

Fuzzy-Genetic Decision Optimization for Optimization of Complex Stochastic Systems

Major Robert H. Kewley, Jr.
United States Military Academy,
Department of Systems Engineering,
West Point, New York, USA
fr6686@usma.edu

Mark J. Embrechts
Rensselaer Polytechnic Institute,
Decision Science and Engineering Systems,
Troy, New York, USA
embrem@rpi.edu

Abstract-Fuzzy-Genetic Decision Optimization performs multi-objective optimization on complex stochastic systems. A stochastic simulation model estimates the results of parameter settings for the system. A fuzzy ordinal preference model aggregates these results into a single fitness value for the input parameter set. A genetic algorithm uses this fitness value to search for a population of high performance parameter sets which achieve the modeler's objectives. In a tactical planning experiment, Fuzzy-Genetic Decision Optimization enabled a human planner to develop a significantly better plan than he developed without automated assistance.

1 Introduction

Fuzzy-genetic decision optimization (FGDO) solves complex problems which require concurrent optimization of multiple objective criteria. It has three modules (Figure 1). The first module is a genetic algorithm which varies the system decision variables according to the system fitness determined by the output of the fuzzy preference module. FGDO accepts input parameters from the genetic algorithm and iterates the model and fuzzy inference system until it finds a population of good solutions to the problem. The second module is a simulation model which evaluates the proposed solutions to the decision problem. The third module is the fuzzy preference module. A graphical user interface allows the user to select the important variables, their ranges, and their order of satisfaction. These selections define a fuzzy inference system which aggregates the outputs of the simulation model into one overall fitness value for that particular solution. FGDO is best suited to complex problems with two distinct features. They cannot be modeled by simple equations suitable for optimization by other methods (such as linear programming, non-linear programming, or goal programming), and they have multiple objective criteria.

2 Genetic Algorithms for Simulation Optimization

A review of simulation optimization reveals very few techniques which allow optimization of non-linear and stochastic systems. Of all the methods studied, genetic algorithms offer the added benefit of performing optimization by varying the structure and the parameters of the system, as opposed to varying the parameters only. They handle qualitative variables, and their parallel structure allows for implementation on parallel computers. For these reasons, they are well suited for recommending solutions to complex problems to a human decision-maker.

Azadivar gives a summary of available methods for simulation optimization (Azadivar, 1992). Some of these are not appropriate for complex stochastic simulations. Gradient search methods require a differentiable closed form formula for the objective as a function of the controllable inputs. Although such an objective function does not exist for complex simulations, it is possible to build and optimize a meta-model of the system. Hurrion used a neural network to replicate a complex manufacturing system simulation. He then optimized the neural network to obtain a value which approximated the optimum value in the simulation (Hurrion, 1997). In the absence of a meta-model, finite difference techniques for estimating gradient descent are not well suited to stochastic models with high variability. A good estimate of the change in the objective function given a small difference in the input requires too many iterations. Furthermore, numerical gradient descent techniques converge to local minima in the neighborhood of the initial point. Azadivar also suggests a variation of the simplex search technique which compares simulation responses at the vertices statistically. If one point is significantly worse than the others, it is dropped. If not, the algorithm requires more simulation runs. This algorithm can also prove quite expensive in the face of high variance and many decision variables. He also suggests simulated annealing. Etgar et al. successfully applied simulated annealing to a scheduling problem (Etgar et al., 1997). Azadivar concludes that more research is needed in the area of non-parametric optimization. Mockus et al. propose a bayesian heuristic

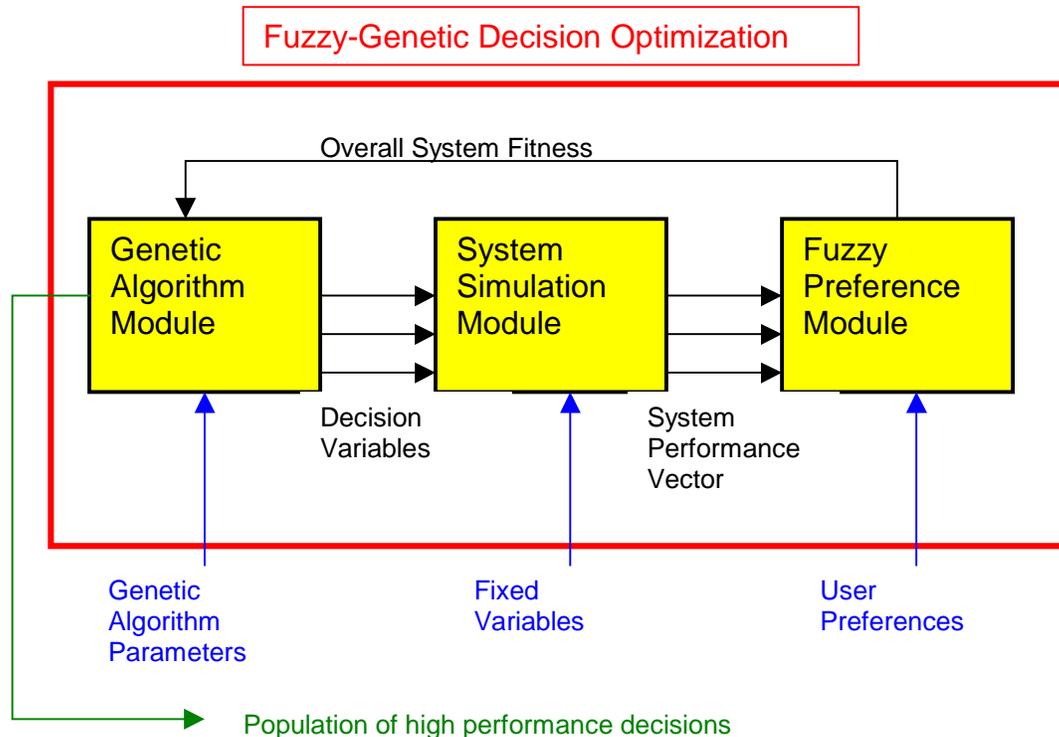


Figure 1. Block diagram for fuzzy-genetic decision optimization. The genetic algorithm module passes proposed solutions to the simulation for evaluation. The fuzzy preference module aggregates the system performance vector into an overall system fitness, which is passed back to the genetic algorithm module. This process continues until it produces a population of high performance solutions to the problem.

approach which combines domain knowledge, knowledge of previously tested solutions, and biased randomization to discrete systems (Mockus et al., 1997). This technique gives a better average performance, given a fixed number of solution evaluations, than a branch and bound technique. While these techniques show promise, they also have some deficiencies. Many complex systems require optimization of their structure, which allows the value, type, and number of inputs variables to vary. Population based evolutionary or genetic algorithms can handle this type of optimization.

Pierreval and Tautou propose evolutionary algorithms as a solution to the problems discussed above (Pierreval and Tautou, 1997). They address the problems of high computation cost, convergence to local extrema, inability to handle qualitative variables, and ease of implementation. Evolutionary algorithms allow for mutation schemes which govern the selection of a new qualitative variable from the list of possibilities. The probabilistic exploration of the solution space by a population as opposed to a single member makes them less sensitive to local extrema as compared to other methods. An evolutionary algorithm is simpler to implement than solution techniques which require extensive knowledge of statistics and mathematics. Finally, they argue that evolutionary algorithms are better able to

handle noisy observations. Most methods move in the solution space from one point to the next based upon estimates of the objective function value at each point. If the observations are noisy, movements in the wrong direction hinder the movement to the optimal point. The stochastic nature of an evolutionary algorithm bases selection on giving the most fit the best chance for survival. Noisy observations may briefly give a temporary distortion of a particular solution's fitness. However, over the course of the entire algorithm, the better solutions will exist in the final population with higher probability than the poorer solutions. In order to experimentally support their conclusions, they used an evolutionary algorithm to estimate the optimal design of a simple manufacturing system with both qualitative and quantitative variables. They tested for a deterministic and stochastic objective function and compared the evolutionary algorithm's performance against the result of a near-exhaustive search. In both cases, the evolutionary algorithm converged to the neighborhood of the point found by the near exhaustive search, and they found no statistically significant difference in the objective function of the two points. They conclude that the evolutionary algorithm is a very flexible and efficient

method for simulation optimization. It has the additional feature of being very easy to parallelize.

3 Stochastic Simulation Models of Complex Systems

A simulation is one of the most powerful tools used in the design and evaluation of complex systems (Pedgen et al., 1995). Many of the systems in the real world are so complex that mathematical models are intractable (Ecker and Kupferschmid, 1988). One alternative is to do experimentation with the real system itself. One can easily see how experimenting with critical systems such as traffic control, communications systems, or local economies is inconvenient, expensive, and time consuming. In other cases, such as system design, the system of interest may not even exist. In these instances, it may be possible to simulate the system of interest (Giordano and Weir, 1985). This technique allows us to investigate new policies, rules, and designs without disturbing the real system. We can visualize system performance. We can control and stop time and view data summaries to gain new insights into the critical variables and their effects on the system. There are some disadvantages to simulation. Model building requires specialized training, and the model quality depends on the modeler's skill. Results are stochastic and often difficult to interpret. The analysis can often be expensive, time consuming, and not worth the effort. Finally, simulation models are input-output models which yield the output of the system for a given input. These models are "run" rather than "solved." They do not generate an optimal solution. It is up to the analyst to develop and evaluate different alternatives in search of better performance (Pedgen et al., 1995). Since most complex systems have many different outputs, a clear notion of a "better" solution escapes the analysis.

Advancing technology alleviates many of the disadvantages of simulation modeling. Many commercial simulation packages allow graphical and intuitive development of simulation models without as much specialized training. Their visual and statistical capabilities allow for easier analysis and interpretation of these models. Furthermore, increasing hardware speed and availability allows for larger run sets. These developments set the conditions for the success of fuzzy-genetic decision optimization. A genetic algorithm may now execute the large number of runs needed to converge to an optimum. The fuzzy preference model allows the user to aggregate simulation outputs to define a clear notion of better system performance. This notion serves as a road map to guide the search for better solutions. A combination of these software, hardware, visual, and algorithmic developments has a synergistic effect which yields a powerful tool for finding better policies, configurations, and designs of complex systems.

4 Fuzzy Preference Module

Many problems confronted by decision-makers require aggregation of preference over a large number of attributes. The military tactical decision used in this experiment is a representative example. If a military commander accepts a mission to eliminate enemy resistance in an objective area, his mission accomplishment may be measured by the number of enemy vehicles and the number of enemy personnel remaining on the objective. These, however, are not his only considerations. He also wants to preserve his soldiers and his vehicles. Furthermore, follow-on operations may necessitate conservation of fuel and ammunition. Definite conflicts exist in the objectives. A course of action which completely eliminates all enemy from the objective will likely result in higher friendly casualties. It will also consume more resources. How does the commander define his preference over these conflicting alternatives so that he may devise a plan which seeks to maximize that preference? The selected preference model should meet a few criteria.

- The preference model, once formulated, must accurately reflect the decision maker's preference of one alternative to the other.

- The model must not be so complex or cumbersome that it will not be used. Too much complexity or a lack of understanding will cause the user to discard the model in favor of his own intuition.

- Finally, it must be flexible. Too many unrealistic simplifying assumptions will limit the model's applicability and accuracy in real decision situations.

This paper uses an implementation of ordinal preference using a fuzzy inference system to meet these needs.

Existing preference models such as additive utility (Clemen, 1996), out-ranking methods (Roy and Vincke, 1984), and ordinal preference (Beroggi and Wallace, 1997) have one of two faults. They require simplifying assumptions such as preferential independence or strict ordinal preference which are often violated in real decision situations. Or they are so complex that it is difficult to extract the appropriate data from the decision-maker. This leads to poor understanding of the results and little confidence in the model. A fuzzy implementation of the ordinal preference model (Beroggi and Wallace, 1997) overcomes these faults. In this model, the user, given a certain satisfaction level for the criteria, defines what improvement to seek next in building a preference from the worst conceivable case to the most optimistic. This is relatively easy to elicit, and it allows for preferential dependence among the criteria.

Consider a problem where a buyer must purchase a car. He or she has three important criteria: gas mileage, acceleration, and cost. The worst case is a car that gets 10 miles per gallon, takes 15 seconds to get to 60 miles per hour, and costs \$30,000. The best case is a car that gets 50 miles per gallon, takes 5 seconds to accelerate to 60 miles

per hour, and costs \$10,000. Note that the best and worst cases are not realistic alternatives. They simply define the range of possible outcomes for each of the criteria. Tell the decision-maker that he or she must currently accept the worst possible alternative. However, he may gradually improve the outcome item by item. The preference level for the worst alternative is defined as “very low.” The decision-maker will first improve alternatives one by one until preference is medium. Given the worst alternative, suppose he would choose to improve gas mileage from 10 to 20 miles per gallon. If that could be done, he would then like to improve acceleration from 15 to 10 seconds. Next, he would like to reduce cost from \$30,000 to \$15,000. At this point, he defines preference as medium. This is the decision-maker’s own idea of an “average” alternative. If all of the above improvements could be realized, the decision-maker would continue to build to a high level of preference. He would first seek to improve gas mileage from 20 to 50 miles per gallon. Then he would reduce cost from \$15,000 to \$10,000. Finally, if all of the above could be achieved, he would improve acceleration from 10 to 5 seconds. The decision-maker has now completed his preference, and the preference of this best possible alternative is defined as “very high.” Note that the decision-maker does not know before the fact how successful the search for alternatives will be. The decision-maker’s preference order guides the search under the assumption that we are not going to be able to find the “best” alternative. However, any alternative we find will be ordered according to how it satisfies the decision-maker’s preferences. It must satisfy each lower preference before it can be given credit for satisfying higher preferences. A very inexpensive (\$15,000) and fast (5 seconds 0 to 60 miles per hour) gas guzzler (10 miles per gallon) would get no credit at all because it did not satisfy the initial preference improvement, to improve gas mileage from 10 miles per gallon to 20.

The preference order defines a rule set that uses criteria values to score any alternative. This scoring system allows us the completely order all alternatives from worst to best. Assume preference ranges on a scale from 0 to 1 where 0 is “very low”, 0.5 is “medium”, and 1 is “very high.” In the auto preference example, there are three rules prior to medium preference and three rules after. Therefore, improvement in preference from 0 (very low) to 0.5 (medium) is divided into three equal intervals, each of which corresponds to the first three criteria improvements in the preference order (See Figure 2). Similarly, improvement in

preference from 0.5 (medium) to 1 (very high) is divided into 3 equal intervals which correspond to the last 3 criteria improvements in the preference order.

The following rules result from the automobile example:

- If gas mileage < 20 mpg, preference = $0 + 0.16 * (\text{gas mileage} - 10 \text{ mpg}) / (20 \text{ mpg} - 10 \text{ mpg})$.
- If gas mileage > 20 mpg and accel > 10 sec, preference = $0.16 + 0.16 * (15 \text{ sec} - \text{accel}) / (15 \text{ sec} - 10 \text{ sec})$
- If gas mileage > 20 mpg and accel < 10 sec and cost > 15k, preference = $0.33 + 0.16 * (30k - \text{cost}) / (30k - 15k)$
- If gas mileage > 20 mpg and accel < 10 sec and cost < 15k and gas mileage < 50 mpg, preference = $0.5 + 0.16 * (\text{gas mileage} - 20 \text{ mpg}) / (50 \text{ mpg} - 20 \text{ mpg})$
- If gas mileage > 50 mpg and accel < 10 sec and cost < 15k and cost > 10k, Preference = $0.67 + 0.16 * (15k - \text{cost}) / (15k - 10k)$
- If gas mileage > 50 mpg and accel < 10 sec and cost < 10k and accel > 5 sec, Preference = $0.84 + 0.16 * (10 \text{ sec} - \text{accel}) / (10 \text{ sec} - 5 \text{ sec})$

Fuzzy variables allow this system to approximate human reasoning. In reality, as we improved gas mileage from 10 mpg to 20 mpg, we would start to gradually consider improvements in acceleration as we approached 20 mpg. After we reached 20 mpg, we would gradually start to lose interest in gas mileage improvements. Fuzzy variables and their linguistic equivalents allow more than one of the rules to apply simultaneously. Consider the fuzzy variables for gas mileage in Figure 3. Similar fuzzy variables are developed for acceleration and cost. Substituting fuzzy variables for crisp values in the rule set, we get the following rules.

- If gas mileage < low, preference = $0 + 0.16 * (\text{gas mileage} - 10 \text{ mpg}) / (20 \text{ mpg} - 10 \text{ mpg})$.
- If gas mileage >= low and accel > med, preference = $0.16 + 0.16 * (15 \text{ sec} - \text{accel}) / (15 \text{ sec} - 10 \text{ sec})$
- If gas mileage >= low and accel <= med and cost > low, preference = $0.33 + 0.16 * (30k - \text{cost}) / (30k - 15k)$
- If gas mileage >= low and accel <= med and cost <= low and gas mileage < very high, preference = $0.5 + 0.16 * (\text{gas mileage} - 20 \text{ mpg}) / (50 \text{ mpg} - 20 \text{ mpg})$
- If gas mileage >= very high and accel <= med and cost <= low and cost > very low, preference = $0.67 + 0.16 * (15k - \text{cost}) / (15k - 10k)$
- If gas mileage >= very high and accel <= med and cost <= very low and accel > very low, preference = $0.84 + 0.16 * (10 \text{ sec} - \text{accel}) / (10 \text{ sec} - 5 \text{ sec})$

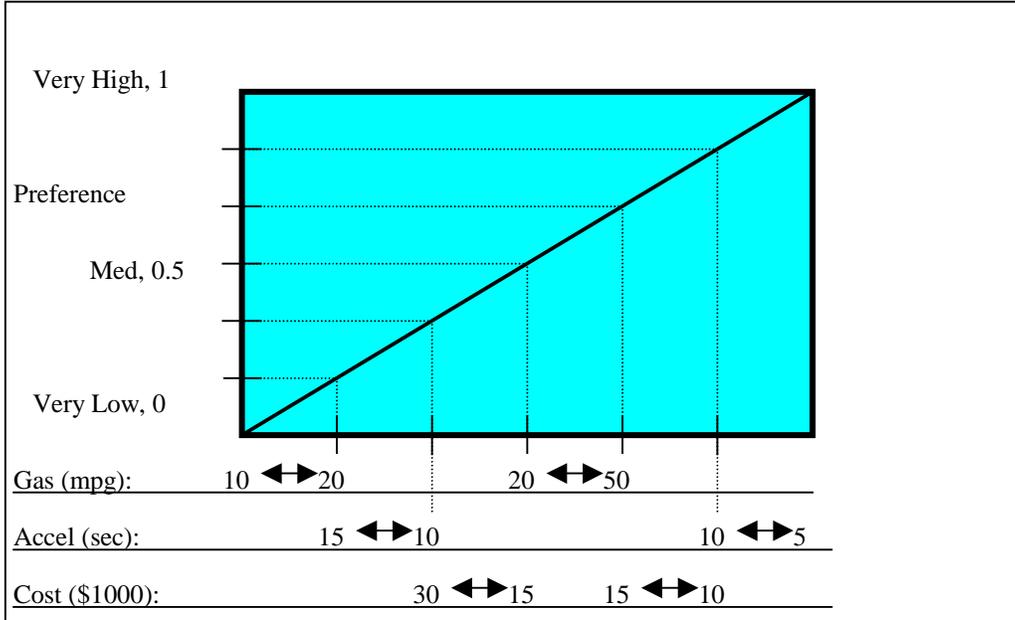


Figure 2. Ordinal preference for an automobile purchase. Improvement in gas mileage from 10 to 20 mpg represents an increase in preference from 0 to 0.16. If gas mileage is better than 20 mpg, then improvement in acceleration from 15 seconds to 10 seconds represents an increase in preference from 0.16 to 0.33. And so forth....

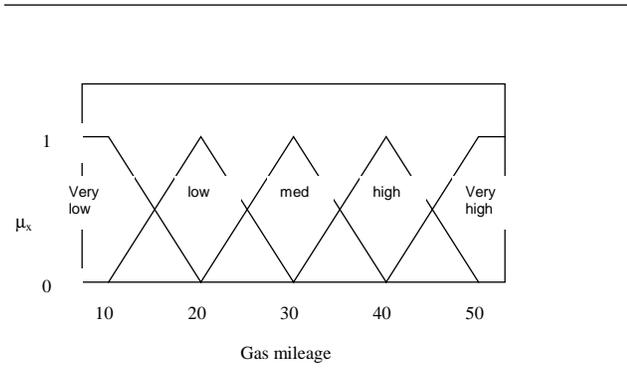


Figure 3. Five fuzzy variables defined for gas mileage. μ_x represents the membership of each value of gas mileage in each of the five variables very low, low, med, high, and very high. For example, 20 miles per gallon has a membership of 1 in the variable low, and 15 miles per gallon has a membership 0.5 in the variable low.

The above rule set represents a Sugeno fuzzy inference system. In general, the output of a Sugeno inference system may be represented by the formula:

$$y = \frac{\sum_{i=1}^m \lambda_i f_i(\bar{X})}{\sum_{i=1}^m \lambda_i}$$

where rule i is one of m rules. λ_i is the degree of applicability of rule i . $f_i(\bar{X})$ is the consequent of rule i , a linear function of the inputs to the system \bar{X} .

$$\lambda_i = \bigwedge_{j=1}^n (\mu_{x_{ij}}(x_j))$$

where $\mu_{x_{ij}}(x_j)$ is the membership of input j in the antecedent of rule i (Jang et al., 1997).

Consider an alternative automobile which gets 18 miles per gallon, accelerates to 60 in 13 seconds, and costs \$20,000. In the first rule, 18 miles per gallon coincides with $< \text{low}$ with a membership of 0.4 ($\lambda_1 = 0.4$). The consequent of rule 1 yields:

$$\begin{aligned} \text{preference} &= 0 + 0.16 * (\text{gas mileage} - 10 \text{ mpg}) / (20 \text{ mpg} - 10 \text{ mpg}) \\ \text{preference} &= 0 + 0.16 * (18 - 10) / (20 - 10) = .128 \end{aligned}$$

In the second rule, gas mileage of 18 is \geq low to with a membership of 0.6. Acceleration is $>$ med with a membership of 1. The minimum of 0.6 and 1 yields 0.6 ($\lambda_2 = 0.6$). The consequent of rule 2 yields:

```

preference = 0.16 + 0.16 * (15 sec - accel) /
(15 sec - 10 sec)
preference = 0.16 + 0.16 * (15 - 13) / (15 -
10) = .224

overall preference = (0.4 * .128 + 0.6 * .224)
/ (0.4 + 0.6) = .186

```

In this example, even though the alternative did not completely satisfy the requirement to achieve low gas mileage prior to considering acceleration, the fuzzy rules allowed near satisfaction of that requirement to phase in added preference for acceleration in rule 2. Wider trapezoidal membership functions for the fuzzy variables would allow even more rules to be considered simultaneously, allowing more trade-offs between criteria.

This preference model approximates the decision-maker's preference, does not require preferential independence, and is comparatively easy to elicit from the user. For these reasons, it will be used to aggregate the simulation outputs in fuzzy-genetic decision optimization in order to get a single fitness value for each simulated alternative.

5 Tactical Planning Experiment

Fuzzy Genetic Decision Optimization performed well in a military tactical planning experiment. The problem was to select attack routes for each unit. The plan as a whole should perform well against a fixed enemy. The experiment used two treatments. In the first treatment, a human planner developed a course of action unaided by FGDO. In the second treatment, the human planner viewed the results of FGDO to get ideas to be used in his final plan.

5.1 Entering Preferences

The commander entered his preference order using a graphical user interface. These preferences defined a fuzzy inference system for preference using the procedure described in section 4. Given values for enemy vehicles destroyed, friendly vehicles destroyed, and friendly vehicles on the objective, the fuzzy inference system determines a preference value from 0 (least preferred) to 1 (most preferred) for the battle outcome. This value is used as a fitness value for the friendly plan evaluated in the simulation.

5.2 Genetic Algorithm Search for Possible Courses of Action

A genetic algorithm search for possible solutions to this tactical problem used the following parameters:

```

Number of solutions generated with each
generation: 40
Percentage of the population replaced with each
generation: 60
Number of generations: 40
Probability of crossover: 0.6
Probability of mutation: 0.1

```

The system ran for approximately seven hours to complete 40 generations. The fitness evolution for the population is shown in Figure 4.

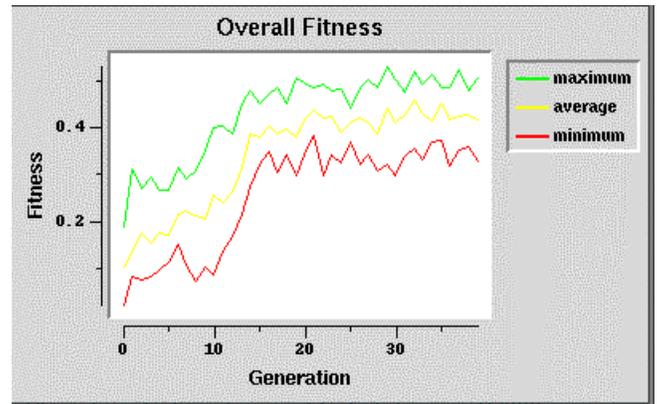


Figure 4. Fitness evolution for population of possible attack plans. The top line represents the fitness of the best plan for each generation. The middle line represents the average fitness while bottom line is the worst. After 20 generations, there was no discernable improvement in the plans generated.

5.3 Course of Action Comparison

A summary of the simulation results generated by the final automated COA and the human COA is shown in Figure 5. The automated course of action significantly outperformed (99.9% confidence level) the human-developed course of action for both friendly casualties and enemy casualties. It reduced friendly casualties from 22% to 15% while increasing enemy kills from 46% to 56%.

6 Conclusion

In this experiment, fuzzy-genetic decision optimization was able to aid the human planner in developing a course of action which was significantly better with respect to the decision-maker's own preferences. While the 7 hour computation time is currently unacceptable, future improvements to computer hardware, improvements in the simulation efficiency, and perhaps a parallel implementation of the genetic algorithm can all combine to reduce this time to a period which fits in the decision cycle for this problem. These results encourage continued research to further develop and refine Fuzzy-Genetic Decision Optimization.

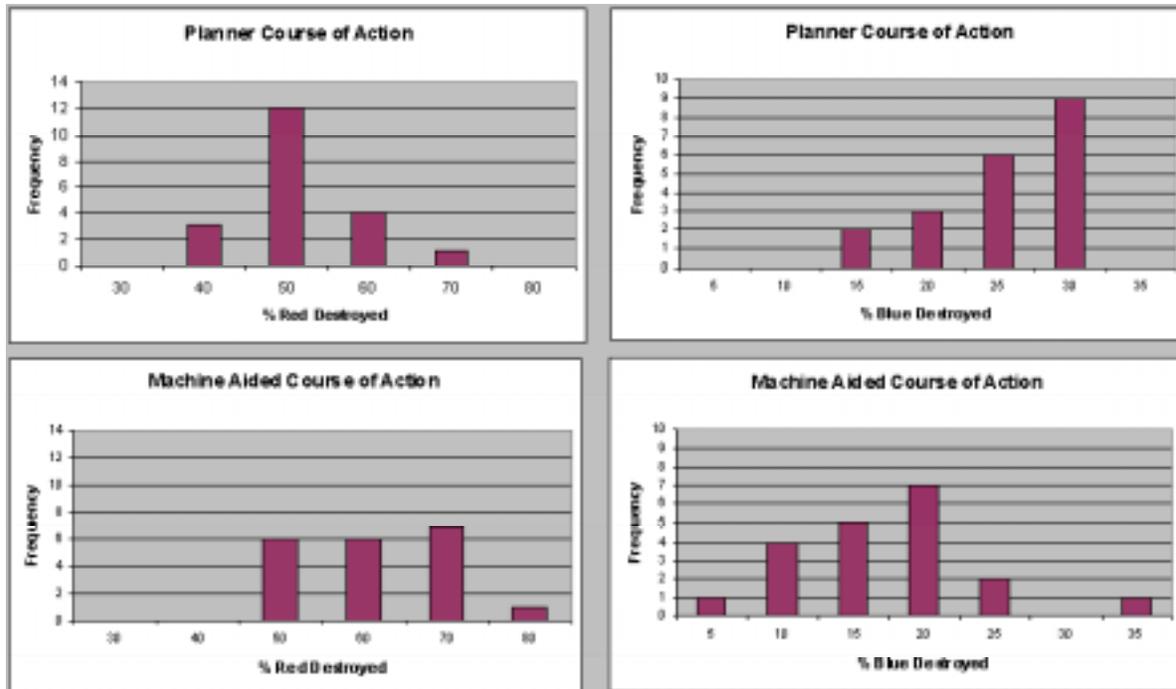


Figure 5. Performance comparison of human developed (top) and machine aided (bottom) courses of action. The machine aided course of action destroyed significantly more enemy (red) forces while losing significantly fewer friendly (blue) forces. It achieved this performance in spite of the complex, multi-objective and stochastic nature of the system.

Bibliography

- Farhad Azadivar. A tutorial on simulation optimization. In James J. Swain, David Goldsman, Robert C. Crain, and James R. Wilson, editors, *Proceeding of the 1992 Winter Simulation Conference*, pages 198-204. Winter Simulation Conference Board of Directors, 1992.
- Giampiero E. G. Beroggi and William A. Wallace. The effect of reasoning logics on real-time decision making. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 27(6):743-749, 1997.
- Robert T. Clemen. *Making Hard Decisions: An Introduction to Decision Analysis*. Duxbury Press, second edition, 1996.
- Joseph G. Ecker and Michael Kupferschmid. *Introduction to Operations Research*. Wiley, New York, 1988.
- Ran Etgar, Avraham Shtub, and Larry J. LeBlanc. Scheduling projects to maximize net present value – the case of time-dependent, contingent, cash flows. *European Journal of Operational Research*, 96(1): 90-96, 1997.
- Frank R. Giordano and Maurice D. Weir. *A First Course in Mathematical Modeling*. Brooks/Cole, Monterey, 1985.
- R.D. Hurrion. An example of simulation optimisation using a neural network metamodel: finding the optimum number of kanbans in a manufacturing system. *Journal of the Operational Research Society*, 48(11):1105-1112, 1997.
- J.-S. R. Jang, C.-T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Prentice Hall, Upper Saddle River, 1997.
- Jonas Mockus, William Eddy, Audris Mockus, Linas Mockus, and Gintaras Reklatis. *Bayesian Heuristic*

Approach to Discrete and Global Optimization.
Kluwer Academic Publishers, Dordrecht, The
Netherlands, 1997.

C. Dennis Peden, Robert E. Shannon, and Randall P.
Sadowski. *Introduction to Simulation Using SIMAN.*
McGraw-Hill, 1995.

Henri Pierrevel and Laure Tautou. Using evolutionary
algorithms and simulation for the optimization of
manufacturing systems. *IIE Transactions*, 29(3):181-
189, 1997.

B. Roy and Ph. Vincke. Relational systems of preference
with one or more pseudo-criteria: some new concepts
and results. *Management Science*, 30(11):1323-1335,
1984.