Minefield Overwatch Using Moving Target Indicator Radar

Anthony Donadio, Robert Ewing, William Kenneally,
John J. SantaPietro

The MITRE Corporation
145 Wyckoff Road
Eatontown, NJ 07724

ABSTRACT

Traditional antipersonnel (AP) land mines are an effective military tool, but they are unable to distinguish friend from foe, or civilian from military personnel. The concept described here uses an advanced moving target indicator (MTI) radar to scan the minefield in order to detect movement towards or within the minefield, coupled with visual identification by a human operator and a communication link for command and control. Selected mines in the minefield can then be activated by means of the command link. In order to demonstrate this concept, a three dimensional, interactive simulation has been developed. This simulation builds on previous work by integrating a detailed analytical model of an MTI radar. This model has been tailored to the specific application of detection of slowly moving dismounted entities immersed in ground clutter. The model incorporates the effects of internal scatterer motion and antenna scanning modulation in order to provide a realistic representation of the detection problem in this environment. The angle information on the MTI target detections is then passed to a virtual three dimensional sight which cues a human operator to the target location. In addition, radar propagation effects and an experimental design in which the radar itself is used as a command link are explored.

1. THE M2MTI SYSTEM CONCEPT FOR APLA

1.1. Background

The problem of accidentally tripping anti-personnel (AP) mines is one which has received substantial attention in recent years. While AP land mines have long been recognized as an important tool for area denial and perimeter control, they are unable to distinguish friend from foe, or civilian from military target. They also remain capable of releasing destructive power long after the military mission is accomplished and the civilian population has returned to the area. The presence of harmful explosive material still remaining after the life of the mission raises concerns about accidental detonation should the triggering mechanism safeguards fail or be inadvertently defeated. These characteristics have motivated the Anti Personnel Land mine Alternative (APLA) program. The APLA concept consist of a sensor within the munition, coupled with a remote controller. The sensor determines intrusion, while a human operator performs munition field overwatch and operates a controller console. The operator provides alert signals when an intruder is detected by the sensor, determines the intent of the intrusion, and controls a fire command to the munition.

One interesting aspect of this concept is that it requires a human external to the minefield, equipped with an optical device to identify the target. Since the human is in essence an external sensor, it seems reasonable to consider separation of the sensing functions of the land mine from the ordnance - external sensors in addition to the human. An example of the external sensor concept is one that uses an advanced moving target indicator (MTI) radar to scan the minefield and detect movement toward or within the minefield. This concept is called minefield monitoring MTI radar (M2MTI). MTI personnel detection radars have been deployed within the U.S. Army for years and have been designed to detect walking as well as crawling soldiers. Advances in signal processing have enabled performance to be significantly improved over deployed systems.

Given registration of the mines with the radar, the radar controller will generally have the advantage of detection of movement outside of the minefield and should in fact be able to detect the likely point of entry into the minefield. This ability to detect movement toward the minefield means that one does not have to wait until mines have been tripped to identify and warn civilians away from the minefield. In the standard APLA design approach, it is necessary that a mine be tripped before it can be command-detonated. However, the overwatch of the MTI radar will permit a "minefield warning" to be initiated automatically when a person approaches the minefield. When the "moving target" gets near the minefield, the controller can issue an audible challenge using the radar as a communication link to small speakers at the edge of the minefield. Depending on the response of the "target", selected mines in the minefield can then be activated using the radar signal as the command link. This approach is illustrated in Figure 1.1

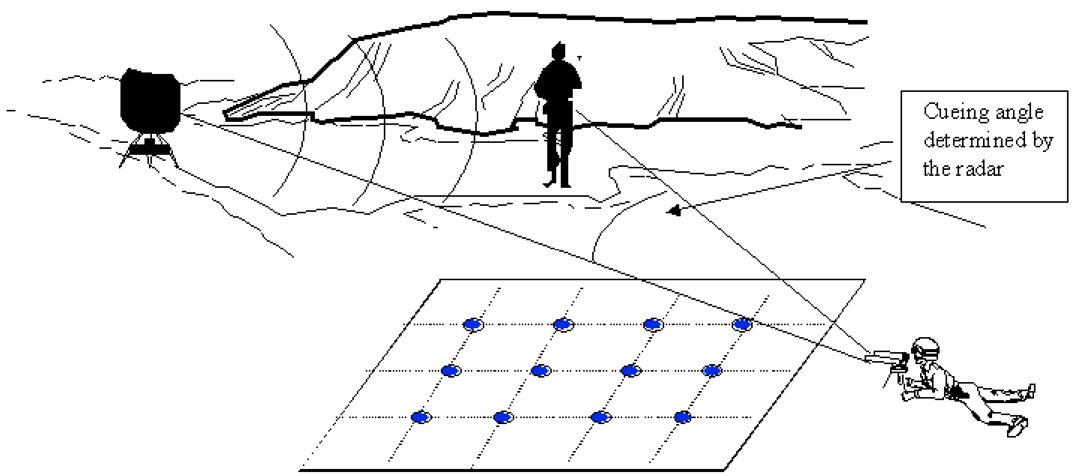

Figure 1-1 : Operational Architecture for M2MTI/APLA Concept

An advantage of this approach is that it provides a directional link at the radar for communication at lower transmitter power levels, while reducing interference with adjacent minefields by virtue of its relatively narrow directional beam. In fact, this concept can be extended to essentially eliminate the need for the mine to a have separate active transmitter by using a passive modulator in the mine to simply modulate the incoming radar signal to provide the required unique signature to complete the C2 link [1]. The issue of passive communication with the radar monitor is addressed by the observation that modulated reflectors may be used to conduct passive communications with modified Army ground surveillance radars. As a preliminary demonstration, MITRE has configured a breadboard passive modulated reflector using a simple continuous wave (CW) Doppler radar and a tone modulated diode reflector. This offers significant advantages relative to battery power requirements as the power required is just that needed to switch a small set of pin diodes in order to provide the mine with some gain in the general direction of the radar.

1.2. MTI Radar Technology

MTI radar operates by detecting the Doppler shift of a transmitted signal from a moving target. The Doppler shift experienced by a signal transmitted at wavelength λ due to reflection from a target moving at speed v is given by

$$f_d = \frac{2\mathrm{v}}{\lambda}\cos\theta \tag{1}$$

where θ is the angle formed between the radar line of sight and the target velocity vector.

There are a number of factors which tend to interfere with detection of moving ground targets by an MTI radar, the most severe being the presence of ground clutter. There are several effects which will serve to spread (in the frequency domain) the clutter seen by an MTI radar. These include radar platform motion, internal scatterer motion (e.g. wind-driven foliage) and antenna scanning modulation. These factors tend to make detection of moving targets, particularly slowly moving targets, a challenging task since this clutter spread can readily mask such targets. Modern sophisticated signal processing can overcome this difficulty. In order to enhance the received target signals, a number of pulses are collected into a coherent pulse interval (CPI) and coherently integrated using a fast Fourier transform (FFT). To further reduce the occurrence of false alarms, a number of clutter cells centered around zero Doppler can be excised. Following this, there is a detection process in which a threshold is constructed to determines the candidates for target processing. Postprocessing is performed to enhance the probability of correct detection, and the result is passed along for display.

MTI personnel detection radars have been deployed within the Army for years and have been designed to detect walking as well as crawling soldiers. Advances in signal processing have enabled significantly improved performance in recent systems. For example, a modern MTI radar system such as the Motorola AN/PPS-25, is credited with being able to detect a walking man at 5 kilometers. The required operational range for M2MTI is more likely to be on the order of a kilometer or less with attendant reductions in required radiated power. The AN/PPS-25 has a target location accuracy of 15 meters (RMS) in range and 0.6 degrees in azimuth. At a range of 700 meters it should be able to locate targets to within less than 20 meters in its present configuration. Azimuth accuracy could be improved at an increase in complexity by using monopulse antenna techniques. Another example of this class of radar is the man-portable Squire radar by MSSC, Inc. The Squire uses a frequency-modulated continuous wave (FMCW) transmitted waveform which has low radiated power and is difficult for a hostile receiver to intercept. The use of FMCW also has other advantages related to the use of the radar as a passive communications link as will be discussed later.

1.3. Minefield Communications Link

In traditional minefield doctrine, an over-watch sentry is posted to observe the minefield in order to deter and delay enemy breaching attempts by calling in direct or indirect weapons fire. Current APLA concepts require a man-in-the-loop to activate a munition. This concept requires a communications link between the man, in this case the minefield overwatch observer/controller or sentry and the munitions. The munition can signal that an intruder has triggered sensors. The sentry can observe and verify hostile presence before activating a munition. The control link can be wired or wireless or it may use a hybrid of complementary technologies for redundancy. Wired links are simple, low cost, and reliable. They are immune to jamming and enable remote powering or battery charging. Wire is ideal in long-term strategic barriers, such as the Korean demilitarized zone (DMZ) which has had minefields for decades. Wire imposes a logistics burden for tactical use. It is time-consuming to lay and difficult to conceal. Intruders or artillery fire may cut it. Conventional radio technology can also be used for two-way mine control links. The main issues are battery life, jamming vulnerability and cost. The intruder-jammer generally has a range ratio advantage over the controller with respect to the munition control link receiver.

If radar is used as an overwatch sensor, it may be modified to enable it to also function as a communications link. Since radar might not be used to overwatch all minefields, radar-based communications would be secondary to other wired or wireless means. The benefit of having a radar communications mode is redundancy. Communicating from the radar to the munition is straightforward. The radar transmitter would be suitably modulated so that a simple receiver on the munition could process the message. If the radar is pulsed, the munition receiver need only consist of a simple diode video detector and a high gain operational amplifier circuit. The munition's microcontroller could have a software subroutine to interpret changes in pulse timing, (pulse repetition rate or pulse position modulation). Communicating from the munition to the control link is not straightforward. Simply relying on the primary communications link, wired or wireless, does not provide redundancy to return link for status messages and acknowledgements. Conventional radar transponders use an active transmitter on the radar frequency. That would be too expensive for munitions. Semi-active transponders amplify and return radar signals using monolithic microwave integrated circuit gain blocks and delay lines with antenna s switches. The DARPA RF tag program is developing such devices. These are cheaper than active transponders, but are probably not cheap enough.

Passive modulated reflectors are devices that effectively change radar cross section in a manner that can be detected at the radar. In the simplest form, a PIN diode connected to an antenna can be biased with a modulating signal from the munition microcontroller. The impedance of the diode will change with the signal and this change will affect the energy returned to radar that is illuminating the munition. The communications range is a function of the radar power and the antenna gains at the radar and passive modulated reflector. Ranges on the order of kilometers have been demonstrated.

2.THE M2MTI/APLA SIMULATION

2.1. Simulation Architecture

A detailed model of the system concept described in section 1 has been developed by MITRE and is used as a tool for concept development and scenario evaluation. This model builds upon core simulation components developed by MITRE and reported on elsewhere [2]. Figure 2-1 shows a block diagram of the architecture of this model.

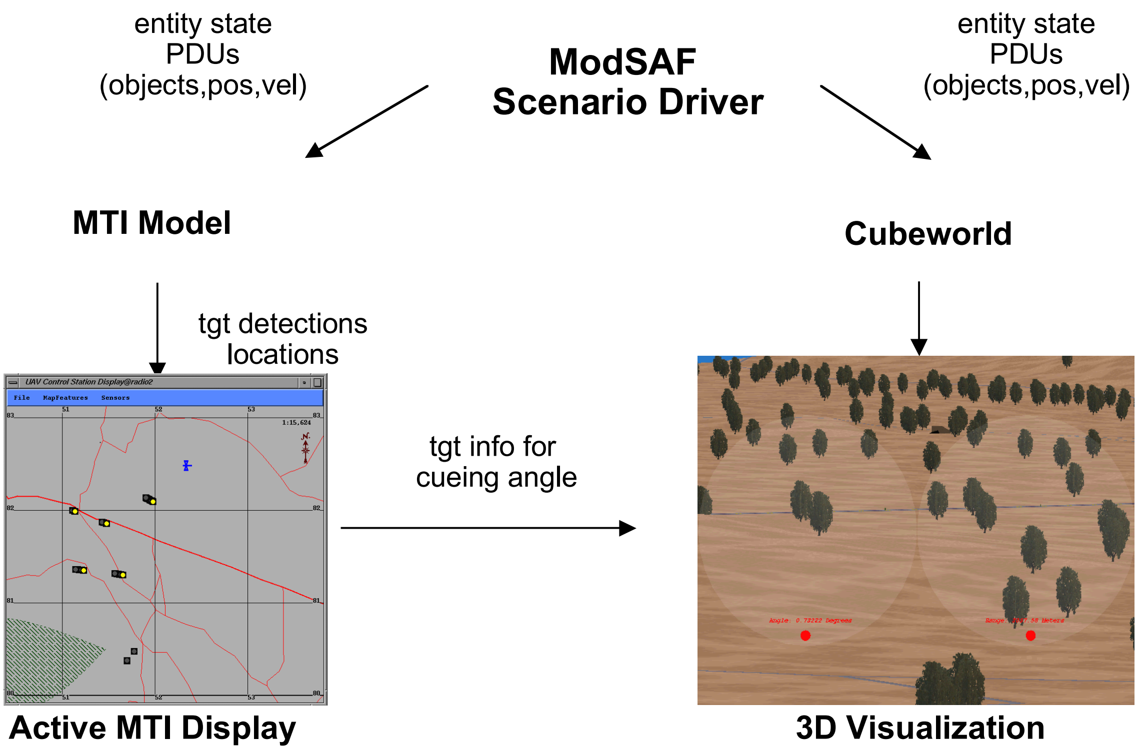

Figure 2-1: M2MTI/APLA Simulation Architecture

The architecture is composed of three components: a scenario driver, an M2MTI radar model, and a three-dimensional visualization tool called Cubeworld. These components run on separate computer platforms which are nodes on a network. ModSAF is used as the scenario driver. Simulation scenarios are generated within ModSAF, and for this application will include dismounted entities moving within the terrain database and deployed minefields. The protocol data units (PDUs) from the scenario are sent to both the MTI model and to Cubeworld. The detection reports from the M2MTI model appear on a tactical map display while the Cubeworld display has the form of a simulated sight which is controlled by a joystick. The M2MTI display indicates detections and tracks, and allows a user to cue the Cubeworld view to any designated track by a point and click operation. . This operation sends to Cubeworld the necessary information to allow it to compute a viewing angle and to cue an operator towards the designated track by a pair of flashing dots in the lower corners of the Cubeworld display. The error signal (difference between the actual sight angle and the cueing angle) activates one of a pair of red dots in the lower corners of the sight. The flashing indicates the slew direction for the operator. Solid red indicates that the designated target is within the field of view. The two simulation components shown in Figure 2-1 will be described in more detail below.

2.2. M2MTI Radar Model

The M2MTI radar model is based on the MTI processing model (MPM) that was designed for the multisensor airborne surveillance platform (ASP) simulation and reported on in detail in [2]. A block diagram of the process is shown in Figure 2-2.

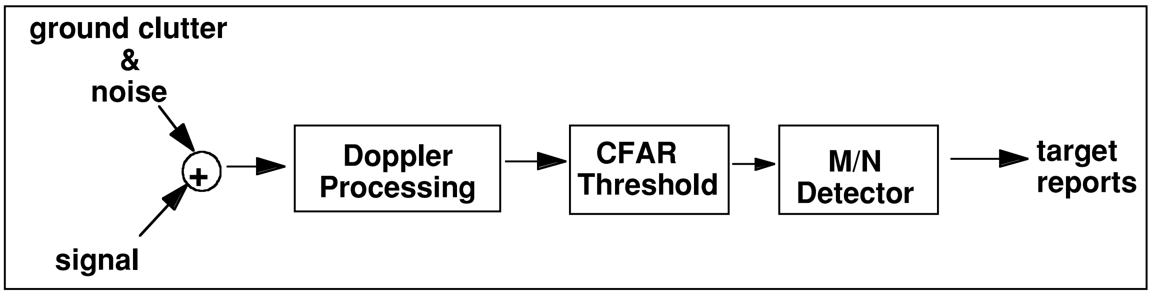

Figure 2-2: The MTI Radar Processing Model

The MPM is very flexible and can be attached to a ground entity as well as an airborne entity. However, the operational scenario of minefield overwatch dictates a substantially different environment, particularly with regard to detection of walking or crawling humans. Therefore a modified clutter model is required. As described in 2.1 above, the most severe impediment to target detection for an MTI radar is the presence of ground clutter. Complete and accurate modeling of ground clutter as seen by an radar is a formidable task. In this effort we are striving for medium-level fidelity and require that the radar model output give a realistic feel for the level of target detection and "drop-out" that will occur over the terrain in the database. As such, a band-limited Gaussian model is satisfactory for representing continuous terrain viewed from a radar at moderate grazing angle. Clutter is represented as a complex Gaussian (Rayleigh amplitude, uniform phase) random process. Because of the effects described above, ground clutter as viewed by a radar cannot be realistically treated as uncorrelated. There are several effects which will serve to correlate or spread the clutter for an MTI radar. The ones considered here are internal scatterer motion due to wind and antenna scanning modulation. These effects will correlate the clutter from pulse to pulse. Viewed in the frequency domain, this introduces a finite bandwidth to the clutter spectrum. This bandwidth can be estimated by means of known relationships. According to [3], the clutter spread σ_w from wind and the spread σ_{sc} from scanning are given by

$$\sigma_w = 4.4\times10^{-5}\left(\frac{f_g v_w^{1.261}}{P} \right)$$
$$\sigma_{sc} = 0.2652\left(\frac{\alpha}{P\beta} \right) \quad (2)$$

where f_g is the center frequency in gigahertz, v_w is the wind speed in knots, P is the radar pulse repetition frequency (PRF) in kilohertz, α is the antenna scan rate in radians/second and β is the radar azimuth beamwidth in radians. Use of (2) along with an assumed total clutter spread given by

$$\sigma_{total}^2 = \sigma_w^2 + \sigma_{sc}^2 \quad (3)$$

will give an estimated clutter spread for a simulated radar mission. A band-limited Gaussian process whose support approximates σ_{total} can be generated by passing an uncorrelated complex Gaussian white noise process W through a linear filter whose frequency response approximates the low pass response of bandwidth σ_{total} centered at DC. The shaping filter used for this purpose is a 2nd order recursive filter with frequency response of the form

$$H\left(z^{-1}\right) = \frac{b_0 z^{-2} + b_1 z^{-1} + b_2}{z^{-2} + a_1 z^{-1} + a_2} \qquad (4)$$

The final clutter power is adjusted by appropriate scaling of W. The total clutter cross-section is obtained by use of a reflectivity coefficient σ_o multiplied by the area of a clutter cell which is determined by the range, range resolution and radar beamwidth. According to equations (2) – (4), the clutter process will depend on σ_{total} through the parameters f_g,P,v_w, α and β. An inspection of the range of variability for these parameters leads to the observation that for the purposes of this application, attention can be restricted to the case where $10^{-4} \leq \sigma_{total} \leq 10^{-1}$. This interval is then partitioned into nine subintervals, and nine sets of coefficients are generated off-line for the filter in (4). The real-time program then uses the calculated value of σ_{total} as an index into the table containing the filter coefficients. σ_o is modeled as a function of f_g over the range $8 \leq f_g \leq 40$ (X to Ka band) using scattering data from [4]. The radar model is set up so that the parameters f_g and P are user adjustable through the graphical user interface (GUI). α and β reside in a text file which can be independently edited, and v_w is generated by ModSAF the scenario driver.

The detection process uses the so-called "cell averaging CFAR" algorithm. This process uses a number of cells on either side of the target and takes the averaged returned envelopes to form an adaptive threshold. The final stage in the detection process is a "binary integration" of a number of CPIs. Thus, a target is declared in a particular Doppler bin if its value there crosses the threshold m out of n times. In this case, n = 6 and m = 3 i.e. 6 CPIs are collected, and a target is declared at a particular Doppler bin if its value has exceeded the threshold on at least 3 out of the 6 CPIs. The resulting detections are then correlated with the appropriate ModSAF entity and marked on an separate x-y grid which is overlaid on the ModSAF PVD and is labeled in terms of the ModSAF topocentric co-ordinates (TCC). TCC corresponds to the UTM grid offset by a base-point.

2.3. The Visualization Model

The visualization component of the APLA/Minefield simulation interacts with the other elements of the simulation by exchanging PDUs over a DIS simulation network, and allows for both 3D viewing and user interaction with the simulated environment. It is implemented using ***Cubeworld***, a Mitre-developed prototype Virtual Reality system designed to model, manipulate and render graphical objects in a three dimensional virtual environment. Cubeworld is an object-oriented system (implemented in C++) which uses OpenGL for rendering. Using Cubeworld as our visualization tool allowed us to leverage a substantial amount of modeling and simulation work that had already been performed on other tasks, including an M16 rifle model equipped with a simulated CIDDS device. (*CIDDS*, or *C*ombat *ID* for the *D*ismounted *S*oldier, is a "friend or foe" identification system for dismounted infantry, in which a coded laser interrogation generates an RF response identifying the recipient as a "friendly.")

2.3.1. General Cubeworld Object Model

Cubeworld's basic object model involves representing the position and orientation of an object in the virtual world using a 4x4 homogeneous coordinate "position" matrix, which takes the simulated environment's global or "world" coordinate system to the origin of the object's local or "model" coordinate system. (Homogeneous coordinates add an extra coordinate to the matrix representation of each point to allow translations to be treated as multiplications, and thus composed with rotations and scalings [5].) This makes manipulating the virtual object a matter of multiplying its position matrix by matrices that implement an appropriate set of rotations, translations, and scalings. The actual geometry of an object in Cubeworld can be modeled in a number of different ways. In our simulation, we focused primarily on using models defined in Multigen's OpenFlight graphics format. When such an object is instantiated, the OpenFlight file representing that object type is read into an internal scene graph structure, from which the object is rendered during the rendering cycle. Cubeworld also provides the capability for stereographic 3D rendering of the virtual environment, using Crystal Eyes 3D glasses.

DIS entities being controlled by other simulators are represented in Cubeworld using a subclass of the FltModel class called *DISEntity*. The *DISEntity* class includes DIS specific data members (such as Entity ID and Type) for associating the Cubeworld VR entity with the DIS simulation entity that it represents, and methods to allow it to respond to basic DIS PDUs. Functions implemented by these methods include translating coordinates from the DIS WGS84 coordinate system to and from Cubeworld's VR world coordinate system, moving the object in response to Entity State PDUs, responding to Fire and Detonation PDUs, and performing dead reckoning. DIS packets are read and dispatched to each entity by a DIS interface subsystem. This reads PDUs from the simulation network, identifies (or if necessary, creates) the corresponding cubeworld VR entity, and then delivers the packet data to the object by calling the appropriate PDU handler method. Given this basic object framework, objects with simulation specific behaviors can be created by creating subclasses of the DISEntity class object (FltModel) class that includes methods to implement these behaviors.

2.3.2. Visualization Scenario

The operational scenario simulated is as follows. An overwatch operator (typically a dismounted infantryman equipped with a CIDDS equipment system) is placed in position to overwatch a minefield and the surrounding area. Another operator with a ground-based MTI radar mounted on a HUMMV is also placed in a different overwatch position, and is equipped with a display that shows MTI detections as dots on a tactical map of the area. The MTI operator can designate a map location to the overwatch operator via a mouse-click, which causes a signal containing the terrain coordinates of the detection to be transmitted to the latter's computer system. The overwatch operator's system responds by calculating the range and azimuth to the detection given his current orientation, and then cues him to turn in that direction with a flashing light on an eyepiece display. Once he is facing in the proper direction, the overwatch operator can use binoculars, rifle scope, and CIDDS system to locate the target and perform a threat assessment.

The simulation implements this scenario as follows. The DI overwatch operator is modeled in Cubeworld as a user-controlled *DIS_ControlledQ_DI* object (see below), from whose viewpoint the visual simulation is rendered. The MTI station is modeled as a ModSAF entity (typically a HUMMV), which the MTI simulator assumes is equipped with a ground based MTI radar. Friendly and enemy DIs are also placed near the minefield and given movement tasks. As the simulation runs, the MTI simulator looks for entity state PDUs from that entity, simulates the returns from a ground-based MTI radar at that location, and generates simulated MTI detections for them. These are placed on the simulation network in the form of "detection PDUs," which in turn are read by the MTI Display simulator and shown as dots on a ModSAF-style tactical map. Mouseclicking on the map in turn generates a custom DIS Action Request "cueing" PDU containing the terrain coordinates (in lat/long format) of the detection point. This PDU is received by Cubeworld's DIS interface, and dispatched for processing to the *DIS_ControlledQ_DI* overwatch entity, which calculates range and azimuth to the cue point and and displays the appropriate cue on a head's up display.

2.3.3. Simulation Class Structure

DIS_Controlled_Entity Classes. The Cubeworld overwatch object is modeled using a special set of classes based on the *DIS_Controlled_Entity* class. This is a subclass of DISEntity whose methods have been overridden to receive inputs from a control device (such as a flybox), and to transmit DIS PDUs rather than receive them. *DIS_ControlledQ* is a further subclass of *DIS_Controlled_Entity*, and contains and maintains data and methods to create a head's up display (a *DIS_HUD* object, see below) that can receive and process cueing PDUs. It also updates the *DIS_HUD*'s cueing indicators in response to movement on the part of the overwatch operator. *DIS_ControlledQ* is further subclassed into the *DIS_ControlledQ_DI* class, which implements specific versions of each of these methods appropriate to a dismounted infantryman.

Control Device Classes. The *DIS_Controlled_Entity* class creates and includes a pointer to a created *DIS_Controller object*, which in turn includes a *Control_Device* object. The *Control_Device* class creates an interface for methods to read discrete and analog actuator values from a control device, such as a flybox stick, levers and buttons. These methods are implemented in turn in subclasses such as *Control_Device_Flybox*. The *DIS_Controller* classes define methods to get movement and other control related data for the device; examples of these methods include getPitch, getForward, getFireButton, and so on. These methods are implemented in subclasses (eg., *DIS_Controller_Flybox*), which call methods of their corresponding *Control_Device* class to get actual input data.

Head's-Up Display Classes. Each *DIS_Controlled_Entity* also creates and maintains a pointer to a *DIS_HUD* object. *DIS_HUD* is designed to draw a head's-up display in front of the entity's viewpoint – the point to which the eye point location is transformed on each rendering cycle. (This is accomplished by *DIS_Controlled_Entity* by composing the entity's position matrix with a matrix specifying an offset for the view point in local coordinates.) *DIS_HUD* is subclassed into *DIS_HUDQ*, which also includes methods to process and display cueing PDUs. Versions of these methods appropriate to a specific entity type are then implemented in subclasses such as *DIS_HUDQ_DI*, which simulates the point of view of a dismounted infantryman with a helmet mounted display employing transparent lenses. The lenses, cueing indicators, and text are drawn using basic GL calls, and positioned by pushing and popping offsets from the view location directly onto and off of the OpenGL stack. The presence of the cueing HUD can be toggled on and off via a button on the control device.

Using this common class framework, new control devices, control device mappings, overwatch entities, HUDs, and overwatch entity behavior schemas can be quickly implemented without making significant code changes to the simulation.

2.3.4. Cue Processing

Cue processing for the HUD is performed as follows. For the DI hud, two small hollow cueing circles are displayed, one near the bottom of each lens to minimize interference with the operator's view of the virtual terrain. When a cueing PDU with lat/long coordinates is received from the DIS interface by a *DIS_ControlledQ* object, two vectors are created: one in the object's current viewing direction, and one from the eye point to the cue point. The angle between them is computed from their dot product using the relation

$$\theta = \cos^{-1}\left(\frac{\mathrm{v1} \bullet \mathrm{v2}}{\|\mathrm{v1}\|\|\mathrm{v2}\|}\right) \tag{5}$$

The relative direction (left or right) to the cue point from the HUD's point of view is computed by taking the cross product of these two vectors, and then taking the dot product of the result with a reference vector pointing upwards. The sign of the result gives the relative orientation to the cue point. The cueing circle on the corresponding lens is then flashed, signaling to the overwatch operator that he should turn in that direction; the range and angle to the cue point are displayed directly above the cueing circles. As the overwatch entity turns, the range and angle to the cue point are updated until he is facing in the proper direction. At that point the flashing stops and both cueing indicators are displayed as solid circles. The operator can then use a flybox switch to magnify the scene (simulating the use of binoculars), and/or the 10X power scope mounted on his virtual rifle to visually locate and assess the source of the MTI detection. This configuration is illustrated in figure 3-3.

2.3.5. M16 Rifle and DIS DI Classes

The visual simulation also allows the user to create a simulated M16 rifle with a 10 power scope, which can be controlled by the user through a position sensor mounted on a toy gun. A switch allows triggering a simulated CIDDS interrogation, which searches for CIDDS equipped entities in the direction the rifle is facing, within the appropriate range and beamwidth. A red light is displayed on the simulated rifle if a positive (friend) CIDDS response was received. A specialized subclass of DISEntity, with the capability to respond positively to a CIDDS query, is used to instantiate friendly DIS DIs.

3. SIMULATION AND EXPERIMENTATION

3.1. Scenario Illustration

In order to evaluate the effectiveness of the simulated GBMTI radar, we performed several ModSAF-based experiments using a mix of friendly and enemy dismounted infantry on the Hunter-Liggett terrain. The basic scenario placed a minefield blocking a road that runs along a valley between two ridges (see Figure 3-1a). The MTI radar was placed halfway up one of the ridges (see Figure 3-1b) overlooking the valley; this is illustrated in Figure 3-1b using a simulated high mobility multi-wheeled vehicle (HMMWV). The overwatch operator was placed on the opposite ridge. The ModSAF controlled dismounted infantry were placed at distances between one and about four kilometers from both the MTI radar location and from the overwatch operator. These units were given movement tasks to approach the minefield from several directions such that variations in terrain elevation and other features would at times obscure line of sight to them from both the radar and the overwatch operator.

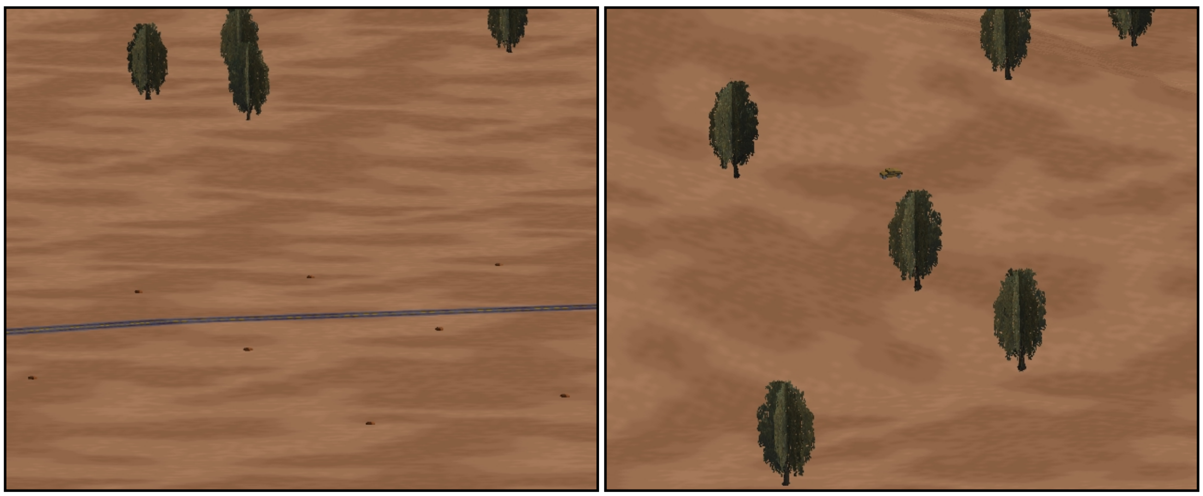

Figure 3-1 a: Simulated Minefield (Visible for Demo Purposes Only).
b: Simulated HMMWV with GBMTI Radar.

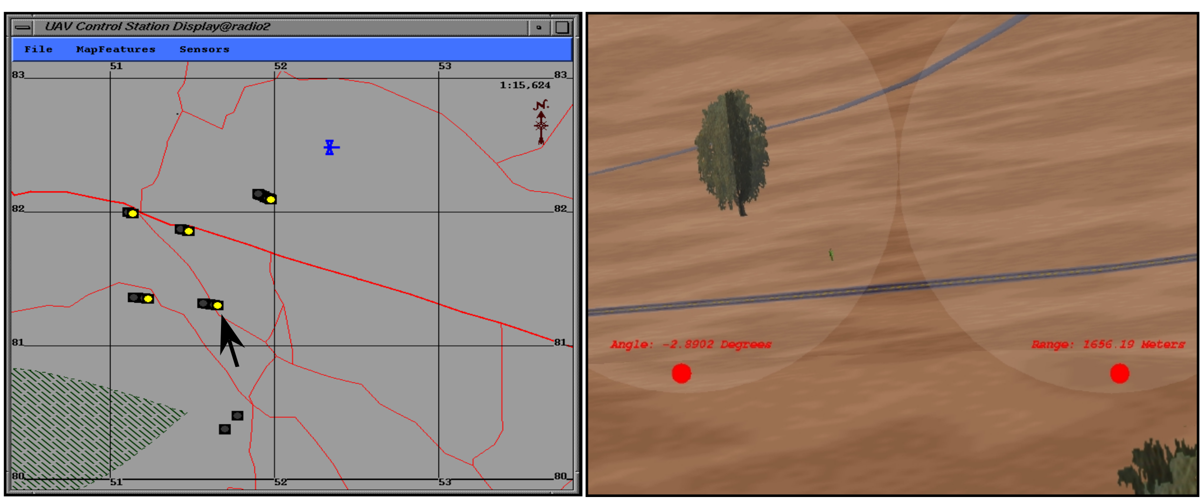

Figure 3-2a: M2MTI Active Display Showing a Target Designation.
b: DIs Detected by M2MTI and Visually Located by a Cue.

Despite the fact that the resulting MTI tracks were sometimes broken and irregular due to terrain obscuration, we found that the simulation operators were typically able to detect and make visual contact with the moving DIs quickly -- usually within several minutes of their coming into unobstructed sight of the radar, and at up to about four kilometers distance. Figure 3-2 illustrates a typical scenario. Figure 3.2a shows the active M2MTI display and the point and click target designation made by the operator, while figure 3-2 b shows the resulting target cue and tracking by the observer. Figure 3-3 shows the overwatch operator tracking two DIs approaching on a road with the rifle scope at about 3.6 km distance. Figure 3.3 b shows a DI being interrogated using the rifle's simulated CIDDS system. Using CIDDS, the observer can determine whether a DI who may be too far away for visual identification is a ***friendly*** (is equipped with a responding CIDDS transponder) or an ***unknown***. This knowledge can be valuable in helping to prevent a potential fratricide incident. In figure 3.4, the red indicator (the upper of the two on the left side of the gun) is activated, indicating a "friendly" CIDDS response.

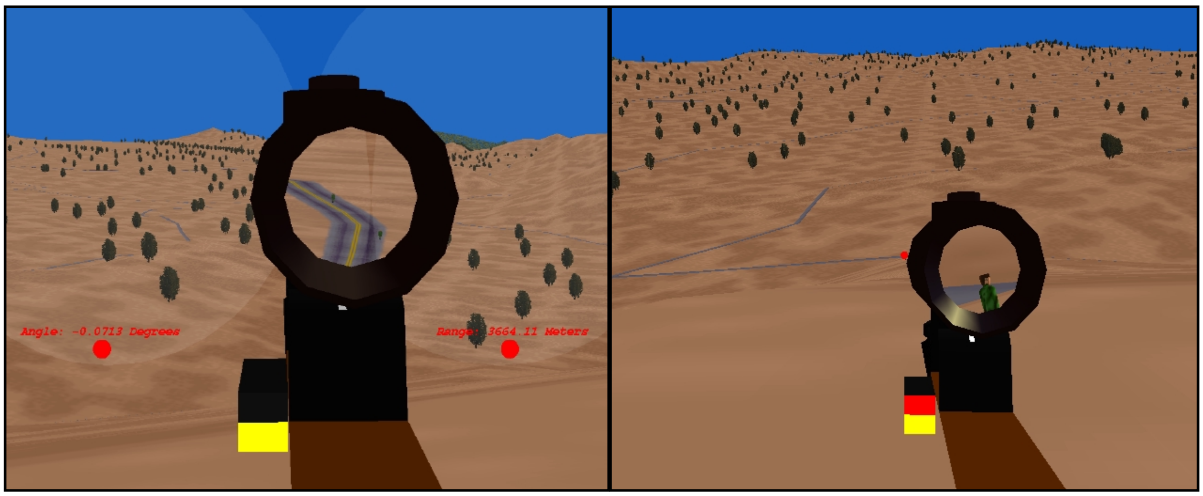

Figure 3-3 a: Approaching DIs Tracked by Observer Using 10X Rifle Scope.
b: Red CIDDS Indicator (Top, Left side of gun) Identifies MTI-Detected DI as Friendly.

4. SUMMARY AND CONCLUSIONS

The APLA overwatch mission encompasses several tactical challenges. These include the need to detect man sized targets at night and in most operational weather conditions, the need to communicate in ground proximity conditions, and to operate during periods of "interference", and the requirement to be able to identify friendly forces. The use of ground-based MTI radar as an external overwatch sensor is a new concept which has many potential benefits in mine warfare. The interactive simulation and analysis described in this paper provide a "proof-of-concept" demonstration and a planning tool for future field experiments which will demonstrate the use of GBMTI radar in the minefield overwatch capacity.

5. ACKONWLEDGEMENTS

The authors wish to acknowledge support from the US Army PM Mines, Countermines, and Demolition (MCD) under contract DAAB07-99-C-C201, and US Army PM Night Vision/Reconaissance Surveillance Target Acquisition (NV/RSTA).

REFERENCES

1. Bracht, R.; Miller, E.K., Kuckerz, T.(1997) "Using an Impedance-Modulated Reflector for Passive Communications" IEEE AP-S International Symposium and URSI North American Radio Science Meeting.

2. J. SantaPietro, "A DIS Model for a Multisensor Airborne Platform", *Proc. of 6th Annual Ground Target Modeling & Validation Conference*, Vol. 1, pp. 1-9, Houghton MI, August 1995

3. D.C. Schlerer, MTI and Pulsed Doppler Radar, Artech House, 1991, Norwood, MA

4. F.T. Ulaby, Handbook of Radar Scaterring Statistics for Terrain, Artech House, 1989, Norwood, MA

5. Foley, Van Dam, Feiner and Hughes, *Computer Graphics: Principles and Practice,* pp. 204-217, Addison-Wesley, 1990, Reading, MA.