Operational Planning and Monitoring with Envelopes

Gerald M. Powell*
CECOM Center for C3 Systems
Ft. Monmouth, NJ 07703
powell@cs.umass.edu

Paul R. Cohen**
Computer and Information Science
University of Massachusetts
cohen@cs.umass.edu

ABSTRACT

Until recently computational models of planning have emphasized the generation of plans. Little attention was given to the processes of plan execution, monitoring and replanning. Clearly, when planning in unpredictable and dynamic environments, there is a significant requirement for effective plan monitoring and replanning capabilities. Operational planning is an exemplar of this class of problems. However, to our knowledge very little of the doctrinal or training literature addresses the problems and processes of plan monitoring and replanning. Consequently, the development of a model of operational planning presents a considerable challenge to our current computational understanding of planning. This paper focuses primarily on data structures called *envelopes*, which are critical to the activities of monitoring plans and projecting their progress. Envelopes have been implemented in an AI planner for real-time control of fires forest, called PHOENIX, which generates plans, monitors them during execution, and revises them during execution when they go amiss. The utility of envelopes is illustrated in several examples from the PHOENIX environment and a hypothetical operational planning problem.

Introduction

The principal topic of this paper is the dynamic behavior of military organizations. What happens once an engagement begins? How can we characterize the progress of our forces, viewed at any level of granularity, from platoons all the way up to the highest echelons of command? What messages need to be passed between echelons to ensure coherent behavior across spatial and temporal extents?To solve the problems entailed by these questions, battlefield planners need representations and inference mechanisms to deal with dynamic situations as they evolve. That is, in addition to constructing plans for alternative scenarios, planners need mechanisms to monitor plans as they are executed, to maintain their aims under widely varying circumstances and to modify their plans and goals as appropriate. To our knowledge, the literature on battlefield planning, including doctrine and training texts, contains very little on the problems and processes of plan monitoring and plan revision. Clearly, these are extremely important aspects of battlefield planning. They are addressed in courses at the Command and General Staff College (CGSC), but since there are no formal models of plan monitoring and revision, both pedagogy and knowledge engineering are impeded.

Until recently, AI planning research emphasized the generation of plans. Little attention was paid to the processes of executing and monitoring plans once they were developed [c.f.,1]. An AI planner for real-time control of forest fires, called PHOENIX, which generates plans, monitors them during execution, and revises them when they go amiss is under development. Critical to these activities are data structures called *envelopes*. A detailed description of PHOENIX can be found in [2]; this paper is primarily about envelopes.

The remainder of this paper begins by discussing operational planning, plan monitoring and replanning in terms of doctrine and pedagogy. This is followed by a summary of an analysis of the battlefield planning activities of experienced Army corps planners, and by a discussion of the doctrine and pedagogy associated with this planning task. Next, we summarize the PHOENIX environment and planners, and illustrate the utility of envelopes in several examples. This is followed by a section that discusses some related research. The paper ends with some concluding remarks about the present research and briefly describes further research on the use of envelopes in attempting to provide a computational solution to battlefield plan monitoring.

Maneuver Planning

The particular battlefield planning problem discussed in this paper is the maneuver aspect of operational planning [3,4]. However, the PHOENIX environment is also similar in many ways to other battlefield functional areas (BFAs) of operational planning (e.g., combat service support) [4]. It appears that envelopes, which are

* Presently at Dept. of Computer and Information Science at the University of Massachusetts at Amherst on the Secretary of the Army Research and Study (SARS) Fellowship. ** Presently on sabbatical at Dartmouth College, Hanover, NH. This research was funded by CECOM and SARS Fellowship Program to the first author and; the U.S. Army Research Office (Del. Order 0875), ONR (N00014-88-K-0009) and DARPA (F30602-85-C0014) to the second author.

CH2875-3/90/0000/0024$1.00 © 1990 IEEE

central to plan monitoring and replanning in PHOENIX, could play a similar role in these other BFAs as well. This section briefly discusses maneuver planning as understood in doctrine, and extends that understanding in two areas: plan generation based on our observations of planning activities of experienced corps planners, and plan monitoring and replanning as modelled in an AI planning domain that appears to share many of the characteristics of the maneuver planning problem.

Doctrine

A U. S. Army corps headquarters is responsible for the conduct of war at the operational level. The operational level of war is the link between strategy and tactics, characterized by "the repositioning or displacement of large units ... to force maximum strength against the enemy's weakest point thereby gaining strategic advantage." [3] "The objective of maneuver at corps level is to place or move brigade or division-sized combat elements into positions where they can bring their fires to bear on the enemy with the greatest effectiveness." [5]

Planning at the corps level is a continuous process. At any time, the commander's planning staff will be operating in one of two contexts: planning for a new operation as directed by higher headquarters, or planning operations to react to the changing battlefield situation. In either context, the corps staff must project battlefield situations into the future (approximately 72 hours) and plan based on that projection. A corps plan encompasses actions involving up to 150,000 soldiers and extending for periods longer than a week. With modifications (Fragmentary Orders) the life of a corps plan could be indefinite. In spatial dimensions a corps plan governs operations over a space at least 150 kilometers wide by 150 kilometers deep, but this can also vary greatly due to variations in terrain, expected enemy resistance, and the type of operation the corps is to conduct.

The Plan-Generation Approach. In attempting to solve a particular planning problem, military planners are taught to follow a decision-making process defined by doctrine [3,6]. In general terms, the process attempts to fully specify information relevant to the current problem, identify a number of alternative, incompletely specified plans or solutions, then evaluate these alternatives and complete the selected plan.

Plan Monitoring and Replanning. After the selected course of action is developed into a complete plan, the plan is disseminated to subordinates. The corps continually assesses the battlefield to estimate the impact of the changing situation on the disseminated plan. Once the plan begins execution, certain actions of the plan as well as other conditions of the battlefield situation are monitored to determine if the corps objectives are being achieved in accordance with the plan. Battlefield events may result in the need for the corps to replan, either to avoid failure or to capitalize on unexpected opportunities. It is noteworthy that the principal document describing the doctrinal planning process [3] provides little discussion of the issues or processes involved in plan monitoring or replanning. A forceful statement about the need for a planning methodology which supports plan monitoring and replanning is provided by recent research conducted on these topics at the U.S. Army's CGSC [7].

Beyond Doctrine

In addition to the absence of a doctrinal model for plan monitoring and replanning, there is evidence that human corps planners use a strategy for plan generation which differs markedly from that specified by doctrine [1] [8,9].

Monitoring and Replanning. Although we have little empirical data on plan monitoring and replanning in human planners, we have derived the following characteristics of these aspects of planning from the available data [8,10], the literature associated with the training of battlefield planners [6], and recent research [7] conducted at the U. S. Army's CGSC:

Planning is Opportunistic. Battlefield events may indicate unanticipated opportunities for success, which should be exploited for retaining or regaining the initiative [6,7].

Planning for Contingencies. If the corps has time, it "continuously develops contingency plans for new alternatives to current operations." [6,10]

Planning is Incremental. Execution begins before the plan is fully specified [8]. Many aspects of a plan, such as the exact rendezvous point of forces, cannot be specified until the plan is underway. Thus, planning and action will occur simultaneously. A particularly important criterion is that potential failures in a plan must be detected as early as possible, and plan repairs should be issued before the failure actually occurs [8].

Planning is a kind of Approximate Processing. Lesser and his colleagues [11] proposed the idea of approximate processing for real-time systems. The basis of approximate processing is that there are always several methods to achieve any goal, some of which are

[1] These findings have considerable significance for further knowledge acquisition and development of planning aids in this domain [8,9].

expensive (in terms of time) but give precise results, and others that are cheaper but less precise. Approximate processing in battlefield planning is evidenced by the following statements made by experienced corps planners [8,12]: "We don't know when the attack is coming. If it comes tomorrow, based on what we've done in a period of over an hour or whatever, you could call the division commanders in right now... they [division commanders] can start doing division plans ... if the attack isn't coming for another three months ... this [corps' plan] continuously gets refined." By monitoring and projecting progress, a planner can anticipate how long it has before a plan (or plan-fragment) fails or succeeds. It can use this estimate to select among planning methods.

Planning is Distributed. Battlefield planning is performed at a variety of communicating levels of the Army command hierarchy. For example, a corps communicates to its major subordinates (typically divisions, regiments and brigades) an operations-plan or -order which specifies: a problem to be solved, the available resources, and some guidance on how the problem is to be solved. Planners at division will take the operations order from corps and construct another operations order, from the division's perspective, and communicate this new specification of the problem, the resources, and guidance to the major subordinates of the division. In addition, within each level of the command hierarchy, numerous experts participate in the planning process. Each BFA has its own collection of experts who contribute to the construction of the overall plan by focusing on the requirements of the operations order mainly from their own BFA perspective.

Section **Utility of Envelopes** illustrates how the envelope mechanism supports all these aspects of operational planning. The next section summarizes the PHOENIX environment and planner.

The PHOENIX System

From a high-level design view PHOENIX consists of three elements: a map representation of the world which models ground-cover, elevations, natural and man-made features, and fire-state information, a discrete event simulator that coordinates the fire simulation and agent tasks, and a generalized architecture for fire-fighting agents [2].

PHOENIX simulates fires in Yellowstone National Park, for which we have constructed a representation from Defense Mapping Agency data. Fires spread in irregular shapes, at variable rates, determined by ground-cover, elevation, moisture content, wind velocity, and natural and man-made boundaries. Fires are fought by removing one or more things that keep them burning: fuel, heat and air. Cutting fireline, dropping water and dropping flame retardant removes fuel, heat and air respectively.

In the current PHOENIX sytem, one fireboss directs a few bulldozers (agents), but does not control them completely. Its directions to agents specify in a coarse fashion what to do, but the agents must decide how to interpret these specifications and choose execution methods to satisfy them. PHOENIX is designed to be a testbed for experiments in distributed control, which characterizes operational planning.

PHOENIX agents have an architecture designed for real-time, incremental planning with approximate processing [11,13]. This means that planners begin executing plans before planning is finished and select from among problem solving methods those that give the best results in the allowed time. Because the environment changes as a plan unfolds, agents must be able to monitor, anticipate plan failures, communicate with other agents, and replan. All these activities are enabled by envelopes.

A comparison of the fire fighting and operational planning environments reveals many striking similarities:

- Resources are allocated to control adversarial processes.
- There are multiple processes and multiple resources.
- All processes are ongoing and dynamic.
- There is considerable uncertainty.
- Planning takes place in real time.
- The environment is spatial and involves moving resources from one place to another, which takes time and communication.
- Resources have different capabilities and are best used for different purposes.
- Weather, terrain, and other environmental factors have a profound influence on the probability of success.

In military domains, one must deal with a sentient adversary; whereas one's adversary in fire-fighting is unpredictable, dangerous, and dynamic, but hardly malicious. Despite this difference, the similarities between these domains suggest that the fire fighting environment provides a rich representation for investigating many of the difficult issues characteristic of battlefield planning.

Envelopes

An agent moves through time and over physical distances, through areas with different physical characteristics, and through gradients (of quality and quantity) of information. We call this space the agent's *environment*. The agent also moves through one or more spaces bounded by failure or other important events. These spaces are called *envelopes*. The concept is easier to illustrate than it is to define. Imagine you have one hour to reach a point five miles away, and your maximum speed is 5 mph. If you are late, by even a moment, you fail. As long as you maintain your maximum speed, you are *within your envelope*. The instant your speed drops below 5 mph, you *lose* or *violate* your envelope. This envelope is *narrow*, because it will not accomodate a range of behavior: any deviation from 5 mph is intolerable. The following problem illustrates a wider envelope. You have one hour to travel five miles, as before, but your maximum speed is 10 mph. You start slowly: your average speed is just 3 mph. After 40 minutes you have travelled just two miles, and you have just 20 minutes to travel the other three. This *is* possible: If you travel at maximum speed (10 mph), you will achieve your goal with about a minute to spare. On the other hand, if you continue to travel 3 mph for another 171 seconds, you will fail---you will not be able to cover the prescribed five miles in one hour.

Clearly, if the agent waits 40 minutes to assess its progress, it has waited too long, because an heroic effort will be required to achieve its goal. In PHOENIX, agents check their progress at regular intervals. They check *failure* envelopes, which tell them whether they will absolutely fail to achieve their goals, and they check *warning* envelopes, which tell them that they are in jeopardy of failure. Typically, there is just one failure envelope but many possible warning envelopes. To continue the previous example, you would violate a warning envelope if your average speed drops below 5 mph, because this is the speed you must maintain to achieve your goal. Violating this envelope says, "You can still achieve your goal, but only by doing better than you have up to this point." These concepts are illustrated in Figure 1 . The failure envelope is a line from "30 minutes" to "five miles," since the agent can achieve its goal as long as it has at least 30 minutes to travel five miles. The average speed warning envelope is a line from the origin to the goal, but our agent violated that envelope immediately by travelling at an average speed of 3 mph. In fact, it has moved perilously close to its failure envelope. The box in the upper right illustrates that the agent can construct another envelope from any point in its progress. This new envelope is extremely narrow, as indicated by the distance from its origin to the point at which the failure envelope intersects the x axis.

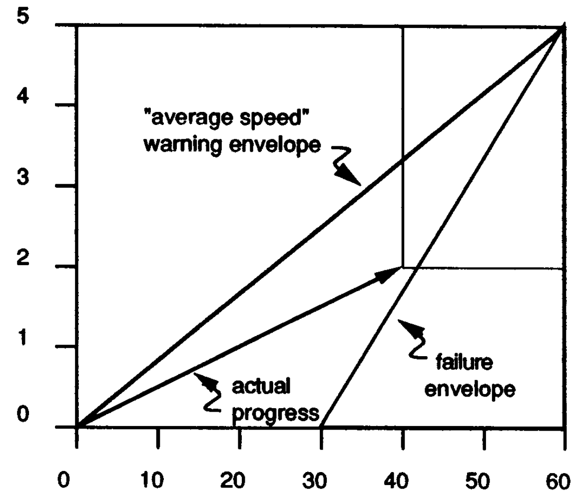

Figure 1 . Depicting actual and projected progress with respect to envelopes

Utility of Envelopes

In operational planning, activities must be coordinated between different levels of a hierarchical command structure, and also between agents at the same level of a hierarchical structure. Envelopes provide a mechanism for both kinds of integration as plans unfold over time. *Agent envelopes* integrate the activities of agents at different levels of a hierarchical structure, while *plan envelopes* integrate agents at the same level of a structure under the jurisdiction of an agent at a higher level. For example, each bulldozer in PHOENIX monitors its own progress with individual agent envelopes, communicating with the fireboss only at points prescribed by the envelopes; and the fireboss maintains one or more plan envelopes that monitor the joint activities of all the bulldozers.

The characteristics of envelopes that make them salient for operational planning are these: First, a planner can represent the progress of its plan by transitions within the plan's envelopes. Progress, failures and potential failures are clearly seen from one's position with respect to envelopes, whereas this information is not apparent from one's position in the environment. Second, just as a planner can project how its actions will propel it through its environment, so it can project how these actions will move it with respect to its envelope. Envelopes function as "early warning" devices: warning envelopes alert the planner to developing problems, and even failure envelopes tell the planner that a plan will fail sometime in the future, so the failure doesn't come as a surprise.

A third advantage is that envelopes integrate agents at different levels of a command hierarchy: An agent P formulates a goal and a corresponding envelope, and gives them to a subordinate agent A_i with the following instructions: "Here is the goal I want you to achieve. I don't care how you do it, and I don't want to hear from you unless you achieve the goal or violate the envelope." A_i then works independently, not monitored by P. If A_i is a bulldozer, it figures out where to go, how to avoid obstacles, and how to keep clear of the fire, until its goal is achieved or its envelope violated. Meanwhile, P is free to think about other agents, other goals, or to replan if necessary. Imagine that P gave A_i a warning envelope, and eventually A_i reports that it is violated. P can now look at A_i's progress within its envelope. By projection P can see when A_i is likely to achieve its goal. How far is A_i from its goal? If it is nearby, the delay might be acceptable. But if A_i still has a long way to go, then it will violate its failure envelope relatively soon. This tells P that it should start formulating an alternative goal for A_i, and approximately when it should redirect A_i, assuming progress doesn't improve. Envelopes grant subordinate agents a kind of autonomy, and grant superordinate agents the opportunity to ignore their subordinates until envelopes are violated.

Agent Envelopes and Plan Envelopes

We distinguish between the envelopes of individual agents and those of multi-agent plans. For example, imagine two bulldozers have been dispatched to surround a fire. Each monitors its own agent envelopes. One progresses nicely, but the other falls behind. Is the plan in trouble? If the faster bulldozer can compensate for the slower one, then the plan may succeed. In PHOENIX, plan envelopes are maintained by the fireboss agent, who coordinates several subordinate bulldozers. Because the environment changes, global plans may be put in jeopardy even if agents are making progress that, from their local perspective, is well within their envelopes. Here is an example we have implemented in PHOENIX: Several bulldozers are dispatched to dig fireline at some distance from the fire; each is given a starting point, an ending point, and a deadline. The fireboss constructs these goals by projecting where the fire will be, and then adding in some "slack time." Typically, the fireboss wants all the bulldozers to have completed their assignments at least one hour before the fire reaches any of their firelines, and the initial plan may allow for several hours of slack time. Plan envelopes monitor this parameter. If the wind speed increases, then the fire will move more quickly, and the slack time will be reduced. In the current implementation, a warning envelope is violated if slack time drops below two hours. The fireboss then replans and typically sends one or more additional bulldozers to help out.

Envelopes in Operational Planning

In this section, we develop an example of envelopes in operational planning and monitoring. Unlike the previous PHOENIX examples, which have been implemented and tested, this example is hypothetical. And, whereas the previous example was concerned with cooperative interactions between agents, this one illustrates both cooperative and competitive interactions.

Envelopment. A familiar tactic, which we will call an *envelopment*, consists of a purposeful delay used to shape a penetration that sets up the conditions for a double envelopment. One's own forces fall back in a parabolic line, drawing the opposing forces into a pocket, where they are surrounded. Imagine an envelopment is underway, and we are monitoring its development to see whether it is working the way we want. What needs to be monitored? The following is a partial list:

- Are our own forces forming a parabolic "pocket" as they fall back?
- Are our own forces distributed in the pocket as we wish?
- Are the opposing forces moving into the pocket as we wish? Or are they moving too far in, and threatening to break through? Are they failing to enter the pocket, not engaging us, slipping away? Are they holding back for some other reason, perhaps to await reinforcements? Are they concentrating on the flank of the pocket, to envelope our flank and mount a counterattack?
- Is our envelopment proceeding as required by our more global strategy?
- Does the opponent have reinforcements within range, or artillery in range, to overwhelm us in the envelopment area?

All these things can be monitored with envelopes. Before we describe how to do it, we need to characterize the envelopment tactic more abstractly and precisely, as shown in Figure 2 .

The first pane in Figure 2 shows an envelopment in progress, with own forces forming a parabola, roughly symmetric around centerline ***c***. The point ***d*** is a coordination point at which 1) the center of gravity of the opposing forces (***cog***) and 2) the onset of the envelopment must intersect in time. In other words, for the envelopment to be successful, ***cog*** must be at ***d***, and the flanks begin to close, simultaneously. If the flanks begin to close too late, or ***cog*** isn't at ***d*** when they begin, the envelopment may fail. In practice, ***d*** is a real area of the battlefield, but it is unknown until shortly before the envelopment begins. This is a common occurrence in real-time planning, where we often plan to achieve states that satisfy particular predicates, without knowing exactly which states they are. In this case, we want ***cog*** to be at a place that satisfies at least these predicates:

- ***cog*** is on the centerline ***c***, or is not "too far" from it, where too far is defined in terms of the distribution of own forces in the pocket, and the exact shape of the pocket.
- ***cog*** is too far into the pocket to escape once the flanks begin to close (assuming that the flanks close without interruption, at the expected rate).
- ***cog*** is not so far into the pocket that the forces at the flanks risk firing on own forces at the rear of the pocket.

These, then, are some of the components of what we mean by "progress" in an envelopment. They partly define an envelope, which we might call the *relative-position* envelope, which measures whether the opposing forces are "moving into position," relative to ours, for us to execute the envelopment successfully. With some additional machinery, described below, we can monitor progress on the three measures above to see how the envelopment is developing. Without knowing exactly where the opposing forces will be when the envelopment begins (***d***), we can nevertheless ask whether they are moving toward such a point or away from it.

To do this, we have to define a few more measures to operationalize the three just described. Consider the first one: ***cog*** is on the centerline ***c***, or is not "too far" from it, where too far is defined in terms of the distribution of own forces in the pocket, and the exact shape of the pocket. One easy operationalization of this involves plotting the trajectory of ***cog*** relative to the trajectory of the focus of the parabola formed by our own forces, abbreviated ***f***. This is shown in the center pane of Figure 2 . In part (b), the distance between ***f*** and ***cog*** is decreasing over time, and in part (c) it is increasing. Clearly, in the former, the opposing forces are moving into a position advantageous to us, and in the latter they are not. We could easily construct more accurate measures, but this simple one---the first derivative of the distance between ***f*** and ***cog***---would probably suffice. For later reference, we'll call this measure the degree of convergence and abbreviate it ***doc***.

Now consider the other criteria for progress in the relative-position envelope: ***cog*** must be far enough inside the pocket to prevent escape, and not so far that the flanks fire on their own forces at the back of the pocket. The former measure depends on the ***cog***'s feasible rate of movement. Fast opposing forces must be deeper in the pocket than slow forces. Once again, we can define a measure to determine whether we are moving towards a successful envelopment or away from one. This measure should probably include more parameters than the last one:

- l_l -- the distance between the endpoints of the flanks on the line denoted ***l*** in Figure 2
- v_{own} -- the maximum velocity of own forces (specifically, those on the flanks)
- $D_{l,cog}$ -- the distance between ***cog*** and the nearest point on line ***l***
- v_{cog} -- the maximum velocity of ***cog*** (more specifically, the maximum velocity out of the pocket.)

With these parameters, we can define a measure to operationalize "far enough into the pocket that they can't escape." Clearly, this measure depends on the rate at which we can close the pocket, determined by l_l and v_{own}, and the ability of the opponent to retreat out of the pocket, determined by $D_{l,cog}$ and v_{cog}. As before, we can make this measure arbitrarily complicated by adding more and more parameters, but these may suffice.

Lastly, we consider the criterion that we shouldn't fire on our own forces. This is a simple measure that depends on the angle of the parabola and the position of ***cog*** within it. Specifically, we are concerned with the

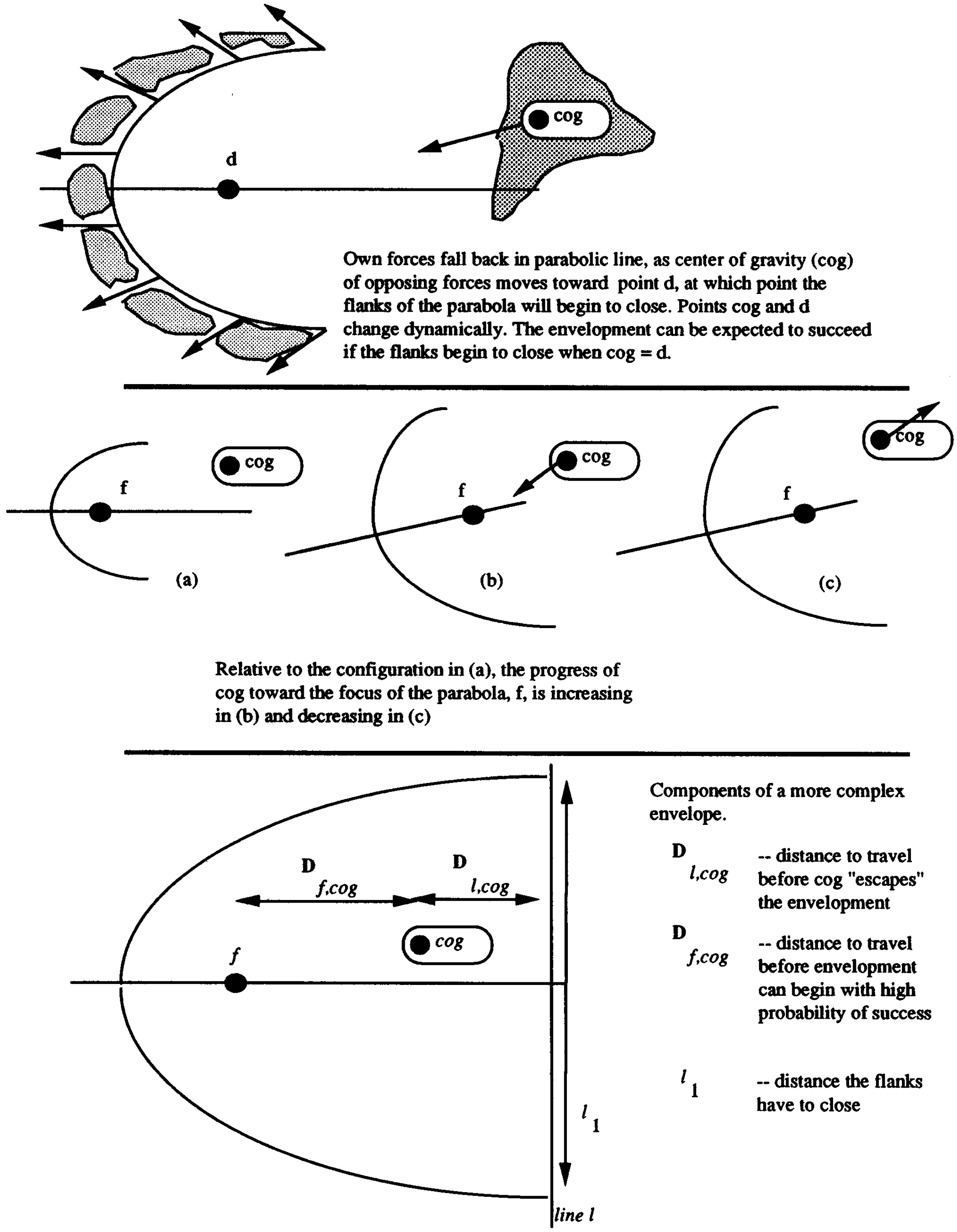

Figure 2. Depiction of the use of envelopes in an envelopment

position of ***cog*** on the axis ***c***. Let's define a measure called ***depth***$_c$ as follows: ***depth***$_c$ = $D_{l,cog} / D_{f,cog}$ (see the third pane of Fig.2). Thus, if ***depth***$_c > 1$, ***cog*** is "more than halfway" into the pocket, and, conversely, if ***depth***$_c < 1$, ***cog*** is relatively near the exit of the pocket. Clearly, we want ***depth***$_c$ to be large enough that ***cog*** can't escape, but small enough that we don't fire on our own forces.

Two points about this analysis must be stressed. First, all these measures are *dynamic*. Consider degree of convergence (***doc***). If ***doc*** is negative it means that the distance between ***f*** and ***cog*** is decreasing, and ***cog*** is moving into the pocket. If ***doc*** > 0 it means ***cog*** is moving away from ***f***, and perhaps out of the pocket. ***doc*** itself can be constant, but is more likely to be dynamic, given that ***cog*** and ***f*** are simultaneously changing (because both opponent and own forces are moving).

Second, the conditions for a tactic, such as an envelopment, are typically specified by *several* measures like ***doc*** or ***depth***$_c$, not just one. For example, the relative-position envelope involves ***doc*** and ***depth***$_c$, which tells us both whether the opponent can escape, and whether we risk firing on our own forces. There is therefore a *probable success* envelope of any plan, such as an envelopment, in which it is likely to succeed (in fact, there are numerous probable success envelopes for a given plan, corresponding not only to the conditions under which success is probable, but also to different probability levels). The probable success envelope is a subspace of the space defined by the parameters described earlier: ***doc***, ***depth***$_c$, ***cog***, ***f***, l_l, v_{own}, $D_{l,cog}$, $D_{f,cog}$, v_{cog} and the angle of the parabola (approximated by l_l and the length of ***c***). It is up to the planner to specify the envelope for a plan. Because it is difficult to specify a complete function relating all these parameters and measures, we would expect the envelope to be specified in *tabular* form, perhaps in a set of rules, based on subsets of the parameters and measures. For example, the planner may stipulate that the envelopment is likely to succeed only when these conjunctive conditions hold:

- ***doc*** < 0 (the opponent is moving into the pocket)
- $l_l > 2$ * length(***c***) (flanks are unlikely to fire on center)
- $v_{own} > v_{cog}$ (own forces can outrun opponent)
- $0.6 <$ ***depth***$_c < 1.0$ (***cog*** is at least partway into the pocket and at most halfway in)

In fact, the planner can stipulate numerous rules of this kind for when the envelopment is likely to succeed.

Related Work

Sanborn and Hendler [14] have developed a system called CROS which uses functional objects called monitors to track aspects of dynamic worlds. The differences between CROS monitors and PHOENIX envelopes appear to be the following. First, in CROS, monitoring is the ongoing independent element of an overall reasoning system that provides a representation of the aspects of the world that are important to the survival and goal-directed behavior of a robot. "Monitoring is an ongoing, independent task driven by events taking place in the world, as opposed to an agents internal goals." [14] In contrast, PHOENIX envelopes are created at the time an action or plan is instantiated. Thus, envelopes are closely associated with plan generation. In addition to monitoring the actions and plans of agents, envelopes are used to monitor features of the environment such as the relation between the actual location of the fire at various points in time and the projected location of the fire for some specified future point in time. Second, in CROS, monitors trigger robot actions based on projected failures. In PHOENIX, agents check failure envelopes which tell them whether they will absolutely fail to achieve their goals, and warning envelopes which tell them they are in jeopardy of failure. An agent may or may not take action when it recognizes the violation of a warning envelope. Thus, unlike CROS, envelope violations in PHOENIX provide the agent with an opportunity to consider options. For example, the agent may choose to: "wait and see" how the situation unfolds before taking further action with respect to the envelope violation; begin reasoning about alternative plans to deal with plan failure; increase monitoring activities in order to be able to respond in a more timely fashion should the violation of the warning envelope persist. This difference in monitoring between CROS and PHOENIX appears to be due to differences in the problem domains these systems address. Although PHOENIX has problems that appear as time-critical (e.g., a bulldozer avoiding being burned by the fire) as the problem faced by CROS, some of the problems that confront PHOENIX agents are less time-critical and, therefore, are amenable to a less reactive approach. Third, in PHOENIX, envelopes (see **Utility of Envelopes** section) allow the agent to monitor progress which may, in some cases, be unexpectedly good. This use of envelopes provides the agent with a basis for capitalizing on unexpected opportunities; this characteristic of monitoring is absent in CROS. Fourth, in CROS, there is a single agent working toward a goal. In PHOENIX, there are typically multiple agents acting simultaneously and toward a common objective. When PHOENIX agents act as part of a multi-agent plan, their joint activities are monitored by plan envelopes which integrate agents who are at the same level of the organizational structure; plan envelopes are used for this purpose by an agent at a higher level in the organization.

Murphy [15] has presented a specification of requirements, and a design description, for an execution monitoring system in Army corps maneuver planning. The objectives of that work on execution monitoring appear similar, in many cases, to those described in the present paper. However, it is difficult to compare the two efforts because the description given in [15] provides little information on how to achieve the desired functionality.

Concluding Remarks

The decision to examine battlefield plan monitoring and replanning within the framework of an existing (and evolving) computational planner was motivated by the apparent lack of a formal model for these problems and processes and by the striking similarities between the PHOENIX problem domain and the battlefield planning problem.

The utility of envelopes for minimizing the interactions between the planner and its agents, and for monitoring plans and projecting their progress is clearly evident in our examples. Envelopes also provide a basis for further knowledge acquisition on plan monitoring issues such as inter-agent communication (what needs to be communicated, to whom, and why). Envelopes may also be useful as an instructional tool for training battlefield planners on the strategies for selecting, applying and monitoring complex tactical maneuvers and overall operations plans.

References

[1] *Proceedings of the DARPA Knowledge-based Planning Workshop*, Austin, Tx, December 1987

[2] Cohen, P.R., Greenberg, M.L., Hart, D.M. and Howe, A.E. Trial by fire: Understanding the design requirements for agents in complex environments. To appear in *AI Magazine*, Fall 1989

[3] FM 101-5 Staff Organization and Operations, *HQ, Department of the Army*,1984

[4] Friedman, D., Maguire, V., Miller, D., Plotkin, N. and Ottenberg, M. System Description for the Army Tactical Command and Control System, The Mitre Corporation, MTR-88W00081, August 1988

[5] FM 100-15 Corps Operations, HQ, Department of the Army, November 1988

[6] ST 100-9 The Command Estimate, *U.S. Army Command and General Staff College, Ft. Leavenworth, KS*, July 1988

[7] Long, C.L. Synchronization of combat power at the task force level: Defining a planning methodology. *Thesis, U.S. Army Command and General Staff College, Ft. Leavenworth, KS*, June 1989

[8] Powell, G.M. and Schmidt, C.F. A first-order computational model of human operational planning, *Technical Report, Command and Control Systems Development Branch, Ft. Monmouth NJ*, August 1988

[9] Powell, G.M. and Schmidt, C.F. Human corps planning: A first-order computational model. *AI Systems in Government Conference Proceedings*, George Washington University, March 1989

[10] Powell, G.M. and Loberg, G. Operational Planning: A Status Report on Knowledge Acquisition with the U. S. Army War College, *Technical Report, Center for Command, Control and Communications Systems, Ft. Monmouth NJ*, October 1987

[11] Lesser, V. R., Durfee, E. H., and Pavlin, J. Approximate processing in real-time problem solving. *AI Magazine*, Spring 1988

[12] U.S. Army War College, Simulated Planning Exercise, Carlisle Barracks PA, 1986

[13] Durfee, E.H. and Lesser, V.R. Using partial global plans to coordinate distributed problem solvers. *Technical Report 87-6, Dept. of Computer and Information Science, University of Massachusetts, Amherst MA*, 1987

[14] Sanborn, J.C. and Hendler, J.A. Dynamic reaction: controlling behavior in dynamic domains. *International Journal of AI & Engineering*, 3(2), April 1988

[15] Murphy, D. Execution Monitoring: Requirements and Design Description. (Airland Battle Management Program). Advanced Decision Systems, Mountain View, CA, July 1989