be provided by real command and control assets, such as Maneuver Control System (MCS) or the Joint Common Tactical Database (U. S. Army 1998) through World Integrator and World View, as discussed in Section 2.1. The Joint Common Tactical Database does not currently exist, and there is no easy mechanism for pulling information from MCS, but this capability will exist within the near future.

The purpose of the Execution Monitor (EM) is to periodically compare the Planned State of the operation at a Node versus the Anticipated State at that Node extrapolated from the current Actual State. When the planner builds the various Branches from a Node, it also creates an initial Planned State of the operation at each new Node. An EM must infer when the Anticipated State of the operation differs "significantly" from the Planned State.

The EM uses simulation to create the Anticipated State. The simulation can be initialized with the current Actual State or by the Anticipated State generated by an EM analyzing an antecedent node in the Plan Description. For instance, assume EMs running on Nodes A, E, and C in Figure 2. The EM at Node C would have to begin with the current Actual State of the operation, while the EM at Node E could begin with the Anticipated State generated by the EM at Node A.

The Anticipated State is generated through simulation by applying the transitions leading from the Actual State to the Node of interest through each Node in between. During this simulation, the EM may discover that one or more of these transitions is impossible (e.g., required resources are not available, required entities no longer exist, etc.). In this case the EM terminates the simulation and immediately informs the PE.

When significant differences exist between the Anticipated State and the Planned State the EM at the Node performs several important tasks. First, it conducts a breadth-first traversal of the PD. At each Node in the PD, the EM determines whether the change in state invalidates any Branches leaving the Node. Recall that in the PD preconditions are associated with each outgoing Branch from a Node. When the differences between the Planned State and the Anticipated State indicate that conditions associated with a Node cannot be met, that Branch of the PD *may* be pruned.

Second, after this pruning has been completed, the EM must determine whether there are "enough" viable Branches from the state. A Planner has previously determined the viability of the Branches. EMs will also determine the likelihood of a Branch. For instance, in the absence of information the likelihood of being able to execute each of the three Branches leaving Node A in Figure 2 might be equal. When intelligence is gathered about enemy activities, for instance, an EM at Node A might determine that Node F is less likely. In any event, the likelihood as well as the viability (i.e., utility as shown in Figure 3) of Branches and the number of available Branches will be used to determine the overall utility of the Node on which the EM is operating.

While the exact computation of a Node's utility will be determined as part of this research, the EM will determine whether it thinks a Planner is needed to generate more options for the human user. If the EM thinks that there are insufficient Branches from a Node or that the viability of the existing branches is poor, the EM makes a recommendation to the PE with some measure of confidence. It is then up to the PE to allocate a Planner to the Node (as discussed previously).

It has already been noted that EMs have access to the Actual State of the operation via World View. EMs can make single, ad hoc queries of either World View or the Plan Description. In addition, so that EMs do not have to do exhaustive searches to look for needed information, EMs have the ability to "subscribe" to information from either the Plan Description or World View. For instance, if the Branch leading from Node A to Node F was based on the enemy moving in a certain direction, but the enemy has uncooperatively moved in another direction, an EM at Node A would probably want to know this. If an EM existed at Node A it could subscribe to information about enemy units within the geographic area of interest. In this sense, while Branches are analogous to Decision Points and Targeted Areas of Interest in military plans, subscriptions can be thought of as roughly analogous to Named Areas of Interest (U. S. Army 1993, U. S. Army 1997).

In addition to comparing the Anticipated State to the Planned State, the EM also looks at all conditions associated with the Node's Branches. The EM periodically checks each Branch's conditions and looks at the Actual State of the operation. If something necessary to fulfill a condition is eliminated (e.g., a mine-clearing device has been destroyed or an infantry company has been wiped out) the EM must notify the PE that the Branch should be considered for pruning.

Although it would be tempting for the PE to eliminate Branches that cannot be reached, this must be done with care. It may be possible that some event in a Node closer to the trunk of the tree will allow the condition to later be met. On the other hand, the PE should automatically prune Branches associated with conditions that can never be met, such as the destruction of a bridge or dam. Branches associated with conditions that might conceivably be met in the future should be retained. For instance, a battalion might receive another mine clearing device, replacement unit, sortie of close air support, or other assets from a higher headquarters. When a "recoverable" condition cannot be met, the EM should notify the PE, so that the PE can notify the user. *If the EM is monitoring a Node sufficiently far into the future, it might be possible for the user to take an action that will allow the condition to be met.*

Surdu and Pooch (Surdu and Pooch 1999) and Surdu, Haines, and Pooch (Surdu, et al. 1999) developed a system called OpSim, designed to monitor the current operation. The result of that reseach verified the feasibility of EMs as described here. OpSim uses a dynamic hierarchy of rational agents, called Operations Monitors to compare the current situation with the plan. The top-level Operations Monitor informs the decision maker when the success of the plan is at risk. OpSim, or a system like it, could be adapted for use as an EM. When OpSim was developed, the PD described in this research did not exist. OpSim could be modified to access and understand the PD. Then in addition to the inferences it makes based on state information, it could also look at whether conditions associated with Nodes can be fulfilled.

2.5 Planner

The planner receives a state (Planned State, Anticipated State, or Actual State) and a mission/objective from the Plan Executive. The Planner invokes a Branches Generator (BG) and passes it the state and mission/objective. The BG returns some number of Branches to the plan, along with their associated preconditions and confidence measure. At the end of the Branch is a new Node and the Planned State that the Planner predicts will exist after that Branch is followed. In an unconstrained environment, the Planner continues to execute a BG at each newly created Node until either the desired end state is reached or the BG determines that the desired end state cannot be reached. The PE can place constraints on the Planner that limits the planning in terms of time, depth, system resources, etc. A Branch Evaluator (BE) evaluates each Branch and returns a viability measure.

If the Planner is operating on a Node with existing Branches (i.e., the Node has already been run through a Planner, but now has an Anticipated State different from the Planned State), the Planner compares the newly generated Branches to the existing Branches. If a new Branch is the same as an old Branch, the old Branch can be considered revalidated. If an old Branch is not revalidated based on the Anticipated State, the Planner notifies the PE that the Branch may be considered for Pruning.

After the Planner is finished, the new Nodes at the end of the Branches may or may not be explored further. It is up to the Plan Executive to decide whether to place Execution Monitors on those Nodes and whether to act on any recommendations from the EMs for further planning.

2.5.1 Branches Generator

The Branches Generator (BG) receives and examines a state and a mission/objective, then uses inferencing systems to generate different options. Prototype systems such as Fox-GA (Schlabach, et al. 1998), Tactical Event Resolution (Hill, et al. 2000), and the modified version of ModSAF used by Porto, et al. (Porto, et al. 1999) have demonstrated the feasibility of automatic generation of courses of action in the military domain. The output of the BG is some number of distinct transitions, the Planned State that will hold after each transition, and the associated confidence measures. The new Planned State will contain differences in the conditions of the entities (battle damage, destruction) and in resource consumption (ammunition, fuel, time). The BG creates a new Branch for each of the transitions, and at the other end of the Branch creates a new Node containing the Planned State.

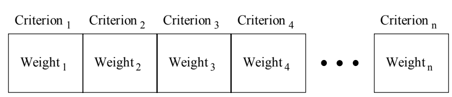
2.5.2 Branch Evaluator

The Branch Evaluator (BE) is given a Branch to evaluate and the mission/objective. The BE compares the Planned State at the end of the Branch with the desired end state of the operation. Using an inference mechanism, the BE determines the impact of the action associated with the Branch on the feasibility, acceptability, and suitability of the plan (i.e., its viability). If the plan is in danger of failure at the new state, the Branch is assigned a low viability measure. If there is little danger of failure, the Branch is assigned a high viability measure. These viability measures are first generated at the leaves and propagated back up the tree. Execution Monitors will use this viability measure when they analyze Nodes.

The purpose of this methodology, along with maintaining as many viable options for the commander as long as possible, is to reduce the "search space" of creating and monitoring the operation. To this end, the system is not designed to explore every possible option. Additionally, branches that have low viability ratings can be pruned so that the evaluation of their children does not consume resources. This increases the difficulty of computing the viability of branches. There are two choices:

- simulate the branch until the simulated operation reaches end state or fails or
- determine a means of inferring the viability of the Anticipated State based on action indicated by that Branch.

The problem with the first alternative is that it requires all subsequent branches between the current Node and the end state or failure to be generated. This defeats the purpose of limiting the search space of the overall system. The challenge with the second option is to develop heuristics which adequately predict the likelihood of achieving an end state precondition based on the current Anticipated State. Given that such heuristics exist, the computation of the viability, or utility, of a branch is as shown in Figure 3. The Likelihood function uses the heuristics to generate a number in the range [0, 1] of the *likelihood* that a given criterion (i.e., precondition for successful end state) will be achieved based on the current Anticipated State. The *weight* is a user-tunable parameter, again in the range [0, 1], that allows the user to rate the importance of the various criteria. For the initial prototype, the set of usable criteria would be fixed, but eventually the user would be able to choose some number of criteria from a list.



$$\text{Utility} = \sum_{i=1}^{n} (\text{Weight}_i)(\text{Likelihood}(\text{Criterion}_i))$$

Figure 3: Computation of Branch Utility

2.6 Simulations

Surdu, Haines, and Pooch describe the requirements for operationally focused simulations (Surdu, et al. 1999). Operationally focused simulations are those specifically designed for the mission operational environment. Fishwick, et al., (Fishwick, et al. 1996) and Blais and Garrabrants (Blais and Garrabrants 1999) have identified the benefits that can be gained from using simulation to support planning. A variety of simulations are needed to support this system, ranging from high to low resolution. For instance, the level of resolution required for the Planner would be less than the level required for the Execution Monitors. Time or system resource constraints may dictate that Planners and EMs be able to select the simulation with the appropriate resolution to provide "good enough" answers "fast enough." It is likely that these simulations will need to be designed specifically for this system.

This methodology does not rely on any particular simulations. Any simulation used to support Anticipatory Planning has to provide a Planned State after an action is taken, the list of preconditions required for that action, and the confidence of achieving that Planned State. It must also be able to accept as inputs a state from the Plan Description.

All but the simplest simulations should consider terrain effects. Terrain representation is necessary for event resolution, route and travel time determination, and fuel or other resource consumption determination. A minimal representation would include elevation and GO / SLOW-GO / NO-GO (U. S. Army 1993, U. S. Army 1997) depiction of the terrain. The terrain fidelity can be as high as permissible for efficiency and timeliness.

The simulation should be flexible and sophisticated enough to handle decomposable events. Multiple levels of resolution will allow APSS to adapt to time and system resource constraints. For instance, the Planner might ask the simulation to resolve a company breach operation. If the Planner requires more detail, the system should be able to individually resolve the support force engagement, the breach force execution, and the assault force. Similarly, the system should be able to resolve a battalion versus company event as four companies versus one company, four companies versus three platoons, or twelve platoons versus three platoons.

3 ANTICIPATORY PLANNING PROTOTYPE

The Anticipatory Planning prototype will be developed to analyze and validate the methodology. A Graphical User Interface (GUI) with drag-and-drop icons and right-click functionality will simplify the interface between human planners and the prototype.

A tool bar will contain all of the controls necessary for the construction of a task organization and initial state. This will include selecting units by icon, organization by drag and drop, visual placement of icons and control measures, and right-click functionality on the icons to view properties or specify actions.

The GUI will contain a Branch Display section. In this section the previous, current, and subsequent Branches will be visible as lines. The user will select a Branch by clicking on it, which will cause it to move to the center and previous/subsequent branches expanded or contracted as appropriate. Some properties of the Branch will be visualized, such as the viability measure (color) and the confidence (thickness). The user can right-click on the Branch for more detailed information. A zoom-out feature will pop up another window where the entire Plan Description can be viewed and Branches and Nodes examined. The user will also be able to add or delete branches manually through the Branch Display. The user will also be able to specify a path through the tree and place the system in "replay" mode in which the actions associated with each branch on the path are played back in the Action Display.

The largest visible area of the prototype will be the Action Display. When a Branch has been selected in the Branch Display, the Action Display will represent the state at the beginning of the Branch. This will include the appropriate map background, the placement of icons, and any control measures in the area. A playback button will allow the user to view the action that takes place during the selected Branch and observe the outcome(s).

Of course, the system will allow the user to save the entire Plan Description for later retrieval. The user will be able to start and stop the Anticipatory Planning process at will and observe the modification of the Plan Description as the operation is executed. The planners at each level and on the flanks will ultimately be able to exchange Plan Descriptions produced by the prototype system.

4 CONCLUSIONS

This research is not intended to produce a fully autonomous planning system. Human military planners do not really want a system to do all of the planning for them. They want a system that supports their planning by taking over the mundane tasks, keeping track of the possibilities, and helping them determine whether the plan is viable.

The Anticipatory Planning process accounts for the chaotic nature of warfare in which possibilities appear and disappear. With the advent of information age technologies, U.S. military planners should have the capability to plan faster and better and stay inside the enemy decision cycle (U. S. Army 1993). The Anticipatory Planning process, aided by the automated support system, will prove to be a decisive advantage.

REFERENCES

Al-Hassan, S., J. B. Gilmer, Jr. and F. J. Sullivan. 1997. A simulation state management technique sensitive to measures of effectiveness. In *Proceedings of the 1997 SCS Simulation MultiConference: Military, Government, and Aerospace Simulation*, ed. M. J. Chinni, San Diego, CA: Society for Computer Simulation.

Blais, C. L. and W. M. Garrabrants. 1999. Simulation in support of mission planning. In *Proceedings of the Advanced Simulation Technologies Conference (ASTC '99): Symposium on Military, Government, and Aerospace Simulation (MGA 2000)*, ed. M. J. Chinni, 117-122. San Diego, CA: Society for Computer Simulation.

Fishwick, P. A., G. Kim and J. J. Lee. 1996. Improved decision making through simulation based planning. *Simulation* 67 (5): 315-327.

Gilmer, J. B., Jr. 1998. Alternative implementations of multitrajectory simulation. In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros, E. F. Watson, J. S. Carson and M. S. Manivannan, 865-872. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Gilmer, J. B., Jr. and F. J. Sullivan. 1996. Combat simulation trajectory management. In *Proceedings of the 1996 Simulation Multiconference: Military, Government, and Aerospace Simulation*, ed. M. J. Chinni, San Diego, CA: Society for Computer Simulation.

Gilmer, J. B., Jr. and F. J. Sullivan. 1999. Multitrajectory simulation performance for varying scenario sizes. In *Proceedings of the 1999 Winter Simulation Conference*, Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Gilmer, J. B., Jr., F. J. Sullivan and S. Al-Hassan. 1997. Eaglet multitrajectory combat simulation research project. Available as `<http://calvin.mathcs.wilkes.edu/mts/Papers/index.html>` [accessed June 19, 2000].

Hill, J. M. D., M. S. Miller, J. Yen and U. W. Pooch. 2000. Tactical event resolution using software agents, crisp rules, and a genetic algorithm. In *Proceedings of the Advanced Simulation Technologies Conference: Military, Government, and Aerospace Simulation*, 15-21.

Lansky, A. L. 1994. Action-based planning. In *Proceedings of the Second International Conference on Artificial Intelligence Planning Systems (AIPS-94)*, 110-115.

Pooch, U. W. and J. A. Wall. 1993. *Discrete Event Simulation: A Practical Approach.* Boca Raton, FL: CRC Press.

Porto, V. W., M. Hardt, D. B. Fogel, K. Kreutz-Delgado and L. J. Fogel. 1999. Evolving tactics using levels of intelligence in computer-generated forces. In *Proceedings of the Proceedings of SPIE: Enabling Technology for Simulation Science III*, ed. A. F. Sisti, 262-270. Bellingham, WA: SPIE, The International Society for Optical Engineering.

Ragsdale, D. J. 1999. Information operations concepts. Personal communication.

Russell, S. and P. Norvig. 1995. *Artificial Intelligence: A Modern Approach.* Englewood Cliffs, NJ: Prentice-Hall.

Schlabach, J. L., C. C. Hayes and D. E. Goldberg. 1998. FOX-GA: A genetic algorithm for generating and analyzing battlefield courses of action. *Evolutionary Computation* 7 (1): 45-68.

Surdu, J. R., G. Haines and U. W. Pooch. 1999. OpSim: a purpose-built distributed simulation for the mission operational environment. In *Proceedings of the International Conference on Web-Based Modeling and Simulation*, 69-74. San Diego, CA: Society for Computer Simulation.

Surdu, J. R. and U. W. Pooch. 1999. Connecting the operational environment to simulation. In *Proceedings of the Advanced Simulation Technology Conference: Military, Government, and Aerospace Simulation*, ed. M. J. Chinni, 94-99. San Diego, CA: Society for Computer Simulation.

U. S. Army. 1993. *FM 100-5: Operations.* Washington, DC: Headquarters, Department of the Army.

U. S. Army. 1997. *Field Manual 101-5: Staff Organization and Operations.* Washington, D.C.: U.S. Government Printing Office.

U. S. Army. 1998. *Staff Leaders Guide for the Army Battle Command System.* Washington, D.C.: U.S. Government Printing Office.

U. S. Army. 2000. Global command & control system - army. PM STCCS. Available as `<http://160.147.21.82/wsdocs/stccs/gcssa.htm>` [accessed January 26th, 2000].

Wass de Czege, H. 1999. Anticipatory planning. Personal communication.

AUTHOR BIOGRAPHIES

JOHN R. SURDU works as an Assistant Professor and a Senior Research Scientist at the United States Military Academy in the Information Technology and Operations Center. He received his Ph.D. in computer science from Texas A&M, his M.S. in computer science from Florida State University, his MBA from Columbus State University, and his B.S. from the United States Military Academy. He is a member of ACM, IEEE, and SCS. His email and Web addresses are `<surdu@acm.org>` and `<www.itoc.usma.edu/Surdu>`, respectively.

JOHN M. D. HILL is a Ph.D. candidate in the Department of Computer Science at Texas A&M University. He received his M.S. in computer science from the University of Texas, Austin, and his B.S. from the United States Military Academy. He is a member of ACM, IEEE, and SCS. His email and web addresses are `<hillj@cs.tamu.edu>` and `<www.cs.tamu.edu/people/hillj>`, respectively.

UDO W. POOCH is E-Systems Professor in the Department of Computer Science at Texas A&M University. He received his Ph.D. in theoretical physics from Notre Dame and his B.S. in physics from the University of California, Los Angeles. He is a member of ACM, IEEE, and SCS. His Email and Web address are `<pooch@cs.tamu.edu>` and `<www.cs.tamu.edu/faculty/pooch>`, respectively.