SIMULATION SOFTWARE FOR NAVAL SURFACE WARFARE TRAINING

Sergiu Dascalu, University of Nevada, Reno, USA, dascalus@cse.unr.edu
Sermsak Buntha, University of Nevada, Reno, USA, buntha@cse.unr.edu

ABSTRACT

This paper presents a prototype simulation software system that provides support for centralized command and control in naval surface warfare. To build this prototype, we have designed the modes of operation for the system and developed algorithms for its execution. These algorithms include intercepting an enemy target platform and planning an escape route. A graphical user interface with easy-to-use options facilitates the interaction with the computer by navy commanders and tactical action officers. The paper provides background information on naval surface warfare, describes the main elements of the prototype's operational capabilities, and details the system's interface and modes of operation.

KEYWORDS: Simulation, Navy, Combat, Command and Control, Interception, Escape Route.

1. INTRODUCTION

Commanding a naval task force is an individual skill that needs to be developed over time. The more experience commanders gain, the higher is the quality of their commands. Obviously, sea training is both highly necessary and costly. Several organizations have constructed various simulations to address this need [2][13]. For example, the Joint Warfighting Center has designed JTLS-JCATS to train combatant commanders for emergency operations [4]. Several other researchers and practitioners have created battlefield displays and combat information systems to provide better combat situation understanding and decision-making support [5][10][7][11][14]. Furthermore, a number of investigators have developed intelligence applications to facilitate commanders' tasks [5][6][12].

Our proposed simulation software is designed to help commanding officers and tactical action officers learn how to take action in combat situations. The software tool described in this paper allows instructors to simulate tactical situations such as escorting high value units or composing a surface action group (SAG) to attack multiple hostile targets.

The paper is organized as follows: Section 2 provides a brief overview of our simulation software system, Section 3 describes the main surface warfare operations supported by the software, Section 4 presents the current status of our work and directions for future development, and Section 5 concludes the paper.

2. GENERAL DESCRIPTION

The proposed simulation software environment is intended to increase the surface warfare experience of action commanders while reducing the need of costly real operations in the ocean. The software we developed allows instructors to create battlefield situations and then save them as text files for later use. Therefore, trainees are able to improve their strategies on later iterations of playing the same situations and facing the same problems. In addition, we have integrated intelligence modules for handling tasks that require human effort, such as finding the interception course or calculating the best escape route. For the latter we have designed a new algorithm whose purpose is to ensure that the ship will arrive at the given destination without entering the enemy hazard areas. Furthermore, the escape route generated by our software system is the shortest possible route under any given set of specified conditions. The software system developed is also designed for cost efficiency. Currently, it runs on regular personal computers powered by the UNIX operating system. We adopted OpenGL [1] and GTK+ [9] to be the software graphical libraries of our software. With the current computational power of common personal computers, our system is able to graphically display and update the battlefield situation in real-time. For training purposes, the users have the option to increase the time speed factor up to 256 times the real time. This allows various missions to be completed in simulation sessions much faster than during actual sea training.

3. OPERATIONAL CAPABILITIES

The operational capabilities of our simulation software can be categorized into three main functionalities: combat information display, command and control, and intelligence operations.

3.1 Combat information display

The Combat information center (CIC) is where gathering, processing, displaying, evaluating, and decision making based on an incoming stream of information takes place. CIC plays a key role for command and control tasks. It is of paramount importance that the in-charge commander clearly comprehends battlefield circumstances in real-time so that he or she can make the best possible decisions. In order to provide an accurate picture of the combat situation, our software system incorporates a combat information display module based on NTDS symbols [3][8]. Figure 1 shows two window panels depicting the battlefield situation (the operation map and the detailed information on own and target ships). It also presents the system's main menu.

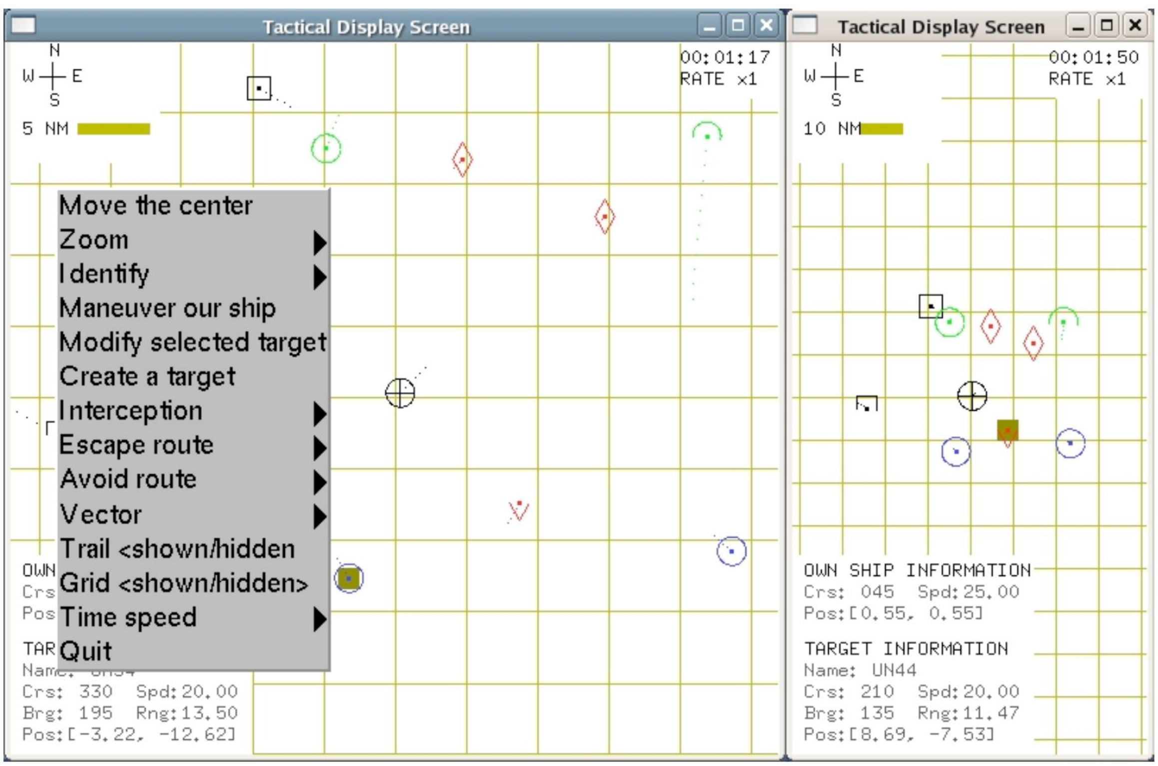


Figure 1. Main interface of the simulation software, including the activated main menu, an overview operation map, and a detailed information panel

The software system allows the users to create multiple windows for monitoring different areas of a battlefield or different display scales of the same area. Figure 1 presents two window panels displaying the same battlefield area with different scales. The commander can use the left panel for monitoring the entire battlefield situation and the right panel to focus on a specific area of the battlefield. On the upper right-hand side of each window panel the time and the time speed factor are displayed. In real-time execution, the time speed factor is 1. Higher time speed factors are for running faster simulations. In the upper left-hand side corner of the screen the cardinal directions (North, East, South, and West) and the display scale are shown. For scale indication, the number under the cardinal directions represents the distance in nautical miles associated with the solid bar on its right. The lower left-hand side corner of the screen provides details on the current navigation data of our own ship and the tactical information of a selected target. The user is able to select a target by simply clicking on the target symbol, and then the clicked target will be indicated (highlighted) on screen by a yellow square. The target's information consists of its name, course, speed, bearing and range from our own ship. The target's coordinates are also displayed in this corner.

Each window has independently displaying properties, such as the coordinates of the window's center, window size, display scale, presentation of targets' moving vector and trail, and so forth. Each window can be moved and resized and can overlap other windows. For current position and other information of targets, every displaying window acquires data from the system's centralized memory. Thus, when information on targets is updated, this will be reflected in all displaying windows. This update is triggered periodically by our system's clock and occasionally by user actions. The time speed factor is also kept in the system's memory.

3.2 Command and control

The main goal of this simulation software is to allow a trainee to issue commands and to control the task force. Identifying unknown targets and maneuvering the own ship are the major tasks of a tactical action officer. First, the in-charge officer must identify hostile targets from the provided data, such as targets' movement, radar emission pattern, and so forth. Then, the officer has to perform actions to counteract existing threats, primarily by making decisions regarding the movement of his or her own task force. The simulation system's embedded intelligence modules, such as interception, can facilitate these types of action.

Furthermore, the software system supports changes in target information while the program is running. This allows the user to add or remove targets as well as to change any target's name, type, course, speed, or position (the latter in the form of bearing and range). Figure 2 shows the dialog box that allows modification of an existing target's information.

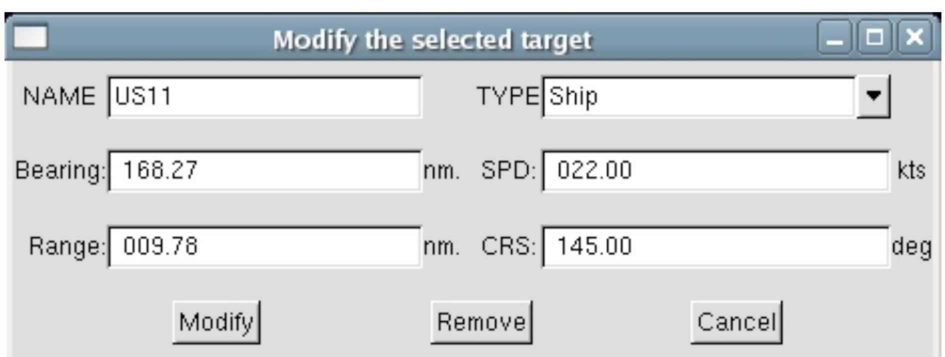


Figure 2. Dialog box for modifying a target's information

3.3 Intelligence operations

Currently, the two main intelligence operations implemented in our system are calculating the interception course and planning the escape route. Determining the interception course with a primary operational speed is a common task for a tactical action officer maneuvering the ship towards a single target. Most navy officers learn to achieve this task by using a grease pencil and a radar-like tactical board to compose relative and true motion vectors of a target and our own ship. Figure 3 shows the composition of the vectors used in this procedure. Done manually, this technique can delay the operational action. However, this delay is avoided via computer calculations, as our software system calculates the interception course instantly. Given the vectors shown in Figure 3, the following formula is used:

$$O = T + R \qquad (1)$$

This yields the following scalar equation, where α and β are angles of T and R, respectively:

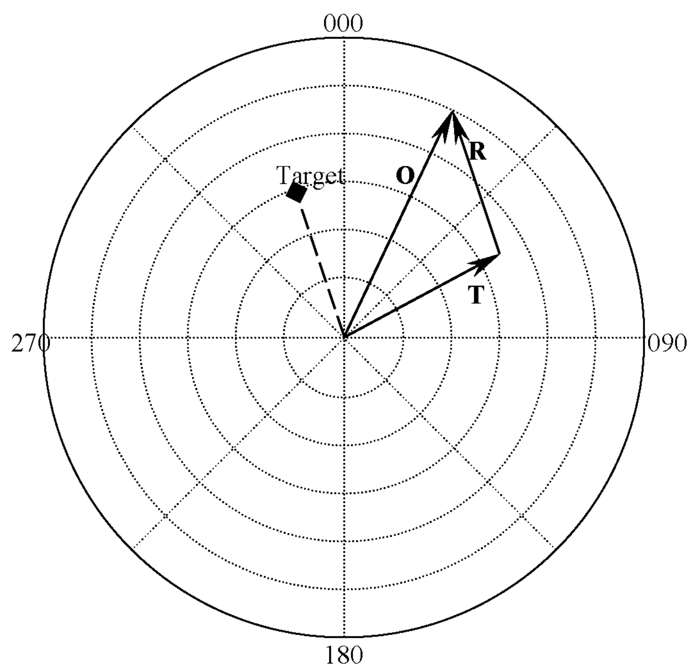
$$|O|^2 = |R|^2 + 2|R||T|\cos(\alpha - \beta) + |T|^2 \qquad (2)$$

To determine |R|, the equation (2) is rewritten as:

$$|R|^2 + \{2|T|\cos(\alpha - \beta)\}\ |R| + \{|T|^2 - |O|^2\} = 0 \qquad (3)$$

Next, the larger positive root is determined. If the root values are negative or imaginary numbers, then that interception is impossible. From (3), the angle of interception θ is calculated as:

$$\theta = \tan^{-1}\left(\frac{|\mathrm{R}|\sin\alpha + |\mathrm{T}|\sin\beta}{|\mathrm{R}|\cos\alpha + |\mathrm{T}|\cos\beta}\right) \qquad (4)$$



Vectors showing course and speed:
O: True vector of interception
T: True vector of target
R: Relative vector of interception

Composition of vectors:
O = T + R ; where
- Speed of **O** is defined by commander
- Speed and course of **T** are obtained from the radar
- Course of **R** is the line from the radar center to the target
- Course of **O** is the interception course, which needs to be found

Result:
When our own ship has speed and calculated course **O**, we will see it on screen moving towards the target

Figure 3. Composition of vectors used to determine the interception course

One of a tactical officer's most difficult tasks is to design the route for navigating through an enemy task force by keeping a particular distance from the enemy's platforms, such as enemy's radar range, or missile engagement zone. Obviously, in naval operations the enemy's platforms are moving. We have developed an algorithm of generating the best escape route through multiple moving platforms without entering the hazard areas of enemy platforms. This algorithm has been implemented into the simulation software system. The escape route is generated under the knowledge of concurrent hostile vessels' positions, moving directions, and speeds. Building on the foundation of interception calculation, we enhance the method with A* search [15] to determine the best possible escape route. Instead of finding the interception course directly to the target, we find all possible interception paths to the end points of created octagons over circular hazard areas of all hostile targets. Then, we update the hostile targets' positions from their courses and speeds and create subsequent paths. The paths are extended until the route reaches the intended destination. Any path entering the hazard area of one or more hostile targets is eliminated and not searched further. Because the algorithm is rather complex, its details are not included in this paper (they will be the subject of one of our future papers).

Figure 4 shows an escape route generated by our software system. The route information is indicated on the left-hand side of the window. This information includes the positions of the turning points, the new course after turning, and the length of each route segment.

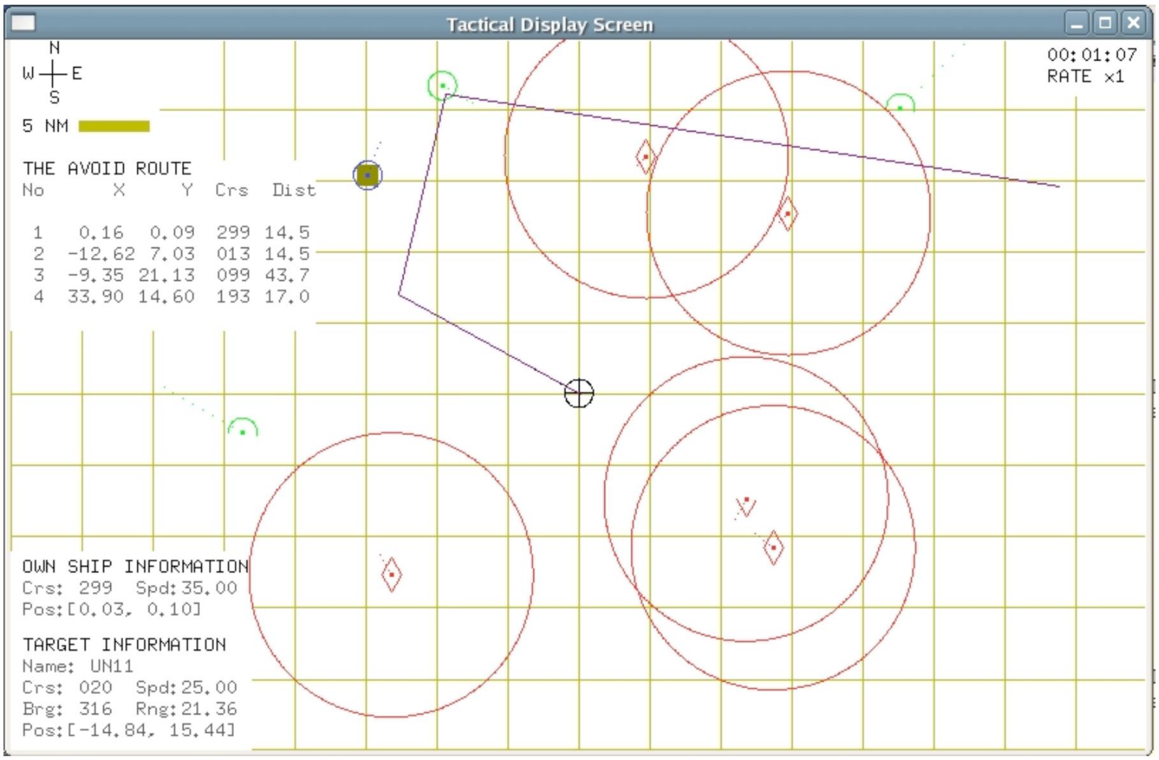


Figure 4. Example of escape route generated by the software system

4. CURRENT STATUS AND FUTURE WORK

At this point in time the proposed simulation software is running on a standalone computer, thus allowing only a single person to be trained at a time. As is the nature of the Navy mission, “no man stands alone”. To complete a mission, the collaborative work of a team is highly important. Our future implementation will support multiple machines in a networked solution that will allow trainees to collaboratively play the simulation against opponent groups. This will also further motivate trainees to discover new operational tactics to compete with their opponents and improve their skills.

Currently, our software system assumes that all incoming pieces of battlefield information have been gathered, processed, stored in memory, and displayed on the screen. A trainee practices for evaluating and making decisions based on the information available in a given situation. In the future, shipboard sensors such as radar and sonar will be embedded in the system to allow trainees to also perform tasks related to gathering and processing other battlefield-related data.

Because in the present era of naval combat electronics warfare is the primary key for victory in many battlefield situations our future system will have increased functionality in this area as well.

5. CONCLUSIONS

This paper has presented the prototype of a simulation software system for surface warfare training. The software system displays top and detailed views of a naval surface battlefield to allow commanders to monitor and comprehend combat situations in real-time. The simulation software allows commanders to command and control their task force in various combat situations such that they are able to gain combat experience without expensive actual sea training. Furthermore, the software system has embedded intelligence modules that significantly reduce computational effort by humans. The intelligence operations described in this paper are for target interception and escape route planning. Currently, the software system is working on a stand-alone computer. Our next work will focus on a networked solution of the system to allow collaborative training and increased simulation capabilities.

6. REFERENCES

[1] E.S. Angle, Interactive Computer Graphics: A Top-Down Approach with OpenGL, Addison-Wesley, 2000.

[2] C.L. Blais, "Prototyping Advanced Warfare Gaming Capabilities," Proc. of the Winter Simulation Conference, vol. 1, 1998, pp. 841-848.

[3] D. L. Boslaugh, When Computers Went to Sea: The Digitization of the United States Navy, IEEE Computer Society Press, Los Alamitos, CA, 2003.

[4] F. A. Bowers and D. L. Prochnow, "JTLS-JCATS Federation Support of Emergency Response Training," Proc. of the 2003 Winter Simulation Conference, vol. 1, 2003, pp. 1052-1060.

[5] S. Buntha, "Command Control Communications Computers and Intelligence (C^4I) Software for Naval Surface Warfare," Master thesis, Department of Computer Science and Engineering, University of Nevada, Reno, NV, USA, 2004.

[6] C. Cox and D. Fu, "AI for Automated Combatants in a Training Application," ACM International Conference Proceeding Series, vol. 123, 2005, pp. 57-64.

[7] D. Gillard and P. K. Heim, "U.S. Navy Progress in Electronic Navigation," Oceans' 02 MTS/IEEE, vol. 2, 2002, pp. 1038-1041.

[8] GlobalSecurity.Org, "Reliable Security Information," accessed February 2006 at http://www.globalsecurity.org/index.html.

[9] GTK+ Team, GTK+: the GIMP Toolkit, accessed January 2006 at http://www.gtk.org.

[10] J. M. Gushing, S. Allen, and S. Guerlain, "Measuring the Effectiveness of an Overview Display for the Mission to Launch Platform Assignment Process Essential to the Tomahawk Strike Coordinator," Proc. of the 2004 IEEE International Conference on Systems, Man, and Cybernetics, vol. 2, 2004, pp. 1533-1537.

[11] S. P. Harrison and M. F. van Norden, "Electronic Chart Display Overlay Development at the Naval Oceanographic Office for Use in Tactical Applications," Oceans' 02 MTS/IEEE, vol. 2, 2002, pp. 1033-1037.

[12] S. J. Louis, J. McDonnell, and N. Gizzi, "Dynamic Strike Force Asset Allocation Using Genetic Search and Case-based Reasoning," Proc. of the Sixth Conference on Systemics, Cybernetics, and Informatics, 2002, pp. 855-861.

[13] G. F. Stone, and G. A. McIntyre, "The Joint Warfare System (JWARS): A Modeling and Analysis Tool for the Defense Department," Proc. Winter Simulation Conference, Vol. 1, 2001, pp. 691-696.

[14] J. H. van Delft and J. M. Schraagen, "Decision Support Interfaces," Proc. 2004 IEEE International Conference on Systems, Man, and Cybernetics, Proc. 2004 IEEE International Conference on Systems, Man, and Cybernetics, vol. 1, 2004, pp. 827-832.

[15] P.H Winston, Artificial Intelligence, Addison-Wesley, 1992.