Image Cover Sheet

CLASSIFICATION	SYSTEM NUMBER 149748
UNCLASSIFIED	

TITLE

TOWARD TIME-DEPENDENT PLANNING OF MISSILE ALLOCATION AND ENGAGEMENT SCHEDULING IN A NAVAL THREAT EVALUATION AND WEAPON ASSIGNMENT \(TEWA\) SYSTEM

AN: 95-00511

System Number:

Patron Number:

Requester:

Notes:

DSIS Use only:

Deliver to: NL

yo4945

Toward Time-Dependent Planning of Missile Allocation and Engagement Scheduling in a Naval Threat Evaluation and Weapon Assignment (TEWA) System

Bruce A. Chalmers[1]
Situation Assessment and Resource Allocation Group
Command and Control Division
Defence Research Establishment Valcartier (DREV)
2459 Pie XI, Courcelette, PQ, Canada, G0A 1R0

#149748

Abstract

Planning the allocation of limited defence resources and their engagement scheduling is an important function of a weapon engagement manager in a supportive single-ship or cooperative multi-ship battle management system. The manager is an intelligent agent in a naval command and control (C2) system for threat evaluation and weapon assignment (TEWA) that must plan, coordinate and direct point or local area defence actions involving hardkill and softkill weapon systems to achieve mission goals. It has to operate in complex dynamic and uncertain battle environments under critical constraints on real-time performance. The system must be capable of interacting with the battle world and a commander (the supportee) in a responsive, reliable and predictable manner.

We present in this paper a model of a deliberative planning function embedded in a real-time layered control architecture for a weapon engagement manager in a naval anti-air warfare (AAW) TEWA system. We then specialize it to develop an adaptive utility-driven planning function for a surface-to-air missile (SAM) engagement manager for single or multiple naval platforms. Its deliberation, based on feedforward control, is concerned with conditionally planning and scheduling assignments to anti-ship missile (ASM) threats of SAMs and the use of illuminator resources to support the guidance of semi-active missiles. Because deliberation uses state information derived from predictions of the evolution of a dynamic world, plans are used to guide actions at lower levels of the control hierarchy, but not to control them. Computing an optimal plan is NP-hard, which makes it impossible to guarantee optimal response within real-time deadlines. To overcome this, we outline an adaptive approach based on off-line knowledge compilation, on-line classification of battle scenarios and almost anytime planning to permit a resource-limited agent to generate effective plans.

Introduction

Driven by the worldwide proliferation of ASMs and technological improvements in sensor, guidance and communication technologies, naval AAW defence systems in the nineties and beyond may be required to deal with high density threat engagements that occur in a timescale of minutes or seconds. This is leading to an enlargening of the size of the battle space as use of a variety of limited renewable and non-renewable defence resources is coordinated in real-time at both the single platform and cooperative multiplatform levels and allocated and scheduled to optimize protection of own ships and defended assets [1, 2].

Such resource allocation problems are an important component of naval battle management. Some problem formulations have previously been studied. For example, cooperative force level missile launch scheduling is conceptualized in [3, 4] as a deterministic production floor shop scheduling problem of minimizing the total weighted flow time, subject to time-window job availability and machine downtime unavailability side constraints. Formulations based on the use of Markov decision processes appear in [5-7]. Techniques for missile launch scheduling for single platforms are presented in [5, 6]. Selection of dynamically arriving tasks (threats) with deterministic deadlines and renewable resource allocation for a naval battle group are considered in [7].

Our research is addressing such resource allocation problems in the general context of developing a weapon engagement manager for a

1 Phone: (418) 844-4691
E-mail: bruce@cc.drev.dnd.ca

supportive autonomous naval TEWA system that must plan, coordinate and direct in real-time point or local area AAW defence actions. Both single-ship and coordinated force level are being considered. Multiplatform resource coordination assumes a robust communications network and a computation capability that permit local information from a variety of distributed sources or sensors, including radar, electronic support measures (ESM) and identification of friend or foe (IFF) responses, to be continuously fused to provide a force tactical situation picture. Fully coordinated firing actions may then be achieved within a centralized or virtually centralized battle management control architecture. In favour of an evolutionary approach to the research, defence response is assumed for now to be layered according to the type of defensive systems employed in a layer and explicit consideration is given only to hardkill systems. Integration of softkill ship defence systems, including electronic countermeasures (ECM), decoys and chaff, is envisaged at a later stage.

Specifically, we present in this paper a model of a deliberative planning function embedded in a real-time layered control architecture for a weapon engagement manager in a naval AAW TEWA system. We then specialize this model to one for conditionally planning SAM firing and guidance actions against ASMs, categorized as sea-skimmers, shallow-divers, or high-divers depending on their attack profile. The reader may consult [3] for a more detailed description of the battle environment in which the manager may be expected to operate and of the opportunities for and the potential performance benefits to be derived by multiplatform coordination. The model is based on using temporal projection of the current tactical situation picture and a normative evaluation of potential histories of the battle world to formulate a conditional action plan. Optimal planning with respect to the model is NP-hard, which makes it impossible to guarantee optimal response within real-time deadlines. To address this, we also outline an adaptive approach based on off-line knowledge compilation, on-line classification of battle scenarios and almost anytime planning to permit a resource-limited agent to generate effective plans.

Block Specification and Layered Architecture

A weapon engagement manager is an intelligent agent in the TEWA system with a number of adaptive real-time performance requirements on its planning. For example, there is a threat-dependent hard deadline associated with processing each of a time-varying number of threats before it causes substantial damage. It has to dynamically interleave, and even overlap, incremental planning and execution, with (re)planning triggered by the occurrence of significant events in an internal model of a dynamic world in which new events that impact the plan horizon may occur at any time. It is a limited rational agent [8] since it is situated in a failure prone computing environment in which it competes for access to limited computational resources (processing time, space) for its real-time problem solving and deliberation. In addition, even if hard deadlines are respected, there is a soft real-time cost associated with the time spent on deliberation prior to acting. For example, the range of options for engaging a threat prior to its hard deadline (the battle space of the threat) generally diminishes with the passage of time. To illustrate this simply, consider that the number of feasible engagements in the period from birth to death of a hostile track is monotonically decreasing in the time of first engagement of the track. Finally, the quality and completeness of temporal projections (track trajectories, etc.) of the state of the battle world on which the manager's reasoning is based can vary from one moment to the next.

We are investigating an approach to designing a manager that satisfies these requirements. Two aspects of this approach are now briefly reviewed. The first, examined in the remainder of this section, is based on the use of a multi-layered real-time architecture. The second, examined in the next section, describes control parameters for engineering an adaptive real-time agent.

The principal idea behind the use of a layered architecture is to hierarchically decompose the functions of the manager depending on a number of considerations, including the level of abstraction in the internal world model used in a layer, the knowledge representation formalisms and type of reasoning involved, the nature of its control mechanisms, the temporal characteristics and real-time constraints on the inter- and intra-layer time-dependent information flows (fluents), as well as the transformations on these fluents. A fundamental consideration is the principle of Saridis [9] of combining "increasing intelligence with decreasing precision", whereby a plan generated at one level of the hierarchy is used to influence actions at a lower level that interacts more directly with the dynamic battle world, but not to control them. There is an extensive literature within the areas of artificial intelligence and intelligent control dealing with layered architectures as a basis for designing intelligent agents for control systems.

Fig. 1 presents a simplified block specification of a two-layered real-time control architecture for the manager. It concentrates on showing the decomposition of the functions (blocks) and information flows (arrows) mentioned in this paper. Not shown, for example, are fluents associated with potential interactions between the manager and the supportee via a man-machine interface (MMI) manager in the C2 system. Such interactions would be based on a definition of the division of responsibility, depending on operational context, between the system and the supportee.

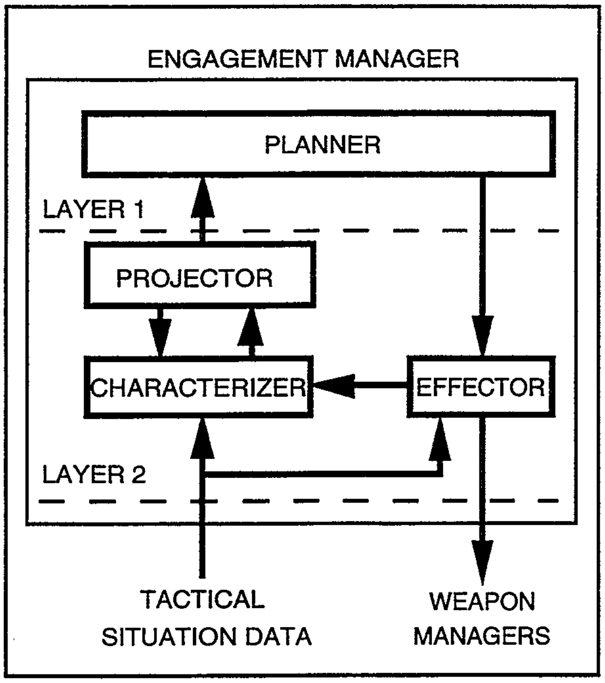

FIGURE 1: Layered architecture.

The *planner* computes an effective plan for assigning and scheduling weapons and guidance systems over some time horizon, subject to firing doctrine and resource availability. It provides service in response to the occurrence of an event arising from a significant change in the tactical picture that requires deliberative (re)planning. Recognizing such significant events within temporal constraints is handled by the *characterizer,* using the internal picture of the battle world and information from the projector and effector. The *projector* extrapolates threat tracks over a forecast horizon. It uses this information to predict when potentially hostile tracks are engageable, from which platforms and by which weapons on such platforms, as well as measures of effectiveness of such strike actions. By communicating with the weapon managers, the *effector* coordinates and directs the execution of plans received from the planner and monitors outcomes of strike actions.

The planner bases its plans on projections from the projector. Such projections will necessarily be defeasible due to statistical error and incomplete or imprecise information, leading to nonmonotonic reasoning. However, they are assumed to be perfect at the level of world abstraction of the planner. An implication is that the effector must implement plan repair strategies when real-time deviations from projections are "small". Hostile tracks that cannot be engaged in this manner must be fed back via the characterizer for deliberative replanning; an example would be a situation in which the effector drops a hostile track from its list of active engagements because it is unable to satisfactorily resolve a real-time resource contention that was not present in the projections on which its higher-level action recommendations were based.

Tactical situation fluents are related to a number of assessments of the threat, the defence force and its mission [10]. They also provide control parameters needed for the next planning episode. There are two data paths in the architecture of Fig. 1 leading from the arrival of such fluents to the weapon managers. The first is via the effector in layer 2 only. This path implements *feedback control.* Fluent updates along this path are periodic (individual fluents may be of different periods), driven by sampling and processing of dynamic real-time sensor information. Reactive responses to changes in the tactical picture are effected. The second data path is via both layers 2 and 1 and implements *feedforward control.* Fluent updates from the projector via the planner to the effector are asynchronous and fluent transformations are of a deliberative nature. While response to changes in the tactical situation along this path is expected to be fast, it is not on the same timescale as that on the other.

Real-Time Specification

We are concerned here only with the deliberative planning function in a manager. A traditional approach to real-time system design is to "guarantee a response after a fixed time has elapsed, where the fixed time is provided as part of the problem statement" [11]. While this approach may be appropriate for real-time specification in layer 2 of our architecture, it is inappropriate in layer 1 in view of its adaptive real-time performance requirements. Moreover, as Zilberstein [12] points out, "this conservative approach leads to inflexible systems that may be under-utilized since in many domains there are no clear, rigid deadlines". He proposes an alternative view of real-time systems based on operational rationality. Its thesis is that "the fundamental property of an operationally rational agent is the capability to vary its deliberation time according to time pressure".

Our real-time specification of the planner is motivated by the above considerations. It provides specific control parameters for engineering an adaptive real-time manager. In addition, further incorporation of Zilberstein's notions of a global time-dependent utility function of an agent (to dynamically measure the performance benefit and cost of planning) and of its conditional performance profile (as a function of the quality of its input fluents) suggests an interesting extension of the specification aimed at developing a metaplanning framework for the manager. This framework would explicitly account for the effect on the quality of the manager's performance of the quality and timeliness of the tactical situation picture it receives; also, it would permit the metaplanning required to actually tradeoff in real-time plan quality, computing time devoted to planning and the allocation of computational resources. Developing and embedding such a framework in the architecture is a subject of ongoing research.

Timing facilities for maintaining wall-clock time to permit synchronization with events in the battle world are assumed to be supported. On a service call, the deliberative planner receives a temporal projection of the force tactical situation picture for a forecast horizon up to time T. This projection is derived from the internal perception of the history of the world up to some elapsed time t. The planner may provide service in one of two anytime modes. In contract mode, its service is to produce in Δt units of computing time an effective plan for a plan horizon within the forecast horizon; Δt is specified at the time of the service call. The plan will be eligible for use by the effector commencing at time $t + \Delta s$ (the start of the planning window for the service call), where $\Delta t < \Delta s$, and it must be consistent with defence commitments up to this time. In interruptible mode, Δt is not specified to the function which may be interrupted unexpectedly. The interruptible mode has an advantage over the contract mode of providing for more run-time scheduling flexibility and it can cope with time pressure created by unpredictable variability or degradation in computational resources.

Values of Δt, Δs, T and the plan horizon are determined dynamically and can be used to engineer an adaptive real-time agent. For example, Δt can depend on time-varying conditions in the battle world, the current processing and communication load of the system, and system timing requirements that may be imposed to allow, depending on battle context, interaction and positive control by the supportee prior to plan implementation. Δs can control the sensitivity of the agent's response to unexpected run-time events. T can be based on estimates of the current quality of projections of the evolution of the battle world (quality may be measured by uncertainty ellipsoids, for example), accounting for loss in accuracy of predictions as they extrapolate into the future. In addition, T can be influenced by an estimate of the completeness of projections (based on run-time monitoring) and can be used to bound plan depth when the velocity of unanticipated change in the world is so high or its behavioural inertia so small that deep planning is ineffective and a waste of computational effort. The length of the plan horizon depends on Δt and on the computational performance profile of the planner.

Conditional Planning Model

The SAM engagement model assumes one or more ships under attack by ASMs. Support ships, of which there is at least one, can perform the functions of surveillance/tracking, SAM launching for point or local area defence in the missile engagement layer, point defence in their close-in layer, kill assessment, and radar support (when necessary) for missile guidance. Included are home-all-the-way (HAW) guidance and mid-course guidance with semi-active terminal homing. Missile firing actions can be shoot-look-shoot or some related generalization of this doctrine involving salvos (e.g., shoot-shoot-look-shoot); firing policy may even use variable salvo size, dynamically determined from one planning episode to the next.

We now present an unbounded rational agent model for the planning function of a SAM engagement manager. Its application assumes a dynamical system representation of kinematic entities in the battle world that over the forecast horizon of a planning episode is reasonably inert. This is to permit its dynamic process model to be estimated with adequate accuracy to justify deliberative reasoning about actions over the horizon. Firing actions are capable of influencing this inertia, subject to uncertainty on the effects of such actions. Plans are conditional on outcome assessments available to the effector at plan execution time. This is aimed at reducing the frequency of deliberative replanning arising from unsuccessful engagements. The price of this approach, however, is potentially greater computational load in generating conditional plans. (An alternative approach, based on unconditional planning, is also currently being examined for purposes of comparison.)

Interaction by the manager with the battle world is modelled as the evolution of a history H [13] over time. H is a mapping from timepoints to the internal representation of the state space of the world. The portion of the history that has been realized up to the current time t is obtained from the perception of the world state contained in the tactical situation picture. The conditional planner is assumed to provide control plans related specifically to missile firing and guidance actions over some plan horizon. A defence plan P, derived from H, is a conditional plan represented by a labelled directed tree. A node with one direct successor corresponds to a defence action/activity. A defence activity takes place over some time interval. An example is continuous wave or interrupted continuous wave illumination of a specific threat by a multifunction radar/illuminator (MFR/I) to enable guidance of a semi-active missile so that it homes to its intercept with the threat. An activity is represented by two discrete defence actions corresponding to starting and ending the activity. A defence action includes these actions as well as missile firing actions. Each firing action involves a specific ship-threat pairing. A node with two or more direct successors, depending on the granularity of the stochastic threat destruction or ship damage model used, is a conditional node corresponding to a stochastic outcome of either a firing action (consequence-of-action uncertainty at action time) or a threat strike on an own ship, including both support and protected ships. Nodes are labelled with their time of occurrence and the state of the battle world at that time. For a plan to be feasible, the time-stamped action/outcome sequence implied by successive nodes on any directed path in the plan tree must be realizable; in particular, this means that timing behaviour determined by kinematic laws governing missile and threat motion must be respected and firing doctrine and renewable and non-renewable defence resource constraints on fire control radars/illuminators and SAMs must not be violated.

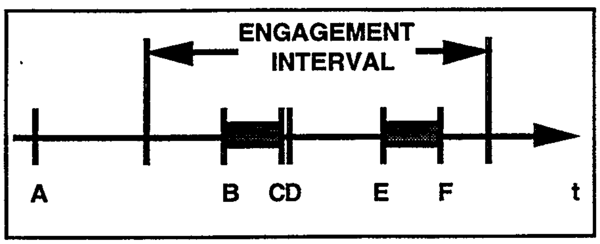

FIGURE 2: Action time line.

Fig. 2 illustrates an action time line of a conditional plan for an engagement involving a single support ship with (at least) two SAMs and a single threat. Illumination is assumed to be feasible in the time intervals indicated (shown shaded). The SAM system uses mid-course guidance and semi-active homing. A SAM is fired at A. It receives command guidance from the ship in A-B, is illuminated in the terminal phase B-C for a possible intercept at C. C corresponds to a conditional node in the plan tree, representing the outcome of the firing action at A. If intercept fails, a second SAM is fired at D, receiving command guidance in D-E and illumination in E-F. The SAM engagement interval, defined as the interval from first feasible time of intercept to last feasible time of intercept, is also indicated. A-C corresponds to the fly out time of the first SAM and D-F to that of the second when the second firing action is initiated.

Based on resource availability and control on a planning episode, the goal of the unbounded rational agent is to choose a most preferred feasible plan over the plan horizon. In a decision-theoretic framework, given the probability distribution on the outcome states, $s \in S$, of the world at the end of the plan horizon and a utility function u on outcome states, the agent chooses an optimal plan, P^*, according to:

$$P^* = argmax_P v(P), \qquad (1)$$

where the value $v(P)$ of plan P is the expected projected value of u resulting from its execution, given by:

$$v(P) = \sum_{s \in S} prob(s \mid P) u(s). \qquad (2)$$

The utility function u may depend on the planning episode. It captures the supportee's current preferences among a variety of individual attributes of world state, depending on mission goals, as well as his disposition or aversion to risk. Mission goals may include: maximize threat value destroyed (subtractive defence), maximize value of surviving assets (preferential defence), and minimize wastage of missiles. The setting of a vector-valued preference measure on world states may be obtained by generalization.

Starting from a perception of the tactical situation at the current time, there may be infinitely many histories that describe all potential evolutions of the battle world thereafter, each consistent with the current perception and each determining a feasible conditional defence plan. This comprises the set of all potential histories [13]. For example, in Fig. 2, firing actions may be rescheduled in any way that satisfies firing doctrine, illuminator availability and the requirement that potential intercepts occur in the engagement interval. This suggests that the optimization in (1) is, in general, with respect to an infinite set of feasible plans. Obtaining a combinatorial optimization problem over a finite plan space requires narrowing attention to a finite subspace. This procedure, which produces a potentially suboptimal plan, currently proceeds in two steps:

Step I

Generate a finite set A of defence actions, each labelled with a time of execution during the current plan horizon.

Step II

Determine a best plan P^* from the restricted finite set $\Pi(A)$ of feasible plans whose actions occur in A; that is,

$$P^* = arg\,max_{P \in \Pi(A)} v(P). \qquad (3)$$

Two approaches to achieving Step I are to exploit some characterization of defence actions in an optimal plan or to use a heuristic selection procedure. The advantage of the former is that it leads to an optimal plan. The second may be used to reduce computational complexity. A hybrid approach is also possible. An example of the first approach is given in the next section. An example of the second would be to choose actions from several time periods within the plan horizon to achieve defence-in-depth.

Step II rests on an encoding of all members of $\Pi(A)$ as solution trees of a finite AND/OR decision tree T. In addition, P^*, defined by (3), corresponds to an optimal solution tree [14]. T can be explicitly generated as needed via temporal projection. This procedure can be viewed as a form of simulation-based planning [15] in which a search is made through a variety of simulations of plans to select the plan with the most successful simulation. The nodes of T are labelled with a time-stamp and state information on the battle world at that time. Various data structures are also maintained at each generated node to permit enforcement of consumable and contentious resource constraints and firing doctrine in simulations of all histories that emanate from a node. Interior nodes of T are of two types, decision or outcome. A decision node is an OR-node. It corresponds to a specific action in A and the two edges emanating from it represent the choice to perform the action or not. An outcome node is an AND-node. It corresponds to the outcome of a specific previous firing action or of a specific threat strike on an own ship. Such a node has two or more direct successors, depending on the granularity of the stochastic threat destruction and ship damage models used. Emanating edges partition the finite set of potential histories according to the outcome, which is unknown at the time of planning, and they are labelled with outcome probability measures supplied by the damage model. The value $v(P)$ of plan P is recursively computed (a foldback analysis) as a composition of the values of its subtrees, using the values of u on the world states at leaf nodes.

Various combinatorial search algorithms for Step II are being designed and tested. They are currently based on top-down or bottom-up search methods [14, 16, 17]. To reduce the run-times of these algorithms and to permit deeper search in the plan space for a given amount of computing time, versions for implementation on single-instruction multiple-data (SIMD) and multiple-instruction multiple-data (MIMD) parallel computer architectures are also being tested.

In general, determining an optimal conditional plan for the model is an NP-hard problem. An illustration in a naive deterministic setting is obtained as follows. There is one support ship with one fire-control radar to provide HAW guidance. SAMs are perfectly reliable (kill probability of unity) and u is given by the number of engaged threats. Each threat or missile moves at some constant speed along a straight-line trajectory. Then the decision problem of determining whether the fire-control radar can be scheduled to process (engage) all tasks (threats) is NP-complete, even if all tasks have the same release time and two distinct deadlines [18].

An Example

The example of this section leads to a multiplatform generalization of the single platform technique in [5], while also accounting for threat hits on an own ship.

A number of assumptions are made. They include: deployment against the same threat or different threats are independent; fly out time is a monotonically decreasing function of time of intercept; kill probability is a monotonically increasing function of time of intercept; the utility function u is a weighted sum of the number of threats eliminated before the end of the plan horizon - a threat weight reflects the value assigned to its elimination. In addition, we assume that the critical renewable defence resource constraint is due to the contention arising from a limit on the number of time intervals of guidance (illumination) activity from each support ship that can overlap at any time. Illumination from a support ship may be provided by multiple mechanically steered or dedicated electronically steered fire-control radars or by splitting or time-sharing the energy from a single MFR/I according to an energy budget. For HAW guidance, the resource is used (nonpreemptively) just prior to the time of fire until the end of intercept; for a mid-course guided missile, it is used during the terminal phase only.

It can be shown that under these assumptions defence actions in an optimal plan can be chosen from a finite (but potentially large) set characterized by an extremal property. Briefly, this property says that firing actions against each threat should be as late as possible, subject to firing doctrine and resource availability. For example, the plan in Figure 2 is dominated by the one obtained by shifting the illumination intervals and associated firing actions as far to the right as possible. Sequential and parallel algorithms based on enumerative backtrack search and temporal line sweep (backwards in time) are currently being used to identify elements of A for this example.

Toward Time-Dependent Planning

The planning problem and the real-time specification for its solution have been developed in a framework in which the manager is a time-dependent planner [19, 12], capable of varying its deliberation according to time pressure. But the specific planning model presented in this paper would require for its implementation an unbounded rational agent that has the luxury of ignoring the cost of deliberation (cost of time consumed and computational resources committed during deliberation). How can this immense gap be bridged? This section outlines one approach to resolving this problem that is being considered. An interesting feature of the approach is that it is model and machine independent. It may be applicable regardless of the specific algorithmic implementation of the planning function that is adopted in the real-time architecture for the manager.

The underlying idea is to control the complexity of plan generation by using a rolling plan horizon whose size is determined at the time of a service call. Depending on the anytime mode of the planner (contract or interruptible) and on its implementation, this may require computing a plan for each of a deepening sequence of plan horizons within the forecast horizon. The motivation depends on an assumption (which remains to be tested in experiments) that an effective plan of immediate actions should be based on looking ahead over a time horizon that is as large as possible. But the computational complexity of look-ahead is a function of the size of the plan window. This suggests a tradeoff. The approach is illustrated in Fig. 3, where l denotes the size of the plan window. Note that we are not concerned here with the choice of the forecast horizon, this choice having already been made prior to invoking the planning function based on considerations indicated in the real-time specification.

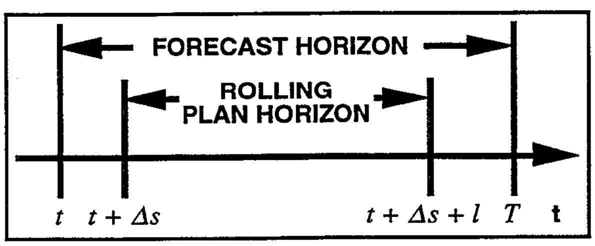

FIGURE 3: Rolling plan horizon.

The approach is reminiscent of one employed in real-time single-agent heuristic search [20] for obtaining a commitment to search moves in constant time. However, there are a number of important differences. Unlike heuristic search where the search process can be naturally limited to a discrete number of levels of look-ahead, our setting requires varying the plan horizon over a time continuum. Also, for a fixed length of plan window, the problem is further complicated by the fact that a varying number of levels of the search tree can fall in the window, depending on the particular battle scenario being processed. Finally, fixing the battle scenario and varying the size of the plan window can lead to disjoint portions of the space of feasible plans being searched.

In the following, M denotes a specific implementation of the planning function. We denote by $\rho(M,c,l)$ the computing time required by M to produce a plan in response to the scenario c (supplied by the projector) over a window of size l. For the kind of implementation we have in mind, it is reasonable to assume that ρ is monotonically increasing in l.

We consider first the case of the planner in contract mode, limited to Δt units of computing time. The planner will maximize l subject to $\rho(M,c,l) \le \Delta t$. It can quickly "solve" this problem with a table look-up if it has precompiled a performance profile table for a variety of sizes of plan window. Since it is clearly impossible to do so for every choice of c, an alternative statistical approach is reasonable. Battle scenarios have a number of natural attributes upon which a partitioning of the input problem space with respect to ρ may be based. Examples include various ASM and SAM parameters, attack profile, force and protected unit parameters, etc. An off-line Bayesian-based statistical technique for attempting this partitioning (class identification and description) is suggested by work in [21]. However, run-time monitoring of the battle world is now needed to make an on-line identification of the specific class C of scenarios confronting the planner when presented with a particular scenario c so that $c \in C$. We can then reformulate the tradeoff analysis of the planner as follows. It introduces the idea of an *almost anytime contract planner* in the sense that its anytime property is modulo a certain level of confidence. Let $0 < r < 1$ be an acceptable level of confidence that the planner will deliver a plan in Δt units of computing time. Then it will maximize l subject to $prob(\rho(M,c,l) \le \Delta t \mid c \in C) \ge r$. As before, this problem can be "solved" with a table look-up if it has precompiled probabilistic performance profile tables for a variety of sizes of plan window and for each scenario class. This is an off-line knowledge compilation process. Neural net technology functioning essentially as an associative memory may be a natural candidate for automating both the on-line scenario identification and table look-up process. An extension of the approach that increases the expected size of the plan window returned is to schedule a number of concurrent instantiations of an almost anytime planner, with confidence levels chosen from a finite strictly decreasing sequence.

Finally, Zilberstein's technique [12] of repeatedly invoking a contract algorithm with exponentially increasing time limits to obtain an interruptible algorithm is readily adapted to our probabilistic setting to produce an *almost anytime interruptible planner* with confidence level r from an almost anytime contract planner that has an equal level of confidence. The overhead of this technique is reflected in a loss of expected size of plan window for a given amount of computing time.

Summary and Ongoing Research

We presented in this paper an agent model for a deliberative planner in a real-time engagement manager. Further details appear in [22]. The model is being used to provide a basis for evaluating several performance aspects, including effectiveness of defence response and the impact of defence and computational resource limitations and of intelligent adaptive control, as well as to assess improvements from using a variety of parallel architectures to implement the manager.

References

1. Naval Forces Update, "Co-operative engagement and self defence," *Jane's Defence Weekly,* pp. 32-34, 19 June, 1993.
2. C. Beal, "Anti-ship missile technology: leaving well enough alone?," *International Defense Review* 10, pp. 957-964, 1992.
3. D. B. Boyer, E. L. Perry, E. L. Price, and S. Godrey, "Force level engagement control in naval anti-air warfare," in *Proceedings of the 1990 Command and Control Research Symposium,* Monterey, California, pp. 204-211, 1990.
4. H. D. Sherali, Y. Lee, and D. D. Boyer, "Models and algorithms for scheduling target illuminators in naval battle group anti-air warfare scenarios," to appear in *Naval Research Logistics Quarterly,* 1993.
5. M. D. Diamond and O. M. Carducci, "Decision processes for large scale resource allocation problems," in *Proceedings of the 8^{th} MIT/ONR Workshop on C^3 Systems,* Cambridge, Massachusetts, pp. 153-160, 1984.
6. M. Diamond, C. Rennolet, J. Kimbel, and M. McNatt, "Optimal resource allocation and optimal launch scheduling for anti-air weapons systems," in *Proceedings of the JDL 1990 Symposium on C^2 Research,* Monterey, California, pp. 194-203, 1990.
7. P. B. Luh, Xi-Yi Miao, S-C. Chang, and D. A. Castanon, "Stochastic task selection and renewable resource allocation," in *Proceedings of the 9^{th} MIT/ONR Workshop on C^3 Systems,* Monterey, California, pp. 59-64, 1986.
8. S. Russell and E. Wefald, "Principles of metareasoning," *Artificial Intelligence*, 49, pp. 361-395, 1991.
9. G. N. Saridis, "Toward the realization of intelligent controls," *Proceedings of the IEEE,* 67(8), pp. 1115-1133, 1979.
10. J. A. H. Miles, E. M. England, H. C. Faulkner, and S. P. Frampton, "Knowledge-based techniques for tactical situation assessment," in *Application of Artificial Intelligence to Command and Control Systems,* ed. C. J. Harris, Peter Peregrinus Ltd., pp. 315-355, 1988.
11. T. J. Laffey, P. A. Cox, J. L. Schmidt, S. M. Kao, and J. Y. Read, "Real-time knowledge-based systems," *AI Magazine* 9(1), pp. 27-45, Spring 1988.
12. S. Zilberstein, "Operational rationality through compilation of anytime algorithms," *Report No. UCB/CSD 93/743,* University of California, Berkeley, California 94720, 1993.
13. Y. Shoham, *Reasoning about Change, Time and Causation from the Standpoint of Artificial Intelligence*, The MIT Press, Cambridge, Massachusetts, 1988.
14. V. Kumar, D. S. Nau, and L. N. Kanal, "A general branch-and-bound formulation for And/Or graph and game tree search," in *Search in Artificial Intelligence,* ed. Kanal and Kumar, Springer-Verlag, pp. 91-130, 1988.
15. D. P. Miller, "A task and resource scheduling system for automated planning," *Annals of Operations Research,* 12, pp. 169-198, 1988.
16. T. Ibaraki, "Generalization of alpha-beta and SSS* search procedures, *Artificial Intelligence,* 29, pp. 73-117.
17. V. Kumar, "A general bottom-up procedure for searching And/Or graphs," in *Proceedings of the National Conference on Artificial Intelligence (AAAI-84),* Austin, Texas, pp. 182-187, 1984.
18. K. I-J. Ho, J. Y-T. Leung, and W-D. Wei, "Complexity of scheduling tasks with time-dependent execution times," to appear in *Operations Research Letters,* 1993.
19. T. Dean and M. Boddy, "An analysis of time-dependent planning," in *Proceedings of the Seventh National Conference on Artificial Intelligence* Minneapolis, Minnesota, pp. 49-54, 1988.
20. R. E. Korf, "Real-time heuristic search," *Artificial Intelligence*, 42, pp. 189-211, 1990.
21. P. Cheeseman, J. Kelly, M. Self, J. Stutz, W. Taylor, and D. Freeman, "AutoClass: A Bayesian classification system," in *Proceedings of the Fifth International Conference on Machine Learning,* pp. 54-64, 1988.
22. B. A. Chalmers, "An adaptive planner in a layered real-time architecture for an engagement manager in a naval threat evaluation and weapon assignment system: An overview," DREV document, 1993.