Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2007)

Next Generation End-To-End Logistics Decision Support Tools. Evolutionary Logistics Planning

Beth DePass
BBN Technologies
10 Moulton Street
Cambridge, MA 02138 USA
Email: bdepass@bbn.com
Phone: (617)873-4289 Fax: (617)873-2794

Abstract - *Logistics planning and decision support systems have traditionally focused on planning large scale military operations with limited forecasting and execution tools causing many military logistics support tools to fall short of providing a true end-to-end solution. A true end-to-end solution will yield a system that can be used for logistics training, long-term logistics planning operations, real-time logistics planning and execution during an operation, and real-time decision support for immediate replanning and response to ongoing operations for all echelons of a military hierarchy. In this paper we will explore technologies that will provide flexible and accurate plan development leading to better plans, increased decision support, and ultimately better execution of military logistic plans.*

*Advanced logistics planning and forecasting tools built by DARPA projects such as the Advanced Logistics Program (ALP), Ultra*Log [2], and Network Centric Logistics (NCL) successfully implemented capabilities that provide portions of an end-to-end logistics solution. These systems were built using the Cognitive Agent Architecture (COUGAAR) [1] which provides support for large multi-agent systems that require distributed processing and allow for numerous applications and technologies to be seamlessly integrated into large scale logistics systems. In order to provide the next generation of forecasting and execution utilities that will lead to an end-to-end solution, large multi- agent systems will need to incorporate technologies that provide the following attributes: technologies that isolate and focus on specific areas of a plan, technologies that provide greater flexibility in planning and technologies that will provide a mechanism for human interactions.*

Under the solutions section of this paper four technical solution areas are discussed:

- *Optimized distribution*
- *Evolutionary planning*
- *Focused forecasting*
- *Execution and simulation.*

Existing and new techniques in these areas will provide the necessary logistics planning attributes for the next generation of logistics decision support systems.

INTRODUCTION

Traditional multi-agent logistics decision support systems fall short in the following three areas: *(1) existing systems have too many agent to agent and computing resource dependencies*, where agent is defined as a computing or optimization process that represents a particular entity such as a consumer, supplier, transporter, etc. *(2) existing systems are not flexible* as requirements (situational or user defined) change, and (3*) existing systems cannot respond quickly enough* to support the fast paced environment surrounding current and future (immediate/near term) missions. Because the planning environment has changed, it has become necessary to revamp our systems in order to meet the military planner's decision support needs.

TRADITIONAL SYSTEM SHORTFALLS

Agent-to-Agent and Computing Resource Dependencies

Application dependencies between agents as well as dependencies on computing resources continue to be a challenge in large systems (agent or other) that develop complex in-depth large scale military logistics plans. These dependencies can limit the timeliness of solutions, the robustness of the system under stress, and the overall flexibility of the logistics plan. Developing adaptive software applications that can continue planning with limited resources proved to be an important step towards timely and robust plan development within the Ultra*Log program. The Ultra*Log Adaptive Logistics application explored possible solutions to these problems including predictors and multi-resolutional forecasting. As resources were available (agents, computing or bandwidth), the application built more details of the plan. When resources were limited, the application reduced the fidelity of the plan and focused on the near term aspects. If coordinating agents such as a customer or a supplier were not available to provide a definitive answer, predictors were temporarily used in their place to allow low fidelity planning to continue.

This approach provided the user with the ability to quickly review low fidelity planning solutions long before the entire plan was completed. While this was a very good solution to providing the user with timely results, it is evident that this is only a partial solution with respect to an end-to-end planning system.

Existing Systems Are Not Flexible

Flexibility is a key component of today's and future decision support systems. Future systems must be able to react immediately to events, perturbations and even human redirection while producing enough information for the user to make decisions in minutes. The system can learn about past planning experiences and employ knowledge bases. Also, the system can develop a huge number of What-If branches to help determine what might happen next and what might be the best course of action given the current state. Further, the system can employ genetic algorithms to determine the most optimal plan. Other techniques may also allow for dynamic system redirection or replanning in response to plan changes. However, the only truly flexible system that will generate the best solutions in a very dynamic planning environment is one that is able to gather some sense of reality from a human. If the human element can be placed in the computing loop in addition to the other techniques mentioned, then and only then will we begin to grow a state of the art decision support tool.

Existing Systems Cannot Respond Quickly Enough

Existing multi-agent systems can provide a detailed plan in about an hour for a large scale military operation spanning 180 days. While this was a significant improvement provided to the users from the ALP and Ultra*Log programs, the military operations of today and the future are much more fast paced and require decisions to be made by the operations and logistics personnel in minutes instead of hours or days. In order to meet today's planning demands, which includes logistics planning and execution at all military echelons, decision support tools must provide flexibility in planning solutions as well as continued improvements in the area of timeliness, logistics forecasting and execution.

In today's military environment, developing plans for 180 day operations that involve entire brigades of soldiers are not as common as developing plans for battalions or platoons that are planning for high intensity operations that may only last a couple of hours. Additionally, traditional logistics planning systems have not focused on brigade and below logistics problems and solutions. Planning and logistics support decisions must be made much more quickly than the current large multi agent systems are capable of delivering especially at the lower level echelons. The military personnel and the system must think and react immediately. Iterations between the system and the logistics planner are necessary more than ever to help direct the system in extremely dynamic environments in order to effectively reduce solution times.

Another consideration is that high intensity battles are often followed immediately by a peacekeeping mission. A major challenge for next generation logistics planners is keeping pace with mission evolution as traditional force-on-force warfighting evolves into occupation and peacekeeping. Not only does the commodity mix change, but the timing and pacing requirements for in-theater supply and distribution also change. For example, the critical logistics demand and associated challenges may change from tons of ammunition to tons of food in a matter of hours. Peacekeeping operations may exceed 180 days or high intensity battles may occur intermittently throughout the peacekeeping mission.

The dynamic nature of the operation requires constant adjustment and refinement of the logistics distribution and resupply plan. Planners must be able to weigh their options more quickly and efficiently than ever before as we move towards the Future Combat System warfighting model and other logistics supply chain paradigm shifts.

In summary three major shortfall areas of traditional multi-agent decision support systems must be improved in order to address the needs of the modern military logistics planner. Decision support systems of the future must employ applications that have reduced dependencies on computational resources as well dependencies on other agents or computational entities within the planning system. The systems must also be flexible to an ever changing logistics environment, while responding quickly to support the fast paced environment surrounding current and future missions.

The Next Generation Solution

The next section will explore potential solutions to the shortfall areas discussed above. These solutions include extensions to existing technologies as well as the integration of new techniques. Four areas of research that will be explored are: ***optimized distribution; evolutionary planning; focused forecasting; plan execution and simulation***. Future systems must not be built from scratch, but rather built on technologies that have moved logistics decision support tools into the 21st century. Many existing applications and technologies can be easily integrated and extended for use in future systems thanks to agent architectures like Cougaar and languages such as Java.

Optimized Distribution

Introduction and Previous Research

Distribution optimization work done in Ultra*Log, NCL and other BBN/DARPA projects have historically employed genetic algorithms. In Ultra*Log the distribution plan was built using the expected logistics demand profile sent to the distribution agents by the agents representing the logistics customers. Most recently, the NCL program sought to assess approaches to supply distribution solutions in the face of uncertainty. The optimization challenge was to explore optimizing supply distribution for not only the expected (projected) demand profile, but also for worst case scenario demand profiles. The NCL CAS (Network Centric Logistics: Complex Adaptive Systems) optimization approach imagined various forces acting on a supply truck. Some forces were defined as ***reactive***, responding to currently observed inventory and demand while others were defined as ***anticipatory***, seeking to anticipate unknown future demand and capabilities. The study scenarios created forces which caused distribution platforms to be attracted to or repelled from consumer platforms. Some examples of such forces include: customer platforms that were low on supplies that attracted distribution vehicles, distribution vehicles containing similar supplies repelled each other, and distribution vehicles that were attracted to a space where customers were expected to be in the near future (according to the plan). The CAS approach was able to bound the distribution problem space by limiting the forces to localized areas such that trucks emitted pressures only within a localized area and reasoned about pressures they felt only in the immediate (local) area. Program results suggest that combining reactive and anticipatory forces allow supply trucks to strike a balance between satisfying immediate needs and being prepared for uncertain future situations. Additionally, the CAS approach was able to generate extremely fast, reasonable simulation results (~ 1second per simulation).

The system did not utilize any global coordinator or global scheduling mechanism. The implementation relied on the ability to tune a set of parameters to optimize overall performance and then employed a genetic algorithm to find tuned sets of parameters for various uncertain environments. This is a good example of an approach that produces timely results and has limited dependencies on other compute parties (agents).

Results

Functional results of the NCL experiments showed that when compared to traditional logistics distribution systems, the NCL capabilities did result in a militarily significant increase in operational availability and reduction in customer wait time, particularly under conditions of dynamic change and uncertainty. It was concluded that the NCL algorithms provided significant warfighting benefit. Warfighting benefit (military utility) was assessed in terms of increased operational availability (Ao) and reduced customer wait time (CWT).

In the unstressed scenario, the average NCL Ao for the Brigade was 95% vs. 86% Ao for the baseline system. For example, the Cav Squadron Ao was 91% in the NCL case vs. 83% Ao in the baseline case. Average CWT for the Brigade overall was 7 hours (NCL) vs. 52 hours (baseline).

In the fully stressed case, NCL was shown to be far superior. Ao for the Brigade was 93% (NCL) vs. 83% (baseline). Most dramatic was the performance during the highest stress period from day 24 to day 28. The overall Brigade baseline Ao dropped to 57% on day 27 compared to the NCL Ao of 78%. CWT for the fully-stressed Brigade averaged 8 hours (NCL) vs. 106 hours (baseline).

Extensions and New Research Area

A more comprehensive, next generation solution involves extending the NCL approach (based on the attraction of support trucks in the logistics arena) so that the logistics supply chain paradigm can be changed to allow for greater flexibility of the distribution assets (trucks) and supplies. Strict supply chains would no longer be necessary and extraneous supply points would be non-existent. Without customer-supplier dependencies on specific agent assets and supply chains, distribution plans can be more easily tailored and changed to meet dynamic resupply requirements. This solution allows distribution planning to continue when agents and assets are unavailable due to bandwidth limitations (or other resource limitations). This solution further addresses dependency issues and allows additional flexibility in the logistics plans.

Optimizations may also be extended if simulations of the enemy forces are available. Rules regarding attraction of enemy forces may affect where resupply occurs, where combat platforms are needed to protect logistics assets, or when combat platforms should engage in battles.

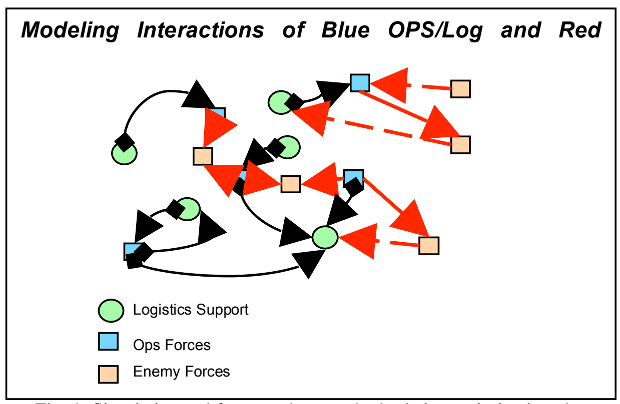

Fig. 1. Simulating red forces enhances the logistics optimizations by allowing for additional attraction forces to be considered while generating the best solution.

In the current systems, movements to provide logistics force protection or changes in resupply locations are often not included in the logistics demand footprint. Modeling these types of movements and changes in the plan will provide a better logistic demand forecast and better distribution plans. Complete logistics plans will move logistics planning systems another step closer to modeling what might actually occur in real time during a military operation.

Evolutionary Planning

Introduction

The idea of evolutionary planning will incorporate previously researched topics such as Planning Horizons and What-If branches and will also incorporate the use of human-machine interactions at defined points within the system and the logistics plan. When we think of evolutionary planning there are two specific realities to consider. The first reality is that as the execution of future plans draw near more details are needed to execute that plan. The second reality is that military situations change rapidly and even immediate actions must often be replanned. Planning horizons deal with these two realities explicitly.

Previous Research

Previous research in the area of Planning Horizons focused on the use of time horizons and plan fidelity horizons. These techniques proved to be useful in reducing overall planning time and generating low fidelity plans that could be used in place of back of the envelope estimates performed by hand by military planners. In Ultra*Log, time horizons were used as tools to gradually build a logistics plan as resources and time allowed. The horizons are by definition chunks of the plan in terms of time. For example, now thru the next 7 days might be defined as the first horizon and days 120-180 of the plan might be the last horizon, with many or few windows representing the parts of the plan that fall between day 7 and 120. The first horizon was given priority to generate a high fidelity logistics plan. The next horizon was given secondary priority to develop a low fidelity logistics plan. As computing resources allowed, the lower priority horizons could be reworked in high fidelity. Computing priority could continue falling down the chain of horizons until planning was done in the most detail possible.

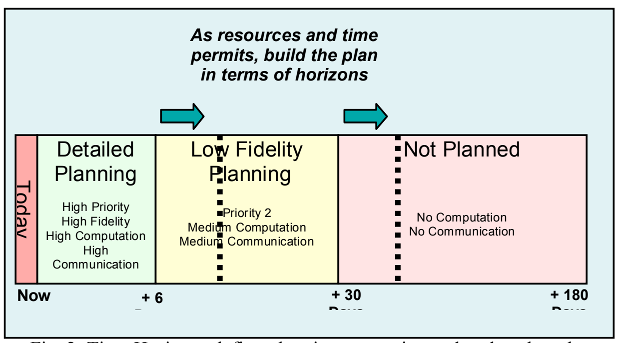

Fig. 2. Time Horizons define planning constraints to be placed on the system. Note how the horizons are maintained as the plan is executed.

Results and Lessons Learned

The 2003 Military Utility Assessment for the Ultra*Log project confirmed adequacy, accuracy, completeness and timeliness of planning, replanning and user interface availability while the system is under attack. The timeliness and availability of information was directly due to employing the time horizons technique.

One of the many lessons learned from the Ultra*log project and others is that while high fidelity 180 day plans are nice to have, in reality users need and care more about the details of what is going to happen in the immediate future. Therefore, planning horizons are very useful to logistics planners.

Estimates about how the entire plan will play out are important at a high level, but in the real world, one change in the near future can cause a huge ripple effect on the rest of the plan. Tools can be leveraged to reduce the stochastic nature of ripple effects caused by small changes in a plan, and in some instances they can attempt to smooth over the effects. However, the plan will always change as time marches forward and no amount of planning can prevent changes to the plan.

Extensions and New Research Areas

Extensions to the idea of time horizons for next generation systems will use horizons to define when it is the most beneficial to run genetic algorithms to reduce risk. They will also use horizons to define when it makes the most sense to perform what-if branch scenarios for the user. Bounding genetic algorithm searches and What-If scenarios reduce unnecessary computing and save time when developing options for the plan that will never be used.

What-If horizons could be defined to set boundaries regarding what time frame or what parts of the plan the What-If branch should explore. For example, a user may determine that it would be very useful to perform some What-If branches that focus on a 2 week period of the plan. In addition, the user could further define the What-If to focus specifically on a class of supply. At this point the system can begin creating What-If branches. Also at this time, the user can begin seeding the What-If engine with specific events that are likely to occur during the 2 week time period. As the system processes it may collect events or conditions from users or from real world inputs. The events could then be weighted or chosen at random. The What-If mechanism would then use these events to supply the user with multiple branches.

There are many ways to add flexibility and human interaction to a What-If architecture. However, What-If branches should be bound by quantity (e.g. 3 branches) for most planning cases in order to reduce excessive compute resource usage and excessive branches. There is a point at which too many What-Ifs do not provide the user with useful information, either because they depart too far from what is likely to happen or they provide the user with too many options to look through in a reasonable amount of time. Some What-If branches may not provide paths that are significantly different from each other; these cases are clearly wasteful in terms of time and resources spent producing the branch and reviewing the branch.

In order for What-Ifs to reach their potential, they must be directed by the user to some degree, bounded in order to avoid branches that are very similar and provide little additional information, and they must be flexible and allow the plan unravel in time and react as necessary. If the What-If branch is too scripted or the What-If branch can not take into account events, it will not be as useful to the user. The What-If engine must allow events that are occurring in real-time to contribute to the overall plan "picture" and it must also allow events or perturbations that occur as the What-If unfolds to change its direction. Finally, the What-If engine must implement a high level plan validation mechanism that provides the user with a logistics feasibility overview as well as a mechanism that promotes cohesive integration with a simulator for detailed validation.

Another area in which horizons might be useful is to define the place in the plan in which to perform What-If experiments. By place in the plan, I mean agents in the plan that may represent certain units or soldiers that could benefit from replanning or planning options. For example, unexpected enemy fire may have just broken out around these units and more ammunition is required. Or perhaps a bridge has unexpectedly been blown up. The forces will be rerouted, causing a change in their near term logistics needs (e.g. fuel). A logistics planner may wish to explore multiple options that focus on in-depth plans to support these agents within the next hour. If replanning and What-If tools can be bound to produce detailed (high fidelity) options for these agents (in this case units), the user can make the necessary support decisions in minutes.

Conclusions

In general, smaller problem areas should require less time to generate a multitude of options. What-If branches traditionally have been a very hard problem to solve especially for large scale operations. The branches are very often as large as or larger than the original plan and they can quickly consume resources and grow to very large entities in themselves. Evaluating, understanding and even switching over to a What-If branch from the mainline branch has proved to be a challenge. Forcing What-If branches to focus around small, concise problems should allow for tractable What-If architectures and solutions to be built and explored.

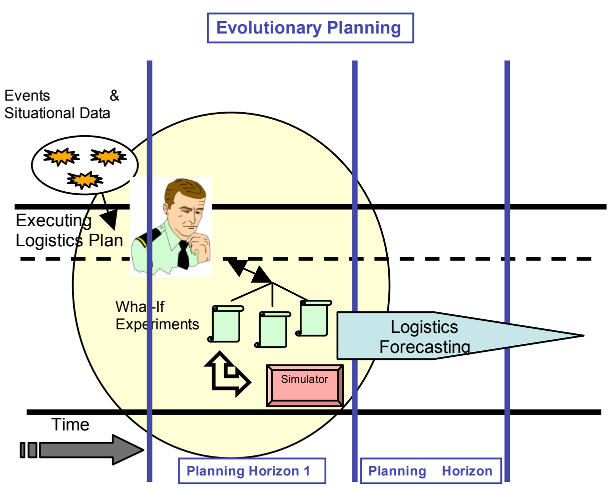

Fig. 3. Evolutionary planning requires continual human/computer iteration that occurs in both the executing and the experimental solution space.

It is clear that the potential uses for planning horizons can quickly grow to include many different flavors of horizons that all affect the plan as a whole. A management substrate will be necessary to coordinate the various horizons in order to gain optimal use of horizons. The manager will allow horizon categories to be prioritized by a planner and will perform horizon boundary merging and optimization.

For example, a user may request that the system focus heavily on ammunition planning and forecasting for a period of 6 hours which happens to represent a high intensive battle. The horizon for high fidelity planning of ammunition would be defined for a period of 6 hours; however, the What-If horizon may span or overlap only 3 of the 6 hours representing the high intensive battle. If ammunition planning is the highest priority, the horizon manager could suggest that the What-If horizon be extended to 6 hours to provide the user with high fidelity options regarding ammunition usage, forecasts and distribution for the 6 hours during the fight. The manager could either notify the user of the suggestion, or automatically extend the What-If horizon depending on a policy setting. The manager should also be flexible in allowing a user to modify a suggestion provided by the system. For example, the user may wish to extend the What-If horizon to include 3 hours before the start of the 6 hour fight in order to anticipate what the state of ammunition might be at the start of the high intensity battle.

Focused Forecasting

By improving specific areas of the plan and focusing replanning efforts on specific agents, we can more readily help users solve particular problem areas and explore multiple options or contingencies for an unstable portion of the plan. The focus area will be defined by a set of agents, a defined planning time horizon, and potentially a defined event for which the system must react to by replanning. Once the focus area is defined, computing resources can be redirected to the components that provide planning solutions, What-If experiments, etc. for the focus area (agents). Previously mentioned problems surrounding agent to agent dependencies must be reduced through more robust application development in order for focused planning to be successful. Predictors and other localizing optimization techniques such as the ones used in NCL will be a necessary part of the logistics application(s). These improvements will make more timely and flexible solutions for a specific portion of the plan available to users. Additionally, these solutions should be provided as high-fidelity solutions that can be executed almost immediately.

In order for the system to provide ultimate flexibility within the focused area, the system must fully understand the problem and be able to generate multiple resolutions with hints from the user. In addition, the system should incorporate knowledge gained from previous planning cycles. By utilizing the intuition and expertise of the user, the problem space will be deliberately constrained yielding timely, targeted and useable logistics support options The system can still autonomously generate support options, but it should redirect its path as soon as the user states any preferences or intuitions that can drive the plan/solution.

The result of this human-machine interaction using focused forecasting tools will be complete logistics plans that are more useable, timely, and realistic since they will be constructed of planning decisions built from many focused forecasting iterations. These types of solutions will allow for a more direct translation from planning to real-world execution for the logistics planner and warfighter.

Execution and Simulation

Planners and decision support tool developers have many different definitions of execution and simulation. For the purpose of this paper, execution and simulation are defined as follows: Execution is defined as a mechanism for the system to consume real-world situational data and events that have or are occurring in real-time. These events are applied to a plan and as the system incorporates event data it replans the future. Simulation is defined as an infrastructure that allows a user to test a plan by mimicking real-world events. The system components that replan and react to information should be the same components whether the execution infrastructure or the simulation infrastructure is being used.

There are both large and small scale simulators in existence today. Many of the larger scale simulators require a significant amount of computing power and are very dependant on simulating the actions occurring at every agent in the society. However, if we were to build or integrate a small scale simulator that could be bound by a problem space as discussed in the focused forecasting section, it would address a critical slice of the end-to-end logistics decision support solution. The details of such a simulator will not be discussed at length in this paper; however, it is important to discuss how such a component might be used.

Assuming that the simulator had limited dependencies on simulating the entire plan, limited dependencies on computing resources and exhibited the ability to allow for human interaction(direction), the simulator could easily be integrated into a system that uses techniques discussed above (evolutionary planning, focused forecasting, and distributed optimizations). Simulations could be performed on What-If scenarios developed in an evolutionary planning cycle or scenarios that were developed for a specific focus area. Simulating logistics support options developed by other components would provide the user with a more complete decision support tool. In fact, you can imagine that simulating some of the options may produce ideas for iterations in planning or What-If cycles that yield ultimate logistics support solutions.

Solution Evaluation and Assessment

The solution areas described above will provide logistics plans that the user and the system will evaluate, assess, and iterate. This will evolve the plan into execution sequences that closely represent the desires of the logistics planner. To aide in the evaluation and assessment process, two algorithms can be used. The first provides an evaluation of the search space that the system will explore in order to solve a problem within a focus area. The second algorithm provides a more comprehensive and multi-dimensional evaluation of the plan or subplans.

Algorithm 1

As previously discussed, the system shall allow for user directed solution searches (e.g. What-If branching and simulation) or autonomous solution searches that can be redirected by the human as desired. The following algorithm can provide the user with an evaluation of search space options that will direct the actual solution search. Consider a case where there is a defined focus area (*FA*) and a defined time (*T*). The system can then generate search options for the user to evaluate that fall within the timeliness and focus area guidelines.

Suppose there is a composite What-If Horizon (*W*) made up of the following tuple: a What-If Time Horizon (*WIT*) defined in hours, a What-If Branch Quantity (*WIB*), and a What-If Perturbation measurement that is defined as HIGH or LOW (*WIP*). The What-If perturbation measurement will represent the frequency of perturbations that can be injected while exploring a single What-If branch.

$W_x = (\mathrm{WIT}_x, \mathrm{WIB}_x, \mathrm{WIP}_x)$

Suppose that the user defines:
T = Solution time as 30 minutes
FA = Focus area as Platoon ABC
FD = Desired plan fidelity level as HIGH.

The system will solve for:
WIT = What if Time Horizon
WIB = What-If Branch Quantity
WIP = What-If Perturbation Measurement
S = Simulation (ON or OFF for a single branch)
TH = Planning Time Horizon

Using the following algorithm:
$T_x = \mathrm{FD}_x\mathrm{FA}_x(\mathrm{TH}_x + \mathrm{W}_x + \mathrm{S}_x)$

An example solution where *FD* = HIGH may be:
(Time Horizon = 2 hours + (What-If Horizon = 2 hours, What-If Branches = 1, What-If Perturbations = LOW) + Simulation = OFF)

The solution space above will cover a planning and What-If time horizon of 2 hours, produce 1 What-If branch with a low frequency of perturbations, and none of the branches will be simulated. The results will be generated within 30 minutes for the defined focus area.

Alternatively, an example using *FD* = LOW might present the user with a solution space as follows:
(Time Horizon = 7 hours + (What-If Horizon = 7 hours, What-If Branches = 3, What-If Perturbations = LOW) + Simulation = ON)

In this solution, low fidelity plans spanning 7 hours (planning and What-If), with 3 What-If branches, low perturbations and a simulation can be provided to the user within 30 minutes for the defined focus area. Note that the simulation is a simulation of the most valuable What-If branch developed during the experiment. The "most valuable" branch will be defined by a different algorithm and or the user.

A key implementation detail of this algorithm is that the options generated must be generated in a second or less, so that the decision time is spent directing the experiments, generating solutions, and assessing the solutions.

Algorithm 2

The second algorithm provides a measurement of how good the overall plan solution is in terms of specific measures. The system may represent the overall, or composite, logistics plan as a set of subplans that can consist of modules. For example, movement resolution (*M*), delivery timeliness (*D*), and load configurations (*L*) might represent an activity module. Suppose movement resolution M_1 is movement accuracy within 5 kilometers and M_2 is movement accuracy within 1 kilometer of a receiving platform (e.g. customer truck). Further, suppose D_1 denotes a required accuracy within 1 day, but D_2 requires delivery accuracy inside of an hour. Finally, suppose that L_1 is a load configuration defined as "basic load", and L_2 is a tailored load configuration. Now consider that we might be able to adjust the resolution of planning by representing the "focal length" of a partial plan by representing its requirements as a MDL tuple, where a platoon support plan for an imminent area clearing operation might define requirements in these terms:

R_1 = (Movement Accuracy = high, Delivery Timeliness = ASAP, Load Specificity = High);

Alternatively, a Brigade's periodic resupply might have requirements of:

R_2 = (Movement Accuracy = Low, Delivery Timeliness = OnTime, Load Specificity = Basic)

R_1 and R_2 are examples of an MDL triple.

Furthermore, consider that multiple plan branches might exist such that partial plans also can be characterized as MDL triples. The desired plan will consist of matching partial plans against the focal length requirements of a plan type.

The plan then becomes:

$Plan_X = W_1M(P_1,R_1) + W_2M(P_2,R_2) + \ldots + W_NM(P_N,R_N)$

where

MDL = 3-tuple of Movement, Delivery Timeliness and Load

Rx =requested MDL (plan need)

Px = partial plan MDL (data created by the system)

M(… , …) = measure of the match quality of the requested plan MDL to the calculated plan MDL, much like a weighted norm.

Wx = weighting on the module or subplan significance

Conclusions and Future Work

The techniques described above will allow users to quickly and efficiently interact with the planning system to generate timely, detailed, robust plans that can be put into action in the real world. The combination of distributed optimization, evolutionary planning, focused forecasting, and execution and simulation enhancements are powerful tools when combined with expert human interactions. These techniques are required to support end-to-end logistics operation planning and decision support tools of the future. Such a system may ultimately be used for training, peacetime planning, as well as mid-stream battling planning and course of action redirection. By reducing the compute power and the complexities of replanning entire logistics support plans, the solution techniques could be used by soldiers on the battlefield, planners at various headquarter levels, or back in CONUS.

These solutions can also effectively breakdown the legacy planning silos (stove pipes) that have limited coordination between operations and logistics warfighters. Perhaps even more importantly, we can imagine a system that bridges planning and execution for operations and logistics, leading to a true, real-time, coordinated, unified tactical planning system.

We intend to further explore the techniques described in this paper in a testbed environment that will allow us to validate the ideas and algorithms presented. The validation process will utilize MOEs (Measures of Effectiveness) whose definitions are beyond the scope of this paper.

References

[1] Cougaar Web Site (http://cougaar.org)

[2] UltraLog Program Description (http://dtsn.darpa.mil/ixo/programs.asp?id=61)

[3] A. Fedyk, J. Berliner, R. Bobrow, M. Davis, B DePass, G Kratkiewicz, R. Lazarus. "Adaptive Optimization of Solution Time In A Distributed Multi-Agent System", IEEE 2005 International Conference on Integration of Knowledge Intensive Multi-Agent Systems

[4] A. Helsinger, M. Thome, T. Wright, "Cougaar: A Scalable, Distributed Multi-Agent Architecture," IEEE 2004 International Conference on Systems, Man and Cybernetics, Oct. 10-13, 2004, The Hague.

[5] The NCL Sapling Evaluation Team, EBR, "Network Centric Logistics "Sapling", Assessing State of the Art Decision Algorithms that Enable NCL", December 2005, unpublished.