FOX-GA: A Planning Support Tool for Assisting Military Planners in a Dynamic and Uncertain Environment

C. C. Hayes
University of Minnesota
111 Church St. S. E.
Minneapolis, Minnesota 55455-0111

J. L. Schlabach
Tech. Integration Office, Office of
Deputy Chief of Staff for Intelligence
Dept. of the Army, Pentagon, VA.

Abstract

This paper describes FOX-GA, an intelligent planning decision support tool for assisting military intelligence and maneuver battlestaff in rapidly generating and assessing battlefield courses of action (COAs). The motivations behind FOX stem from the need to plan and replan rapidly in such a way as to allow users flexibility and control over planning objectives and options. The environment in which plans are executed (the battlefield) is one that is inherently uncertain and rapidly changing, demanding frequent re-planning during execution. Planners, however, are limited, especially at crisis times; there are far more relevant plan options to be considered than time and resources allow. To help meet these rapid replanning needs, we designed a number of efficient representations containing only the most essential high-level aspects of the battlefield and the plans. This use of an "engineering approximation" allows FOX to very rapidly generate and evaluate a broader variety of variety of high quality COAs faster than they could do so themselves. Because the situation in inherently uncertain, plans are assessed in a variety of likely situations, and their average performance is determined. The more situationally robust plans tend to be rated more highly. Fox presents only a few, most highly rated options to users, and allows them to reassess those options according by their own judgment and select a small subset. Initial evaluations on domain experts have been very promising.

1 Introduction

FOX-GA is an intelligent planning decision support tool designed to assist military intelligence and maneuver battlestaff in rapidly generating and assessing battlefield courses of action (COAs). The motivations for FOX stemmed from the need to rapidly plan and re-plan in such a way as to allow users flexibility and control over planer inputs determining objectives, and planner outputs in terms of options selected for further development.

Challenges. The environment in which COA generation and assessment takes places is one that is inherently uncertain, and rapidly changing. The situation is uncertain both because information gathered about the current situation is incomplete and inaccurate, and because and enemy intentions are only partially predictable. Because of the uncertain nature of the current situation and the future course of the battle, it is often (but not always) desirable to create plans that are robust under a wide variety of situations. Additionally, since the battlefield situation is rapidly changing, it frequently renders previous plans irrelevant and demands frequent re-planning during execution (i.e. the course of the battle).

Planners, however, are limited. There are often too few of them, particularly recently given institutional moves to reduce the size of battlestaffs. At crisis times, these few planners are bombarded with more information than they can process, and the search space of relevant plan options is enormous (there several million the coarse-grained representation used in FOX. This number multiplies when these abstract representations are further refined.) There are far more relevant plan options that should be considered than planners possibly have the time and resources to investigate. Thus, planners have a very difficult job. Both their time, attention and computational resources are severely constrained, while at the same time they must do their job quickly and accurately since the battle outcome and many lives are at stake. In practice, there is a limit to how well planners can perform under such constraints; even expert planners can only explore a few major options, and it is not uncommon to miss some possibly important ones. Given the challenges of the task, it is amazing that they perform as well as they do. However, planners are clearly in dire need of planning support tools that can assist them to perform better. The payoff in doing so is very great.

Current methods. Many standard solution procedures for addressing dynamic and uncertain situations, some hundreds of years old, have already been mapped out and recorded in military doctrine. For example, since it is uncertain what the enemy will do planners identify several possible, and likely, enemy COAs. (*enemy COAs* are actions the enemy will take, while *friendly COAs* are actions that "our" side takes, although they would not be considered very friendly by the other side.) Next they identify several possible friendly COAs which can be used in response to the possible enemy COAs. To reduce uncertainty further, intelligence staff use the friendly and enemy COAs developed by the planning staff to develop a very focused information collection plan which will help to identify which of the many possible enemy COAs the enemy appears to be putting in place, which in turn will help the commanders decide with greater certainty which friendly COA to follow (at least initially, until the situation changes).

In theory, these methods can be very effective in dealing with a dynamic, uncertain, and incompletely known environment. However, the problem lies in the limited time and computational resources available; a small team of planners with limited time can process information only so fast. Thus, in practice fewer COAs are explored than might be, and less re-planning happens during the battle less than is desired. The planning process is too time consuming to do it more often or more throughly.

Approach. Our goal in this work has been to produce a planning support tool, FOX-GA, that can assist battle planners by fitting in with and augmenting their existing methods; this is not only convenient but necessary since users are unlikely to accept tools that require drastic changes to tried-and-true procedures. Essentially, our objective was to find ways to give the current planning methods a "power boost."

We chose to focus on the specific subtask of helping planners (at the brigade level and below) to generate a wide variety of friendly COA options more rapidly. At Fox's starting point, a mission statement of what is to be accomplished has been handed down from above, and the battlestaff have already identified a small set of likely enemy COAs. FOX uses a genetic algorithm (GA) to generate candidate COAs and a coarse-grained wargaming simulator to rapidly evaluate them. Each COA is simulated in battle against all enemy COAs currently under consideration, so that its robustness under many circumstances can be assessed. Its average performance in the face of all considered enemy situations is used in determining the COA's "fitness." FOX evaluates thousands of COAs in a few seconds to minutes, then presents the user with a small set of most fit COAs (10 to 20). The user then re-assesses these COAs in their head (they may or may not agree with FOX's fitness scores) and selects a handful (3 to 5) for more detailed development. If the user does not like any of the COAs, he or she can adjust the input parameters (such as risk, GA population size, etc) and try again.

A detailed user evaluation is scheduled for summer 1998, so as of yet, we do not have detailed results describing how FOX affects user performance. However, early demonstrations of FOX to military users have yielded very promising results.

In terms of lessons learned, we found that 1) there was a very important speed/accuracy trade-off to be considered. The key in this application (and many) was to quickly generate and evaluate a number of reasonable plan options. The uncertainty of the situation greatly increased the cost of the evaluations since performance of each plan had to be examined under a variety of conditions. However, rough assessments were good enough. Highly accurate performance simulations and robustness assessments would probably have been counterproductive. 2) consideration of human-computer interaction (HCI) issues, such as control over inputs and final choices, flexibility, and representations chosen presentation of results, played a surprisingly large role in the usability, success and acceptance of FOX. We suspected that these issues would be important, but the degree to which they were important was a surprise (if not a shock).

2 FOX-GA

One of the largest bottlenecks in rapidly generating and evaluating large numbers of COAs in the search for a few good ones, was the evaluation process. Typically, COAs are evaluated by simulating battles under various conditions and seeing how well they perform. The focus in most battle simulations (designed for training purposes) is accuracy. Each battle may require several hours to run. Using such simulations to evaluate COAs is not practical when one needs to evaluate thousands. To address this challenge we developed an efficient wargamer. It is efficient because it uses coarse-grained representations of the problem domain. Some accuracy is lost, but the results are still sufficiently accurate to allow Fox to identify COAs which the users find reasonable. The coarse-grained representations allow appropriate yet intelligent trade-offs to be made between computational efficiency and accuracy. The need to make trade-offs between efficiency and detail through use of estimates is a common one found in many engineering applications.

An additional challenge was that users in this domain (and probably in most decision support applications) prefer to have a diverse set of significantly different plan options from which they can choose, each offering different trade-offs. Standard GAs tend to develop a group of "best" solutions that are all very similar or identical. We addressed this challenge by adding a niching strategy to the selection mechanism to insure diversity in the solution set, providing users with a more satisfactory range of choices.

3 Background Concepts

The terrain on which a unit fights is the *maneuver box.* Maps provide very detailed representations of the terrain inside a maneuver box. However, FOX-GA requires very specific abstractions which battlestaffs draw from maps in order to plan. Figure 1 shows an example of these abstract representations. The broad gray arrows represent avenues of approach (AAs). AAs 1-3 are wide routes down which subordinate units will move in an attack. Typically there are two to five AAs in any given maneuver box. The large oval areas on the left are Tactical Assembly Areas (TAAs). They are the starting locations for the friendly units. The ovals to the right represent terrain objectives (OBJs) that the friendly forces wish to capture. The Forward Edge of the Battlefield (FEBA) is the demarcation line between

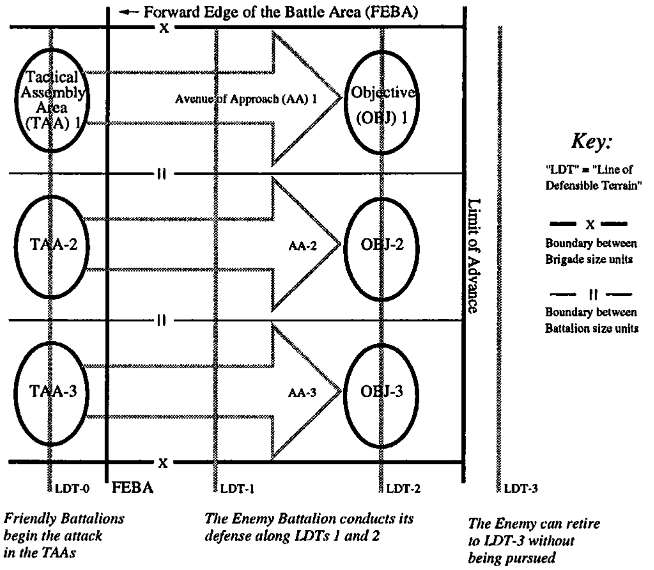

Figure 1: The Terrain Maneuver Box for FOX-GA

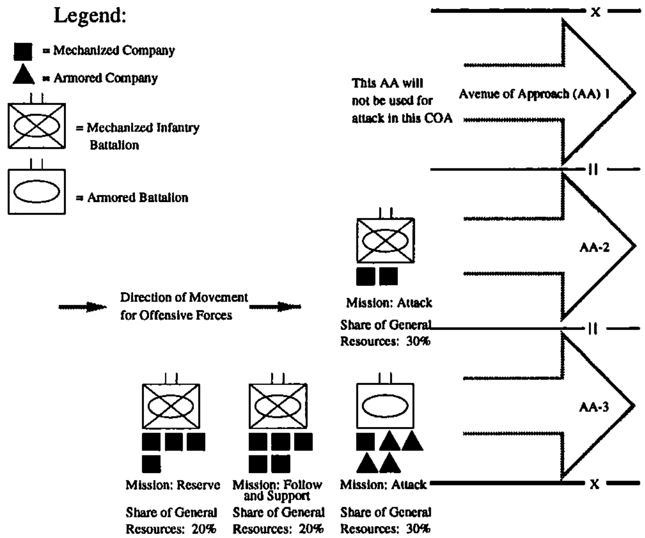

Figure 2: An Example of an Offensive FCOA

enemy and friendly forces *prior* to battle, and the Limit of Advance (LOA) is the demarcation *after* battle (if the attacker is completely successful). The gray vertical lines in Figure 1 represent lines of defensible terrain (LDTs). LDTs are formed by identifying a string of roughly adjacent narrow areas or *choke points* cutting across all AAs. Although LDTs appear as straight lines in the abstract maneuver box, they are not usually straight lines when plotted on a map. If obstacles are placed in each of the choke points along an LDT it forms a "dam" to movement, thus LDTs are usually good positions at which to set up a defense. They are also places at which conflicts between attacker and defender tend to occur.

In its most abstract form a COA assigns and sequences subordinate units in the various AAs. In other words, a COA only specifies the starting positions for the various players. It does not define a specific and detailed set of steps (yet) for the reason that one does not know initially what will happen on the battlefield. Instead the plan is defined by a set of doctrinally recommended behaviors (if situation x arises do y) which are captured in the wargaming simulator. By simulating a given friendly COA against a given enemy COA using the simulator, one can predict how the COA will "play out" in that situation. Thus, a given COA does not represent one particular plan sequence, but a whole family of situationally dependent plans, each of which can be animated and viewed on the map if the user desires.

Figure 2 shows an example of an offensive COA where a mechanized battalion attacks along AA-2 and the three other battalions attack in column along AA-3. There are many possible ways of organizing a set of subordinate units into COAs in a given maneuver box. COAs are also distinguished by the mission, number and type of subordinates, and share of general resources assigned to each unit during battle.

4 Genetic Algorithms

A Genetic Algorithm (GA) is a search technique inspired by the theory of natural evolution. GAs created successive "populations" of solutions, evaluate those solutions, and select the best ones for survival to the next "generation" of solutions. GAs apply techniques such as *crossover, selection,* and *mutation.* Crossover enables existing solutions to combine their best attributes in their children, sometimes producing a child that is better than either of the two parents. Selection ensures that only the best solutions survive to the next generation. Mutation introduces new schemas into the population by making minor random changes to some selected percentage of the population.

A common technique used in genetic algorithms is *niching*, which preserves *diversity* within a population. Without a niching mechanism, a GA tends to create populations that contain multiple copies of the same best solution. Niching ensures that distinct solutions survive in each generation.[2].

5 Related Work

The work most closely related to FOX includes the AirLand Battle Management (ALBM) Project [1] (developed at Lockheed Missiles and Space Company), and the Systems for Operations Crisis Action Planning (SOCAP) [5] developed at Stanford Research Institute (SRI). Both systems focused on the automated generation of COAs. FOX-GA can be contrasted to these systems because in addition to generating COAs, it also assesses how well the COAs would perform in combat against an anticipated enemy. Two GA applications in other military domains investigated decision making in the simulation of small unit actions [4] and control of Naval Surface-to-Air weapons [3].

However, the domains of these applications provide only limited insights into the use of GAs for COA generation for battlestaffs.

6 Approach

The current implementation of FOX-GA operates on one particular scenario, in which a mechanized infantry brigade attacks an enemy mechanized infantry battalion. FOX-GA generates and evaluates friendly COAs for this scenario. Work is currently underway to generalize FOX to operate on a broad range of scenarios. Evaluation is performed using an efficient wargaming algorithm to play each friendly COAs against several possible enemy COAs. This efficiency of the wargamer comes from using the abstract maneuver box terrain representation (shown in Figure 1) which leaves out the detail typically used in training simulations, but retains the information pertinent for COA evaluation. The resulting increase in speed enables FOX-GA to evaluate significantly more COAs per minute than is possible otherwise.

Inputs: FOX-GA takes two types of inputs: 1) a set of user specified parameters, and 2) a set of fixed parameters (specified inside the program) which define a battlefield scenario. User inputs include: allowable risk, status of friendly forces, status of enemy forces, fitness evaluation mode (terrain or enemy – to be explained later), population size and number of generations for the GA. The scenario specification is made up of the terrain maneuver box (Figure 1), six scripted representative enemy COAs and a description of available friendly forces. The six enemy COAs are generic representations of typical enemy options.

Outputs: FOX-GA outputs a set of the best COAs. The number returned is determined by the user.

7 System Description

FOX-GA was implemented in C++. Its major components include a 1) bit string used by the GA to represent COAs, 2) genetic algorithm which uses a niching strategy, 3) combat wargamer used to estimate losses incurred in battle using each COA in a variety of situations, and 4) fitness function used to assess the quality of the COAs based on wargaming.

Bit String Representation. FOX-GA uses the 36 bit string representation in Figure 3. It allows for approximately 45 billion valid friendly COAs after subtracting the 20 billion illegal combinations. The bit string is composed of ten variables which represent COA properties of importance to battlestaff planners in wargaming. These variables are used to create COAs and evaluate their performance.

Genetic Algorithm. FOX-GA employs a genetic algorithm for search, and a niching strategy to maintain population diversity. The GA implements single point random crossover with mutation using elitist tournament style selection without replacement. The mutation scheme provides for one bit chosen at random immediately after crossover. First, an initial population of COAs is generated randomly. Second, a new generation is created. Third, each member is evaluated by a fitness function, and lastly, the best COAs are selected to survive to the next generation. The last three steps are repeated for as many generations as the user chooses. To produce a new population, FOX-GA mates each member of the old population with another COA string at random. The crossover point between the two parents is chosen at random. Mating produces two children, which are then subjected to a small mutation rate (2.5 percent). Finally, the children are checked to ensure that they represent valid friendly COAs. Each time a new generation of COAs is created, the quality (or fitness) of each COA is estimated using a two step process., 1) Each COA is run through a combat wargamer that uses force ratios[1] to assess the probable losses that would be incurred on each side in a variety of situations. The combat wargamer (to be described below) generates parameters such as "terrain captured," and "remaining enemy strength," to describe their performance. 2) A fitness function (to be described below) uses the parameters generated by the wargamer to compute a fitness score for each COA. Fitness is a function of parameters such as remaining friendly strength, enemy strength, and gains or losses in terrain holdings. The fitness for each COA score is computed by averaging its performances against all 6 enemy COAs. Lastly, the COAs deemed most fit according to their fitness scores, are selected for survival to the next generation. This cycle continues until the number of generations specified by the user is reached.

Niching Strategy. Since users prefer plan options presented to them to each offer distinctly different trade-offs, the selection made by the elitist tournament strategy is constrained by a niching strategy. The niching strategy ensures that FOX-GA will return a wide variety of distinctly different COAs, rather than small variants on the best COA it finds.

Most niching strategies find niches as part of the niching process. However, since it was of critical importance in this application that users be able to understand the outputs of the GA, we were concerned that the GA's niches might not make sense to the domain experts. Thus, in order to match the domain experts' concept of niches (i.e. all solutions in one niche appear to the user to be essentially similar variants of each other), it was necessary to implement a variation of existing niching strategies that imposed the domain experts' concept of niches, rather than letting the GA find its own niches.

In FOX's domain, niches are defined by the most important variable, "formation of subordinates." Formation of subordinates refers to the physical positioning of the various subordinate units in each of the avenues of approach (AA); each different value for this parameter defines a dis-

[1] Force ratios are a traditional method of computing the losses inflicted on each side during battle. This system of modeling combat has been empirically derived from historical studies of combat.

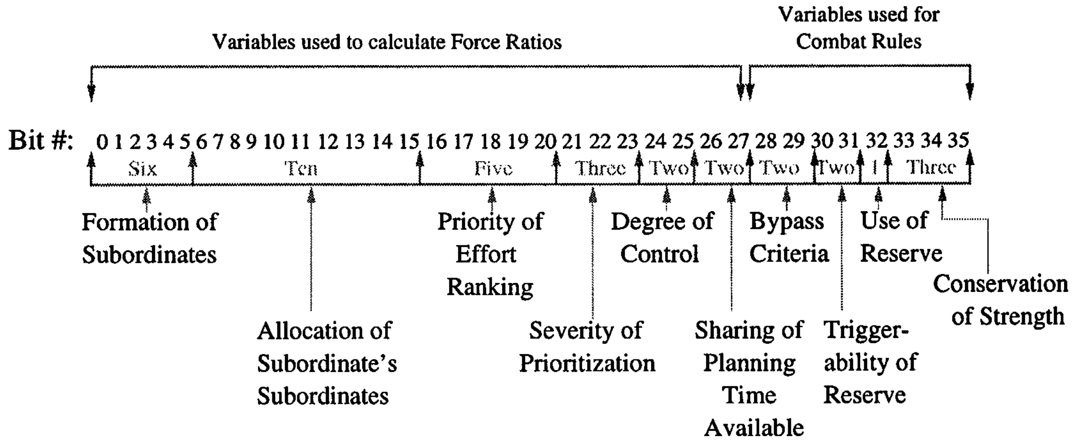

Figure 3: FOX-GA's Bit String Representation for tactical Courses of Action (COAs)

tinct family of COAs in the user's mind (in other words, a niche). These families of COAs are also easy for the user to visually distinguish from one another when displayed on the terrain box or on a map.

We implemented a variation of a deterministic crowding strategy for niching the COAs which we call *fixed niching*, in which these domain relevant niches are imposed on the strategy a priori. In this strategy, when two parent COAs produce two child COAs, we keep only two: the most fit of those 4 COAs, and the second most fit that is a different niche from the best. This creates a bias towards preserving solutions which the user views as different from each other.

Combat Wargamer. The majority of the implementation effort (approximately 95 percent) in FOX-GA went into designing and building the wargamer. The wargamer can be viewed as a coarse-grained, agent-based simulator. Each friendly COA fights 6 battles against a variety of enemy COAs to determine how each will perform under a *variety* of enemy situations. It is important to look at a variety of situations because the commander typically does not know what the enemy will do. This type of assessment allows the commander to understand each COA's overall robustness under varying circumstances. The enemy COAs are fixed (not computed by the FOX-GA) and were selected to be representative of some of the COAs which the enemy commander might choose. Each battle is modeled as a collection of minor engagements between subordinate forces in which friendly and enemy subordinate units act as agents. Agents move on a grid within the maneuver box. The vertices of that grid are the intersections between the LDTs and AAs (Figure 1). These intersections are significant because they represent the choke points in the AAs where enemy forces may set up defenses. At each time step, the moves that agents are allowed to make on the grid are determined by a set of *combat rules.* They tell each agent to either stay in place, move forward, backward, or laterally by one vertex. At the start of each battle, all friendly subordinate units are positioned in the tactical assembly areas (TAAs) (see Figure 1). Enemy forces start defensive operations from positions on LDTs 1 and 2. Given these starting conditions, the combat wargamer iterates through the following sequence until there is no more fighting in any AA or 20 time steps have passed: 1) **Examine intersections** of LDTs and AAs. Identify all intersections having both friendly and enemy units on them. These are the locations at which fire fights will occur. 2)**Compute force ratios** to assess losses. 3) **Apply combat rules** to allow all agents (units) to decide their next move and whether to change their mission status.

8 The Fitness Function

Once the wargamer has finished all six battles for a given COA, its fitness can be computed. Fitness is a function of the remaining strengths of the friendly units, the remaining strength of the enemy units, and the amount of terrain gained or lost during the battle. However, depending upon the situation commanders may want to place varying importance on each of these factors. To allow commanders to adapt the fitness function to the situation, FOX-GA allows selection from two different fitness functions, *enemy oriented* or *terrain oriented.* The enemy oriented fitness function measures success of the battle in terms of the reduction in enemy to friendly forces. The terrain oriented fitness function places importance on both the amount by which the enemy has been weakened and the amount of terrain captured or lost.

9 Performance of FOX-GA

FOX was tested on a Sun Ultra 1 workstation with 192 Megabytes of RAM. FOX evaluated 50 COAs (300 battles) per second, which is equivalent to 3000 COAs per minute. FOX-GA tends to produce good COAs after as few as 50 generations, with best performance occurring after 200-300 generations. To provide a baseline for comparison, two additional search engines, MC and MCHC, were built using FOX-GA's combat wargamer and fitness function. The two additional search engines use non-GA techniques and search strategies. **MC** searches through possible COAs using a "Monte Carlo" scheme that generates random solutions in a single generation. **MCHC** uses a "Monte Carlo with Hill Climbing" scheme.

Results. We found that both FOX-GA and MCHC substantially outperformed MC; the ten best COAs returned by MC were far lower in quality than those returned by the other two. We also found that FOX-GA's niching strategy allowed it to concentrate on the most promising niches at the expense of the less promising niches. For example, FOX-GA produced COAs in fewer niches (23 niches) than MCHC, which produced COAs in 45 niches. However, FOX-GA found better COAs in any given niche than did MCHC. In general, battlestaff prefer to be presented with large sets of diverse COAs so that they can consider different trade-offs. However, it is better, in general, to select from a smaller set of higher quality choices than it is to select from a wider choice of mediocre choices. Thus, FOX-GA provides a more appropriate set of choices to battlestaffs than does MCHC.

10 Summary and Conclusions

FOX-GA is a planning support tool for assisting military planners who operate in dynamic, uncertain, incompletely known, and risky environments. It assists them by rapidly identifying a small set of high quality plan options. In this domain, existing (manual) techniques work sufficiently in theory, but are limited in practice by the time constraints and computational computational limitations of human planners. It is difficult for human planners to re-plan as often as is desired during execution (i.e. battle) and it is difficult for them to explore as many plan options as they would like. Frequently battlestaff do not have time to explore more than a handful of COAs (and sometimes no more than 1). FOX-GA is designed to speed up the military planning and re-planning process by using rapidly generating a larger set of friendly plan options faster than they could do so themselves. Fox uses a genetic algorithm to explore a a very large plan search space, and a course-grained battle simulator to rapidly evaluate their fitness. Since it is unknown what actions the enemy will take or what battlefield conditions will exist when the plan is executed, FOX simulates the performance of the plan in battle against all the user's hypotheses about the possible plans that the enemy may follow. A "fixed" niching strategy using a set of domain-relevant niches, helps to insure that the solutions generated are not just small variants of each other. The best few COAs are presented to the user, the can then compare these options and select the ones which they judge to be the most suitable for further development.

FOX-GA offers four advantages over the typical manual methods currently used by battlestaffs. First, FOX-GA produces large numbers of distinctly different COAs. Typical battlestaffs sometimes have trouble identifying more than a handful of COAs. Second, COAs generated by FOX-GA are generated and stored and represented in the computer, eliminating the need to manually enter COAs. Third, FOX-GA provides more comprehensive wargaming data than humans can usually produce manually. A battlestaff typically requires 10-15 minutes to wargame one friendly COA against one enemy COA manually, while FOX evaluates 3000 friendly COAs per minute. Fourth, FOX-GA provides users with a very flexible way to explore COA options because of the flexibility inherent in GAs.

A major difficulty in addressing the challenges of this problem was in satisfying all of the needs simultaneously: diversity, quality and efficiency. The niching strategy used proved important in producing solutions that were both diverse and high quality. The course-grained battle simulator was important in evaluating and identifying high quality COAs, efficiently. An important issue in the design of the simulator was to develop a representation that provided sufficient accuracy without sacrificing speed. We also found the design of the human-computer interface to be of critical importance in making FOX's results accessible to the users. Early user evaluations have yielded promising results (and enthusiasm). Extensive user evaluations are scheduled for the near future.

11 Acknowledgements

This work was prepared through collaborative participation in the Advanced and Interactive Displays Consortium, sponsored by the U.S. Army Research Laboratory under the Federated Laboratory Program, Cooperative Agreement DAAL01-96-2-0003. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes. Additional thanks go to Carolyn Fiebig and Robert Winkler for all their help in this work.

References

[1] ALBM final report, LMSC F376129. Lockheed Missiles and Space Company's Final Report on the AirLand Battle Management Project.

[2] D. E. Goldberg. *Genetic Alg. in Search, Optimization, and Machine Learning.* Add. Wesley, 1989.

[3] Michael J. Kuchinski. Battle management systems control rule optimization using artificial intelligence. Technical Report MP 84-329, Naval Surface Weapons Center, 1985.

[4] William V. Porto and Lawrence J. Fogel. Evolution of intelligently interactive behaviors for simulated forces. In *Evolutionary Programming IV, 6th Internat'l Conf., EP97, Indianapolis, IN*, 1997.

[5] David E. Wilkins and Roberto V. Desimone. Applying an AI planner to military operations planning. In Monte Zweben and Mark Fox, editors, *Intelligent Scheduling.* Morgan-Kaufmann Publishing, 1994.