AI Magazine Volume 9 Number 3 (1988) (© AAAI)

WORKSHOP REPORT

What AI Can Do for Battle Management

A Report of the first AAAI Workshop on AI Applications to Battle Management

R. Peter Bonasso

The following is a synopsis of the findings of the first AAAI Workshop on AI Applications to Battle Management held at the University of Washington, 16 July 1987. The workshop organizer, Pete Bonasso, sent a point paper to a number of invited presenters giving his opinion of what AI could and could not do for battle management . This paper served as a focus for the workshop presentations and discussions and was augmented by the workshop presentations; it can also serve as a roadmap of topics for future workshops. AI can provide battle management with such capabilities as sensor data fusion and adaptive simulations. Also, several key needs in battle management will be AI research topics for years to come, such as understanding free text and inferencing in real time. Finally, there are several areas—cooperating systems and terrain reasoning, for example—where, given some impetus, AI might be able to provide help in the near future.

Despite the flurry of indignant notes across the Arpanet following the call for participation, there were no picketers or hecklers at the first AAAI workshop on AI Applications to Battle Management. About 30 people gathered at the University of Washington's South Campus Center on Thursday of the week of the national AAAI conference in July 1987 to discuss the use of AI technologies in battle management.

A representative cross section of agencies gave presentations: the AI centers at SRI International, General Electric (GE), Texas Instruments, and the MITRE Corporation's Washington C^{3}I division; Advanced Decision Systems (ADS); Rome Air Development Center (RADC); the Naval Research Laboratory (NRL); the Admiralty Research Establishment (ARE) in Hampshire, England; and Dartmouth College. There were on-lookers from several of these agencies as well as from Unisys, the Rand Corporation, the Army's training and doctrine and materiel development commands, Lockheed, the Defense Advanced Research Projects Agency (DARPA Lieutenant Colonel Robert Simpson), and Ford Aerospace.

As the conference organizer, I prepared a point paper on what I thought the status was of the application of AI to battle management to focus the presentations and discussions. This article uses this paper as its basis and includes workshop discussion comments.

The purpose of this first battle management (alias, BatMan) workshop was to come up with three lists: (1) what problem-solving battle-management tasks the AI community knows technically how to do; (2) what tasks we can't do and have little hope of doing before 1997; and (3) what tasks we think we might know how to do, but it's still too soon to tell. Abstracts I received for the workshop didn't fall easily into these categories but were representative in their problem-solving tasks. So I put forth my version of these lists to stimulate discussion at the workshop and to provide an impetus for presenters to tailor their presentations along lines that agreed or disagreed with the lists.

I first held a short discussion of requirements. I expected (perhaps erroneously) that there would be little argument among the workshop attendees about the main battle-management functions. Then I presented my lists with some justification and rationale for them. Several statements were unqualified. If everyone agreed, then they would serve as absolutes; if not, I hoped to get qualifications or, better, counter-arguments at the workshop. The workshop attendees seemed to work well with the three-list format. As expected, a good deal

of useful discussion centered around which AI techniques fell into which lists.

Required Problem-Solving Tasks

Basically, the battle-management scene is an air-land battlefield or ocean situation in which two opposing forces try to accomplish their respective goals by armed conflict. With the "friendly" force—and at the level of, say, the Army Corps, Air Tactical Operations Center (ATOC), Navy fleet—a goal exists that implies a situation other than the "current situation," that is, the "desired situation." A basic function of battle management is to assess the current situation in light of how close it is to the desired situation (Dept. of the Army 1982).

A subtask of this assessment involves situation development, in which information is acquired about the current situation and put together into a coherent picture of the military situation. The information concerns the status and capability of both the friendly force and the opposing force and data about the environment, that is, terrain and weather. Most of this information, in turn, is acquired through friendly unit situation reports and from electromagnetic and optical sensors that range from satellites to human observers on the ground. Putting the sensor data together is sensor fusion and putting the sensor data together with own-force status, terrain, and weather information, and a priori data such as the opposing force's doctrine of battle, is information fusion. Also, much intelligence information, as well as the friendly force status, arrives at the operations centers by way of free-text messages.

A current plan for operations is usually in effect, and situation assessment goes hand in glove with monitoring the current situation. Monitoring the current situation involves watching certain key battle information items that will indicate whether the current plan is on track. If the deviations are tolerable (the criticality of the deviations are assessed as part of situation assessment), perhaps only a small portion of the existing plan must be replanned. If the deviations are significant or if a new goal is directed by higher authority, then a new plan of operations must be prepared. So, planning and replanning are major tasks in battle management.

The planning process generally involves selecting a course of action (COA) among several reasonable candidates, evaluating them against possible opposing-force countermoves (wargaming), and then developing a detailed plan of how the selected COA will be executed. This detailed plan usually includes subplans for maneuver and fire-support forces, for combat service support (supply and services actions), for intelligence gathering (for monitoring the current situation), and for communications and special support operations (such as engineering tasks).

Most of the tasks outlined here are carried out by staff sections working together in support of the commander in an operations center. Staff sections usually include, but are not limited to, an operations section (usually with sections for managing the current battle and for preparing future plans), an intelligence section, a fire-support section, and an administrative and logistics section. These staff sections work on managing day-to-day problems associated with their particular domains. But for developing, disseminating, and tracking the overall battle plan, these sections work in a cooperative manner, each viewing the situations and COAs from their individual perspectives.

Besides the large search spaces involved (for example, hundreds to thousands of units, many with multiple capabilities and several classes of supply requirements) and the need for real-time evaluation of COAs, what makes battle management difficult in general is the poor data environment of the battle arena. Sensor information is nonuniform, sporadic, misleading, redundant, and erroneous and has varying degrees of accuracy. Communications delays or breakdowns prevent key information from arriving at the operations center or cause it to arrive in an untimely manner. And accurate knowledge of the opposing-force doctrine and capabilities is usually not really determined until the hostilities are under way.

Humans use more heuristic than analytic techniques and often go on intuition in light of this poor data environment. Herein, I believe, lies the genesis of the rationale for using AI and, in particular, expert systems technology. The argument goes, "Whatever it is that battle managers do, let's automate it to help them out. If conventional automation can't support heuristics and intuition, then perhaps AI can."

Workshop Comments: There was essentially total agreement with the workshop's description of the environment, problem-solving processes, and what might be termed battle management software system requirements. Tom Garvey, Charley Ortiz, Jim Vance, and Tom Strat of SRI in their battle management planning paper (see We Can Do Hierarchical Planning) stressed three other aspects of the battle-management environment that make automating the staff processes so difficult. Risk management colors all aspects of battle management, because the environment is malign and provides little margin for error when lives are at stake. Resources constrain all battle-management processes, requiring a constant assessment of the trade-off options. And time is treated as a resource; the situation is dynamic, and predictions and projections tend to be as important if not more important than an assessment of the current status.

What AI Can Do in Support of Battle Management

Generally, most of what we can do for battle management will suffer from what most current applications suffer from: little or no theoretical foundation for reasoning in the domains of interest. What we see now in AI systems for battle management are mostly small, ad hoc toys that if scaled up would fall apart or become unusable. But there are some positive notes.

Workshop Comments: There was general agreement about this section, with the exception of how much planning was achievable.

We Can Do Hierarchical Planning

After a ten-year lull, about five years ago the AI community saw a resurgence of research in planning (McDermott 1987). One area of planning that seems applicable to battle management is hierarchical, nonlinear planning in the manner of MOLGEN, NOAH, and SIPE. This nonlinear approach seems to be what battle staffs take when they develop plans; the COA analysis involves taking a hard look at the first few levels of abstraction of the plan, and the detailed staff planning results in the rest. MITRE's OPLANNER (Benoit, Davidson, and Powell 1985) is an example of how to make the planning system less domain-specific, and it has been used successfully for both strategic and tactical joint-operations planning, though only in a prototype mode.

Workshop Comments: Charley Ortiz of SRI presented a paper on issues in the development of planning systems for battle management (Garvey et al. 1987).In it, the authors agreed that the previous premise was true only in a few well-defined (that is, uninteresting) problem situations. They felt that even then, because we do not know how to formally evaluate such systems, it is not clear how soon any such support software would be a viable reality for battle managers. The presentation proposed the development of a planning testbed, centered around an interactive workstation, where really tough planning problems could be incrementally attacked and where applicable technology-transition issues could be addressed.

We Can Do Stereotypical Planning

The KNOBS development (Brown et al. 1986), which has led to the TEMPLAR contract, is an example of taking a domain-specific function, such as filling out an Air Tasking Order, and using constraint satisfaction to keep the values in the order logically and operationally consistent.

We Can Do Sensor Fusion

The HASP-SIAPS project (Nii and Feigenbaum 1978) was the first in a history of systems developed for detecting patterns in sensor signals that would be indicative of directed activity. SRI's multisensor ESM system (Garvey and fischler 1979), MITRE's ANALYST (Bonasso 1981), and ADS' AMUIDS (Spain et al. 1984) were representative systems that would translate sensor streams into enemy units using pattern-directed inferencing. A DARPA experiment with the 9th Infantry Division demonstrated the near-term potential of such systems in an operational setting (Zymelmann 1985).

Workshop Comments: John Montgomery of ARE gave a presentation describing research in battle management in the naval environment (Montgomery 1987). Four of the six major battle-management functions being addressed in this work concerned sensor fusion and intelligence support, and the sensor fusion work, including knowledge engineering, was two years old. This was yet another example of the direct application of expert system technology to sensor fusion. The main discussion about this topic centered around why the military intelligence community hadn't made knowledge-based sensor data fusion a requirement for its next-generation systems. No one had a clear answer.

Allen's Time Representation Will Do

It seems that once we arrive at a decent logical formalism for time, we can endow AI programs with useful temporal reasoning capabilities. Into what was a dearth of time-reasoning research, several contributors, notably James Allen and Drew McDermott, put forth such formalisms. The AI groups at MITRE (SCAN [SCripted ANALYST]) (Lakowski and Hofmann 1987) and ADS (design for the DARPA Air-Land Battle-Management system) believe that Allen's interval formulation for time is well suited for temporal representation in battle management. I suspect there are others who are following suit. I am not saying that we've solved the frame problem but that we have a representation that can be used for certain temporal reasoning tasks in battle management.

Workshop Comments: Allen Brown, presenting a paper that he and Dale Gaucas prepared on prospective situation assessment essentially agreed with the main intuitions of Allen's interval representation (Brown and Gaucus 1987). But they found this representation to have insufficiently rigorous foundations, to be inadequate to express certain causal phenomena, and to be inefficient computationally. They felt that causal and inertial theories are convenient for expressing the causal content of plans and are more efficient computationally. He thus proposed making use of Shoham's causal and inertial theories and modal logics of belief (with Kripke interpretations) layered atop a general reified logic of time that subsumes Allen's work.

In general, however, participants were cautious about settling on one representation for such a complex phenomenon as temporal reasoning.

We Can Make Great Decision Aids

The Lisp machine development environments (and carbon copies thereof) have been producing more in the way of better user interfaces and decision aids and less in the way of making the artifact intelligent. Windows and mousing, smart editors and debuggers, and all manner of frame-representation-language (FRL) -based data viewers are not only becoming preferred by users waiting for AI, they can also be developed in one-third the development time of a conventional system.

Workshop Comments: Don Henager of RADC really seconded this point by describing RADC's advanced C^2 environment designed to test and evaluate a variety of AI and non-AI techniques for decision aids, many (perhaps most) of which were going to succeed on the basis of the rapid development capability of AI software. The overly ambitious projects would be soon found out to be so and could be reoriented early enough to regroup and reassign funds.

Laura Davis of NRL put forth the thesis that there is a real need for a family of command decision aid development tools in an integrated, rapid prototyping environment (Davis

and Liebowitz 1987). Most of the participants believed that, on the whole, such an endeavor would not be practical because it would be premature. But Bob Simpson pointed out that the ABE (A Better Environment) project at DARPA should be of interest here. ABE is a project aimed toward producing an architecture and methodology for building intelligent systems in general by integrating heterogeneous components as well as providing a modular, expandable library of skeletal system structures.

A main discussion here centered around what is lost when AI systems made in these rapid-development environments are carried over to target environments of lesser capability. Going from Lisp machines to Sun systems requires some compromises, but going from Lisp machines to PCs for any significant problem seems to require a total reevaluation of what we wanted out of the AI system to begin with.

We Can Make Better Simulations

Object-oriented simulations such as SWIRL (Simulated Warfare in the ROSS Language) and TWIRL (Tactical Warfare in the ROSS Language) at Rand (Klahr et al. 1982, 1984) and BEM (Battlefield Environment Model) at MITRE (Nugent 1984) are more effective for studying battle management or for evaluating the effects of plans and intelligence collection than conventional models of warfare. Making explicit the behavior of intelligent agents on the battlefield in simulations allows for a more causal understanding of the dynamics of warfare.

Workshop Comments: Some people think that object-oriented programming is not a spin-off of AI. SIMULA, a simulation language, is purported to have message passing and method-like procedures. Yet the inheritance of methods and the mixing of methods pertaining to objects seem still to be unique to the AI development environments.

What AI Cannot Do in Support of Battle Management

Natural language understanding of messages, building deep systems, managing uncertainty, and obtaining real-time performance are among the tough problems facing AI researchers. These problems are significant enough to dampen most hopes for good AI battle-management systems for the next 10 years.

Workshop Comments: Again, there was general agreement with the items in this section. Perhaps to AI researchers these are obvious; but with help from such commercial influences as Star Wars and Speak-and-Spell, the military often doesn't seem to recognize these limitations.

We Can't Read Free Text

As mentioned earlier, text messages arrive at an operations center minute by minute. The content of the messages ranges from periodic updates of unit materiel to critical intelligence messages, and, except for some header information, the form of the message is mostly free text. So, the main use of natural language understanding in an operations center will not be to aid discourse but to interpret messages and to figure out what databases to update and what addressees ought to have the information. But natural language systems that read free text and discern its meaning for these or similar purposes still elude researchers, and they won't be something to consider as a possibility in the near future.

Workshop Comments: Tom Garvey pointed out that researchers at the Naval Ocean Systems Center have been working on a system to help constrain free-text generation when the message is composed, with the intent of making it readily machine readable at a later time. This approach seems much more practical for both the near and far terms.

We Can't Build Deep-Knowledge Systems

There are no experts in battle management. Battle management is learned at military training centers and then practiced during military exercises, but it has been over 15 years since this country was involved in a major military operation (Viet Nam) and well over 30 years since our involvement in a more conventional type of conflict (Korea). Too, the battle environment anticipated for the next conflict will be vastly different from that of either of these involvements. The Pattons and MacArthurs—with their expert (compiled) knowledge of battle management, tactics, and strategy—just don't exist. So what we build using knowledge-based architectures are not systems that perform as experts but systems that perform as practitioners, with hardly any actual battlefield experience to justify them.

What is needed then to shore these systems up and to put them into a better position to be adapted to the next war is a better understanding of the nature of the ground, air, or sea battle environment and the causal relations involved in the management of military operations. In short, a commonsense theory of battle is required. Commonsense reasoning is a rich research field at the moment, but the knowledge required is perhaps several orders of magnitude greater than expert knowledge, and there have been no breakthroughs yet that point to our being able to manage this kind of knowledge in the near future.

Workshop Comments: Joe Tatem (1987) of Texas Instruments supported this point by recounting his experiences in managing the large knowledge bases associated with the FRESH (Navy Force Requirements Expert System) resource allocation system. Tatem recounted some of the challenges and the lessons that were learned while acquiring and maintaining these large knowledge bases, and we can extrapolate to conclude that managing a knowledge base with additional, everyday naval knowledge will be even more difficult.

We Can't Manage Uncertainty

There are several representations of uncertainty in the AI community, and there are spirited discussions each year about the relative merits of each, but it remains to be seen which or what combination of techniques is correct for battle management. There is even speculation that it is more appropriate for battle management to use an alternative-worlds representation rather than a numeric one. Suffice it to say that no one is

attempting to determine what representation is correct for battle management, and I submit that this problem is compounded by the fact that we don't actually build systems that are expert as MYCIN or PROSPECTOR, for example.

Workshop Comments: Ben Wise (1987) of Dartmouth gave a presentation on this topic, and the thrust of his premise was that at least the use of classic probabilities would be essential for managing uncertainty in battle management. His comparison of other calculi for uncertainty—fuzzy sets, the Dempster-Shafer rule, Bayes theorem—provoked quite an animated discussion that was not resolved, thus proving this point. Wise also proposed that the treatment of uncertainty in battle management had to be integrated with measuring the utility of the various types of information for which there was uncertainty.

SRI participants believe that both multiple-worlds assumptions and numeric representations will be equally important in the pursuit of uncertainty reasoning for battle management and that looking for "the" way of managing uncertainty might be a mistake.

Most attendees also agreed that we don't really know how humans conceptualize or make use of the notion of uncertainty, for battle management in particular, and that there is a need to gather empirical data through experimentation.

Going from Lisp machines to Sun systems requires some compromises, but going from Lisp machines to PCs for any significant problem seems to require a total reevaluation of what we wanted out of the AI system to begin with.

We Can't Reason in Real Time

The time frames in operations-center battle management for conventional conflicts is hours to days; the time for managing the application of defensive weapons against nuclear missile threats ranges from minutes to seconds. To date, AI-based battle-management systems are often not much more than toys that prove the principle of the technology application, and they don't execute especially well. Scale them up with real-world data and adequate knowledge bases, and a significant amount of research will still have to be done to get them to perform within conventional war time frames. The community is already looking to parallel architectures for better performance with these applications in mind. Now, try for another order-of-magnitude speed-up of a full grown system for, say, the Survivable Adaptive Planning Experiment (SAPE) or the Strategic Defense Initiative (SDI) and we're easily 10 years away.

What AI Cannot Do Now in Support of Battle Management

There are several research areas in battle management in which the AI community, for whatever reasons, has not put forth a lot of effort. Replanning, cooperating systems, wargaming, and terrain reasoning are such areas, ones for which I believe solutions can be found over the next five years or so.

Workshop Comments: Most of the discussions of the workshop centered around this set of points; that is, how much effort some of these would take to move them into the "things we can do" list, whether some (such as replanning) even belonged here, and so on. It seems, therefore, that subsequent workshops ought to address some of the issues in this section.

We Can't Replan

Right now, replanning in AI means to plan again. But in battle management, replanning means being able to correct a plan that's not succeeding or that's succeeding too well. We shouldn't plan again, because the time available for planning is usually greatly reduced when the need for replanning is discovered, and the resources that were so carefully balanced during planning have already been consumed or committed. Some research efforts are under way in this area (for example, one at MITRE that uses an assumption-based truth maintenance system [ATMS] to keep track of alternative plans) that lead me to believe that replanning will be soon be possible.

Workshop Comments: Despite the current work in this area, Jack Benoit of MITRE, giving a presentation on fundamental replanning issues (Benoit, Davidson, and Powell 1986a), outlined how difficult this area is. He raised such questions as, how do you

know that you have a relevant, significant, causal event that will dictate that replanning is in order? How much replanning is sufficient to fix a plan? How do you counteract or make use of the inertia of the agents carrying out the original plan? It was not clear, based on the discussion about whether nonlinear planning belonged on the "can do" list, whether (1) replanning isn't so intricately bound up in planning in general as to be a separate topic, or (2) it belongs on the "can't do" list.

... some believe that cooperating systems is a "can do" item for battle management, in the sense that software systems engineers could take it over.

We Don't Know How to Make Systems Cooperate

The MITRE ALLIES experiment sponsored by RADC (Benoit et al. 1986b) was the first instance of getting two disparate knowledge-based systems to cooperate toward a common goal. Investigators discovered that asynchronous cooperation among AI systems supporting the primary staff sections was desirable but required a common set of semantics to convey the capabilities of each system to the others. This language of battle management needs developing, and work is under way to do so, but it will easily be several years or longer before a reasonable subset is nailed down to be useful in an operations center setting.

Workshop Comments: Actually, except for cognitive science issues involving human models of cooperation that machines ought to emulate, many of the workshop participants believe that, for battle management, cooperating systems are an engineering problem. The problems discussed in the ARE presentation (the use of blackboard architectures to control cooperation) and in the ADS presentation by Roland Payne on the air-land battle management (ALBM) program (the use of a separate agent—force control—to orchestrate cooperation) chiefly covered data sharing and intersystem communication, rather than making inferences across a heterogeneous set of knowledge bases. Apparently, though the human staff elements cooperate in battle management, it is more a model of delineated areas of responsibility that have a few overlapping facets. As a result, some believe that cooperating systems is a "can do" item for battle management, in the sense that software systems engineers could take it over.

Reinforcing the belief that cooperating systems is a "can do" item, Tom Garvey pointed out that SRI's LADDER system was developed in the mid-to-late seventies to compose answers to queries that required access to multiple databases with inference in order to draw appropriate conclusions. Current work at RADC also centers around an intelligent operating system that takes a similar approach to supporting a battle staff that uses multiple decision aids and a common database.

We Don't Really Do Situation Assessment

Although sensor fusion seems to be amenable to knowledge-based systems, and examples of determining the order of battle and enemy intentions are increasing, situation assessment—measuring the success of the operations plan based on the current situation—has not been done much to date. I'm not even talking about projection into the future, just a static evaluation of whether the current situation is closer or farther away from the desired situation.

Workshop Comments: Both the ARE presentation (Montgomery 1987) and Brown's (1987) paper on prospective situation assessment pointed to work in true situation assessment. The main discussion point here was whether situation assessment was integral to or just a companion of the planning function. Brown contended that it was integral to planning and gave a convincing calculus for achieving a look-ahead view of the situation as part of a planning problem. The ARE work pointed to situation assessment as a module in a multilayered assess-decide-act decision system.

We Don't Wargame Much

Chess problems used to be (and in many places still are) the grist for the AI graduate student mill. The n-ply search strategies and heuristic evaluation schemes seem appropriate to COA analysis. While some work is under way in contingency networks (McDermott 1987), few people seem to be looking at AI approaches to gaming the strategies of conducting an air, sea, land, or joint campaign, though I think it ought to be within the reach of a couple of research grants.

Workshop Comments: Again, the discussion here centered around whether wargaming was just another aspect of planning—evaluating plan alternatives. If the ideal simulation consists of intelligent gaming agents, then the simulations will have a planning function imbedded in them, and there will perhaps be an orchestrating agent (worrying about making sure the wargame was achieving its purpose) that would be able to plan the game.

We Haven't Done Terrain Reasoning

Much ground-battle causality is directly tied to the movement of forces across the earth's surface. Most of the reasoning that goes into the analysis of the battle environment before the battle (Intelligence Preparation of the Battlefield) is focused on reasoning about what the opposing

forces can do on a given piece of terrain. Yet, no existing knowledge-based system for battle management uses terrain reasoning as a basis for developing operations plans or for performing a situation assessment.

I believe that finding a suitable qualitative representation for terrain features and then using reasonably straightforward, knowledge-based techniques ought to be feasible, but until someone takes the job on in earnest, terrain reasoning is a major missing piece in the AI community for battle management.

Workshop Comments: Several research endeavors making much use of terrain reasoning were pointed out: DARPA's autonomous land vehicle (ALV) program, for example, and the synthetic aperature radar (SAR) programs. SRI's helicopter planning work is also addressing some of these issues (see also Kuipers' and Levitt's [1988] cognitive map treatment). The problem seems to be that none of these programs are relating their terrain representations or their reasoning heuristics to a formal model of terrain that could also be useful for other projects, such as situation assessment for battle management.

Summary

I believe we formulated a reasonable list of topics to be discussed at future workshops: planning and replanning, uncertainty in battle management, formation of logical theories of the battlefield, and reasoning in real time. Because DARPA continues to host excellent workshops on planning under the Strategic Computing Program, perhaps an alternative topic could be exploration of the relationship of planning to the other battle-management functions. This article hopefully provides some signposts about what can be expected from AI for battle management in future years.

References

Benoit, J. W.; Davidson, J. R.; and Powell, E. G. 1986a. Lessons Learned Developing an AI Planning System. Paper presented at the Applications of Artificial Intelligence to Engineering Problems Conference at Southhampton University, UK.

Benoit, J. W.; Bonasso, R. P.; Davidson, J. R.; Ellenbogen, J. C.; Laskowski, S. J.; Wong, R. W.; Powell, E. G.; Schwamb, K. B.; and Tallis, H. C. 1986b. ALLIES: An Experiment in Cooperating Expert Systems for Command and Control. In Proceedings of the IEEE Expert Systems in Government Conference, 372-380. Washington D.C.: Institute for Electronics and Electrical Engineers..

Benoit, J. W.; Davidson, J. R.; and Powell, E. G. 1985. Artificial Intelligence in Joint Deployment Planning. Paper presented at the Artificial Intelligence in Engineering Conference at George Washington University.

Bonasso, R. P. 1981. ANALYST: An Expert System for Processing Sensor Returns. In Proceedings of the first Army Conference on Knowledge-Based Systems for C3I, 221-245. Ft. Leavenworth, Kan.

Brown, A. L., Jr., and Gaucas, D. E. 1987. Prospective Situation Assessment (Extended Abstract), Technical Report, GE Corporate Research and Development Center.

Brown, R. H.; Millen, J. K.; and Scarl, E. A. 1986. KNOBS—The final Report (1982), Technical Report, RADC-TR-86-86, Rome Air Development Center, Griffiss Air Force Base, N.Y.

Davis, L., and Liebowitz, J.1987. Sharing the Solution: The Need for Generic AI Decision Support Development Tools in Battle Management. Paper presented at The first AAAI Workshop on AI Applications to Battle Management at the Universtiy of Washington.

Department of the Army. 1982. Staff Organization and Procedure, Technical Report, FM 101-5, U.S. Army AG Publications Center, Baltimore, Md.

Garvey, T. D., and Fischler, M. A. 1979. Machine-Intelligence-Based Multi-Sensor ESM System, Technical Report, AFAL-TR-89-1162, AF Wright Avionics Lab, Wright-Patterson AFB, Oh.

Garvey, T. D.; Ortiz, C. L.; Strat, T. M.; and Vance, J. O. 1987. Fundamental Issues in the Development of Planning Systems for Battle Management. Paper presented at The first AAAI Workshop on AI Applications to Battle Management at the Universtiy of Washington.

Klahr, P.; Ellis, J.; Giarla, W.; Narain, S.; Cesar, E.; and Turner, S. 1984. TWIRL: Tactical Warfare in the ROSS Language, Technical Report, R-3158-AF, The Rand Corporation, Santa Monica, Calif.

Klahr, P.; McArthur, D.; Narain, S.; and Best, E. 1982. SWIRL: Simulated Warfare in the ROSS Language, Technical Report, N-1885-AF, The Rand Corporation, Santa Monica, Calif.

Kuipers, B. J., and Levitt, T. S. 1988. Navigation and Mapping in Large-Scale Space. *AI Magazine* 9(2): 25-43.

Laskowski, S. J., and Hofmann, E. J. 1987. Script-Based Reasoning for Situation Monitoring. In Proceedings of the Sixth National Conference on Artificial Intelligence, 818-823. Menlo Park, Calif.: American Association for Artificial Intelligence.

McDermott, D. V. 1987. DARPA-Sponsored Planning Research: Report and Prospectus, Technical Report, YALEU/CSD/RR #552, Dept. of Computer Science, Yale Univ.

Montgomery, J. D. 1987. Battle Management. Paper presented at The first AAAI Workshop on AI Applications to Battle Management at the Universtiy of Washington.

Nii, H. P., and Feigenbaum, E. A. 1978. Rule-Based Understanding of Signals. In P*attern-Directed Inference Systems,* eds. D. A. Waterman and R. Hayes-Roth, 483-501. New York: Academic.

Nugent, R. O. 1984. MITRE's Battlefield Environment Model. Working group presentation at the Operations Research Society of America/The Institute of Management Sciences Conference in San Francisco, Calif.

Spain, D. S.; Orr, G. L.; O'Reilly, C. A.; and Drazovich, R. J. 1984. Automated Multi-Sensor Unit Identification, final Report, TR-3030-2, AI & DS, Mountain View, Calif.

Tatem, J. E. 1987. Engineering Issues in the Maintenance, Validation, and Acquisition of Large Knowledge Bases. Paper presented at The first AAAI Workshop on AI Applications to Battle Management at the Universtiy of Washington.

Wise, B. 1987. Uncertainty in Battle Management. Paper presented at he first AAAI Workshop on AI Applications to Battle Management at the Universtiy of Washington.

Zymelmann, A. S. 1985. An AI Technology Insertion Experiment With ANALYST. In Proceedings of the Expert Systems in Government Symposium, ed. by K. N. Karna, 200-209. Washington D.C.: IEEE Computer Society.