

# ANTICIPATORY PLANNING USING EXECUTION MONITORING AND A CONSTRAINED PLANNING FRONTIER

JOHN M. D. HILL, JOHN R. SURDU, and UDO W. POOCH  
{ hillj | surdu | pooch }@cs.tamu.edu  
Department of Computer Science  
Texas A&M University  
College Station, TX 77843, USA  
Tel: 1-409-845-9216

## ABSTRACT

A new approach to military planning and execution has been proposed, called Anticipatory Planning, that seeks to merge planning and execution and replace reaction to events with anticipation of events. This paper describes a methodology for building an Anticipatory Planning Support System. A Plan Description is developed to manage the many tree-like branches that occur in planning and execution of an operation. A Planning Executive uses 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 while constraining the planning space. At the heart of the system are inference mechanisms for determining branches in the plan and simulations for predicting future states.

## KEYWORDS

Simulation, Planning, Decision Support

## INTRODUCTION

Traditional military planning has focused on wargaming a few carefully considered friendly courses of action against the most likely or most dangerous enemy courses of action. A new approach that more closely follows the merged planning and execution that occurs during military operations has been proposed. This paper presents a methodology for building a support system that enables this new anticipatory approach to military planning. Previous work in adversarial planning, simulation, artificial intelligence techniques, and software agents provide some insights into how such a system can be built. The contribution of this paper lies in development of a methodology for keeping as many reasonable options open for as long as possible, while still minimizing the amount of information stored and constraining the planning effort along the most-important paths.

## Nature of the Problem

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 [1]. Each of these friendly COAs are wargamed 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.

In military planning 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 new approach that merges planning and execution to help the staff anticipate events rather than react to them.

This Anticipatory Planning approach will benefit from automation of many of its functions. The coupling of planning and execution and the anticipation of events will require a well-designed methodology. 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. The final requirement would be the implementation of an Anticipatory Planning Support System.

## Previous Work

There has already been some work done on planning in dynamic or adversarial domains and on plan representation that draw on traditional artificial intelligence planning techniques. Similarly, artificial intelligence methods and simulation have been used to support Course of Action (COA) generation and analysis for military planning. Some other work has focused on reducing options to representative sets, execution monitoring, and the combination of execution monitoring and planning. The use of software agents to support military planning is becoming more prevalent as well.

### *Planning in Dynamic or Adversarial Domains:*

Applegate, et al., examined the additional complexity that AI planning techniques face in adversarial situations [2]. This complexity arises because the environment is unpredictable and dynamic, the plan must be adjusted dynamically during execution, and the presence of an adversary must be considered during plan development. This means that the AI planner must make assumptions about the outcome of events, and re-plan as necessary. These assumptions can then be monitored during the execution of the plan.

Elsaesser described an Adversarial Planner (AP) which addresses the complexity of battle planning by limiting the problem space through determination of the adversary's counter-plans, monitoring of execution, and re-planning when the original plan is in jeopardy of failure [3, 4]. AP uses task decomposition planning to develop a complete plan in layers, and defers expansion of the plan until it is required. Counter-planning is used to represent possible adversary plans and determine ways to defeat them.

Atkin, et al., describe the use of a multi-goal partial hierarchical planning approach to planning in continuous, uncertain, adversarial real-time domains [5]. They developed the General Reasoning using Abstract Physics (GRASP) planner to address resource allocation among multiple goals, determining plan operator effects, reacting to and exploiting unforeseen events, and generating workable plans quickly.

### *Plan Representation:*

Applegate uses an approach to plan representation that places less significance on states of the world than on the derivation and persistence of desired conditions in the world [2]. Leaves in the plan representation are orders to units rather than discrete executable actions, but still require many of the aspects of traditional AI planners (list of actions, temporal constraints, variable bindings, and preconditions). Applegate's approach, based on agents that represent units, is different from the approach taken by this research, but many of her points are still relevant. For instance, intelligence gathering gains added

importance, since knowledge of the actual situation will trigger re-planning.

### *Artificial Intelligence Support for Course of Action Generation*

Several prototype systems have demonstrated the feasibility of using artificial intelligence techniques for automatic generation of courses of action in the military domain. Schlabach and Hayes used a genetic algorithm for generating courses of action in the Fox-GA system [6]. Hill and Miller used software agents, crisp rules, and a genetic algorithm to resolve tactical events [7]. Porto, et al., used a modified version of ModSAF to demonstrate the feasibility of using evolutionary algorithms to solve platoon-level tactical problems [8].

### *Simulation Support for Military Planning*

Fishwick, et al., [9] and Blais and Garrabrants [10] have identified benefits gained from using simulation to support planning. Surdu, Haines, and Pooch describe the requirements for operationally focused simulations specifically designed for the mission operational environment [11].

### *Reducing Options to Representative Sets*

Gilmer and Sullivan are working on management of multiple outcomes resulting from an event in a stochastic simulation [12]. Rather than allowing "trajectories" along every possible outcome, they restrict the possible outcomes to a representative set with associated probabilities. They have reported positive results in converging towards the set of representative outcomes, but caution that there are still some limitations to the approach [13]. Al-Hassan has investigated the use of measures of effectiveness to prevent the problem of discarding interesting outcomes that have low probabilities [14]. The modified system is designed to be sensitive to loss ratios while still determining representative outcomes.

### *Execution Monitoring*

Hansen presented an approach that includes the cost of sensing into the determination of which actions to take in the plan [15]. Hansen notes that in many realistic environments some actions have different associated risks and/or "error prone-ness" associated with them. In such environments a variable-interval may be more appropriate. Although not directly applicable to military planning in a complex, adversarial, real-time environment, this work does suggest the utility of including an intelligent sensing strategy into a combined execution/planning system. Cohen, et al., discuss a technique for determining an efficient monitoring interval they call the Interval Reduction Strategy [16]. The Interval Reduction Strategy can help determine an efficient monitoring schedule but relies on the agent being able to sense the entire state at each monitoring attempt and makes no allowance for the cost of monitoring.

Surdu and Pooch [17] developed a system called OpSim designed to monitor the current operation. The result of that research 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 Execution Monitor (EM). When OpSim was developed, the Plan Description (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.

#### *Combining Execution Monitoring and Planning*

St. Amant, et al., describe the idea of "envelopes" in dynamic programming to monitor the progress of an agent in accomplishing its goal [18]. Essentially, the envelope represents the portion of the state space in which the goal can still be reached. When the agent moves across the boundary of the envelope it is certain to fail in achieving its goal. The important idea here is that the agent ought to be aware of the envelope and avoid the boundaries. This approach, based on a single agent monitoring its own progress towards accomplishing the goal and constructing its own plans, can be extended into the military planning domain. The Anticipatory Planning Support System presented in this paper uses Execution Monitors to watch progress, and the comparison between Anticipated States and the Planned States to predict failure and initiate re-planning.

Wilkins and Myers describe the CYPRESS system that provides the framework for the creation and control of taskable, reactive agents [19, 20]. Taskable, reactive agents have two main components: an executor and a planner. The executor constantly monitors the world state for situations requiring it to take action. The planner synthesizes sequences of actions that serve as a template for later refinement by the executor. Communication and coordination between the executor and the planner are accomplished through what they call the ACT formalism [20].

#### *Software Agents Support for Military Planning*

Atkins's GRASP planner extends the partial hierarchical planning framework by explicitly representing multiple goals and integrating the planner into an action hierarchy, Hierarchical Agent Control (HAC), that handles resource arbitration and failure recovery [21]. HAC provides a general skeleton for controlling agents and for management of sensing information, scheduling of actions, and message passing. In HAC, a forward simulation process evaluates plans that denote an action that satisfies a goal. In this way, the planner determines what would happen if the plan were executed.

## **Contribution**

The Anticipatory Planning methodology and the Anticipatory Planning Support System implementation rely on new applications of techniques from several areas, such as artificial intelligence, planning, inference mechanisms, evolutionary algorithms, and software agents. These techniques have been modified for application to the dynamic and uncertain domain of military operations, while retaining control over the computation effort and the size of the planning space. The result is a new system for military planning that keeps as many reasonable options open for as long as possible, while still minimizing the amount of information stored and constraining the planning effort along the most-likely and most-important paths.

## **ANTICIPATORY PLANNING**

General (ret.) Wass de Czege has proposed a radically new approach to military planning and execution, which he calls Anticipatory Planning [22]. 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.

According to Wass de Czege, 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 [22].

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 Czege 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 [22].

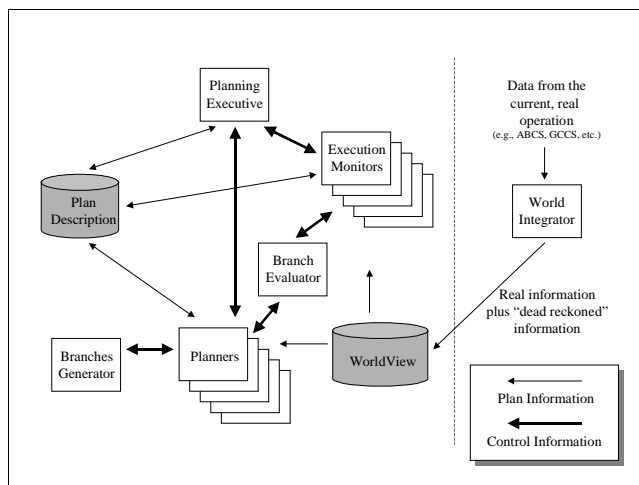
The ability to develop and consider many branches in a plan necessitates an Anticipatory Planning process. Rather than choosing a single course of action and following it to conclusion, Anticipatory Planning involves

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 [23]. This research is designed to address the third leg.

## ANTICIPATORY PLANNING SUPPORT SYSTEM DESIGN OVERVIEW

This section presents the design of an Anticipatory Planning Support System (APSS) that implements the Anticipatory Planning Methodology [24]. See Figure 1 for a depiction of the methodology. For clarity, in this discussion components of the methodology are capitalized.



**Figure 1: Anticipatory Planning Support System**

Actual or estimated information from a World Integrator is provided to a World View that maintains the Actual State of execution. 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, derived by forward simulation 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.

## TRACKING THE ACTUAL OPERATION

The methodology requires a representation of the Actual State of the operation. Surdu and Pooch describe the use of a World Integrator and World View mechanism in providing the Actual State [25]. The World Integrator filters information about the actual operation and forwards it to the World View for maintenance of the Actual State.

### World Integrator

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 [26]. 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 and Planners.

### World View

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.

### Issues

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.

## REPRESENTING THE PLAN

One of the most significant pieces of this research is the development of a method for representing the plan and storing all of the planned Nodes and Branches, all the while remaining cognizant of space and computational limitations. The Plan Description is being developed to provide such a capability.

### 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 something that causes a change in state. APSS is only concerned with significant changes in state, so Branches correspond more directly with transitions composed of several actions rather than with the traditional notion of an event.

### 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.” [27] 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.

### 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.

### Branches

A Branch represents significant state transitions caused by actions taken by the friendly or enemy forces. 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 Planning Executive (PE), several of the best transitions become Branches in the Plan Description.

## CONTROLLING PLANNING AND MONITORING

The purpose of the 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.

### Operation of the Planning Executive

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 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.

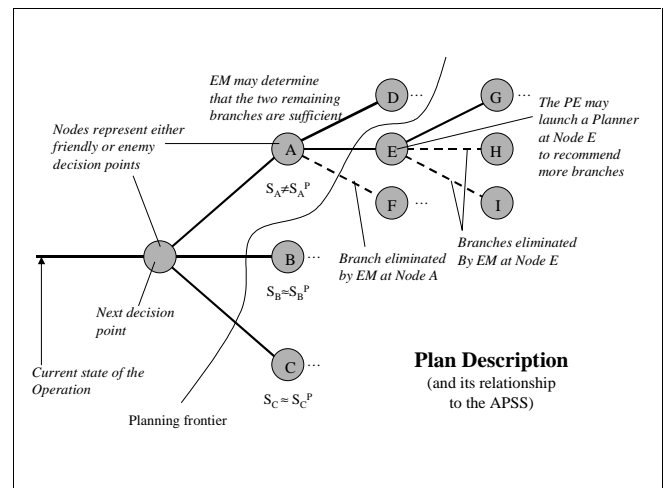


Figure 2: Plan Description

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.

### **Constraining the Planning Frontier**

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 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 migrate from Node to Node under the control of 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). 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. [12-14].

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.

### **DETECTING PLAN DEVIATIONS**

One of the key components in constraining the amount of planning effort and the size of the plan space is to remove branches (and all subsequent states and branches) when they are no longer viable. The APSS uses Execution Monitors to observe the Actual State and identify problems with previously Planned States. Each

EM uses forward simulation to arrive at an Anticipated State. If there is a significant difference between the two, the EM will identify branches that may be pruned and apply a utility computation to the remaining branches. The EM will then notify the Planning Executive of its recommendations for pruning and re-planning.

### **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 is provided by real command and control assets through World Integrator and World View. The Joint Common Tactical Database does not currently exist, and there is no easy mechanism for pulling information from the Maneuver Control System (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.

### **Forward Simulation**

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.

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 [28] 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.

### Response to Significant Differences

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.

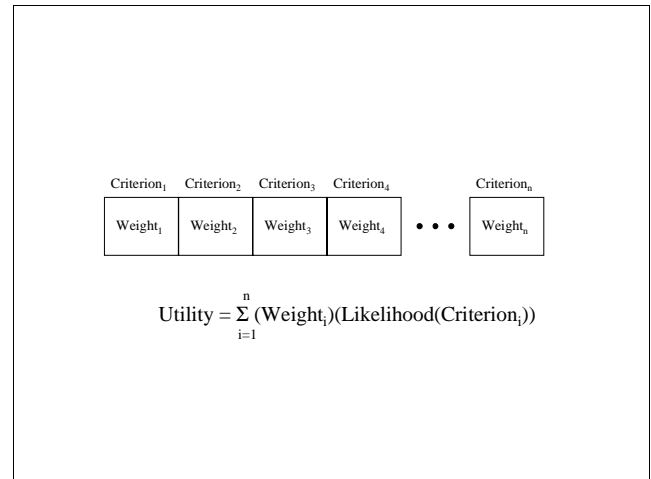


Figure 3: Computation of Branch Utility

### Utility Computation

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 [1, 28].

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.*

## PLANNING AND RE-PLANNING

The Planning Executive uses Planners to determine possible branches in the plan and determine their likelihood and viability. The Planners do not operate on the entire plan and they do not try to plan to conclusion. Rather, the Planners operate on small pieces of the plan, allowing the Planning Executive to focus planning effort along more likely or more important avenues.

### 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.

### Branches Generator

The Branches Generator (BG) receives and examines a state and a mission/objective, then uses inference systems to generate different options. 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.

### 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. 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 goal of limiting the search space of the overall system. The challenge with the second option is to develop heuristics that 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. Eventually, the user would be able to choose some number of criteria from a list or the system could employ machine-learning techniques.

## APSS PROTOTYPE

The Anticipatory Planning prototype is being 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. 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. One important note: since there is currently no system to provide the sensor fusion capabilities of the World Integrator into a World View, a simulation will be used to represent the actual operation.

### *Scenario Construction*

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. Of course, the system will allow the user to save the entire Plan Description for later retrieval.

### *Branch Management*

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.

### *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).

## CONCLUSION

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 [28]. The Anticipatory Planning process, aided by the automated support system, will prove to be a decisive advantage.

## REFERENCES

- [1] U. S. Army, Field Manual 101-5: Staff Organization and Operations, Washington, D.C.: U.S. Government Printing Office, 1997.
- [2] C. Applegate, C. Elsaesser, and J. Sanborn, "An Architecture for Adversarial Planning," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 20, No. 1, January/February 1990, pp. 186-194.
- [3] C. Elsaessar, "Adversarial Planner User Guide," Available at <http://www-leav.army.mil/nsc/warsim/reason/adplan/index.htm>, March 21st, 2000.
- [4] C. Elsaessar, "Adversarial Planner," Available at <http://www-leav.army.mil/nsc/warsim/reason/adplan/getstart/eagle/index.htm>, March 21st, 2000.
- [5] M. Atkin, D. L. Westbrook, and P. R. Cohen, "Planning in Continuous Adversarial Domains," Submitted to the 17th National Conference on Artificial Intelligence, Austin, Texas, July 30th - August 3rd 2000.
- [6] J. L. Schlabach, C. C. Hayes, and D. E. Goldberg, "FOX-GA: A Genetic Algorithm for Generating and Analyzing Battlefield Courses of Action," Evolutionary Computation, Vol. 7, No. 1, 1998, pp. 45-68.

- [7] J. M. D. Hill, M. S. Miller, J. Yen, and U. W. Pooch, "Tactical Event Resolution Using Software Agents, Crisp Rules, and a Genetic Algorithm," *Advanced Simulation Technologies Conference: Military, Government, and Aerospace Simulation*, Washington, DC, April 17-20 2000.
- [8] V. W. Porto, M. Hardt, D. B. Fogel, K. Kreutz-Delgado, and L. J. Fogel, "Evolving Tactics using Levels of Intelligence in Computer-Generated Forces," *Proceedings of SPIE: Enabling Technology for Simulation Science III*, Orlando, Florida, April 6-8 1999, pp. 262-270.
- [9] P. A. Fishwick, G. Kim, and J. J. Lee, "Improved Decision Making Through Simulation Based Planning," *Simulation*, Vol. 67, No. 5, 1996, pp. 315-327.
- [10] C. L. Blais and W. M. Garrabrants, "Simulation in Support of Mission Planning," *Advanced Simulation Technologies Conference (ASTC '99): Symposium on Military, Government, and Aerospace Simulation (MGA 2000)*, San Diego, California, April 11-15 1999, pp. 117-122.
- [11] J. R. Surdu, G. Haines, and U. W. Pooch, "OpSim: a Purpose-built Distributed Simulation for the Mission Operational Environment," *International Conference on Web-Based Modeling and Simulation*, San Francisco, CA, 17-20 January 1999, pp. 69-74.
- [12] J. B. Gilmer, Jr. and F. J. Sullivan, "Combat Simulation Trajectory Management," *1996 Simulation Multiconference: Military, Government, and Aerospace Simulation*, New Orleans, Louisiana, April 8-11 1996.
- [13] J. B. Gilmer, Jr. and F. J. Sullivan, "Multitrajectory Simulation Performance for Varying Scenario Sizes," *1999 Winter Simulation Conference*, Phoenix, Arizona, December 5-8 1999.
- [14] S. Al-Hassan, J. B. Gilmer, Jr., and F. J. Sullivan, "A Simulation State Management Technique Sensitive to Measures of Effectiveness," *1997 SCS Simulation MultiConference: Military, Government, and Aerospace Simulation*, Atlanta, Georgia, April 6-10 1997.
- [15] E. A. Hansen, "Cost-Effective Sensing During Plan Execution," *Twelfth National Conference on Artificial Intelligence (AAAI-94)*, Seattle, Washington, August 1-4 1994, pp. 1029-1035.
- [16] P. R. Cohen, M. S. Atkin, and E. A. Hansen, "The Interval Reduction Strategy for Monitoring Cupcake Problems," *Third International Conference on Simulation of Adaptive Behavior*, Brighton, UK, August 8-12 1994.
- [17] J. R. Surdu and U. W. Pooch, "Simulation Technologies in the Mission Operational Environment," *Simulation*, Vol. 74, No. 3, March 2000, pp. 138-161.
- [18] R. St. Amant, Y. Kuwata, and P. R. Cohen, "Monitoring Progress with Dynamic Programming Envelopes," *Third International Conference on Simulation of Adaptive Behavior*, Washington, DC, November 5-8 1995, pp. 426-433.
- [19] D. Wilkins and K. L. Myers, "Integrating Planning and Reactive Control," *Third International Symposium on Artificial Intelligence, Robotics, and Automation for Space*, Jet Propulsion Laboratory, Pasadena, California 1994.
- [20] D. E. Wilkins and K. L. Myers, "A Common Knowledge Representation for Plan Generation and Reactive Execution," *Journal of Logic and Computation*, Vol. 5, No. 6, 1995, pp. 731-761.
- [21] M. Atkin, D. L. Westbrook, and P. R. Cohen, "AFS and HAC: Domain-General Agent Simulation and Control," *AAAI Workshop on Software Tools for Developing Agents (AAAI-98)*, March 22-24, 1999 1998, pp. 89-95.
- [22] H. Wass de Czege, "Anticipatory Planning," Consultant, U. S. Army Training and Doctrine Command Advanced Warfighting Experiments, Fort Monroe, Virginia, October, 1999.
- [23] D. J. Ragsdale, "Information Operations Concepts," *Information Technology and Operations Center*, West Point, New York, October, 1999.
- [24] J. M. D. Hill, J. R. Surdu, and U. W. Pooch, "A Methodology to Support Anticipatory Planning," *Advanced Simulation Technologies Conference: Military, Government, and Aerospace Simulation*, Washington, DC, April 17-20 2000.
- [25] J. R. Surdu and U. W. Pooch, "Connecting the Operational Environment to Simulation," *Advanced Simulation Technology Conference: Military, Government, and Aerospace Simulation*, San Diego, CA, 11-14 April 1999, pp. 94-99.
- [26] U. S. Army, "Global Command & Control System - Army," Available at <http://160.147.21.82/wsdocs/stccs/gcssa.htm>, January 26th, 2000.
- [27] U. W. Pooch and J. A. Wall, *Discrete Event Simulation: A Practical Approach*, Boca Raton, FL: CRC Press, 1993.
- [28] U. S. Army, *FM 100-5: Operations*, Washington, DC: Headquarters, Department of the Army, 1993.