The Sensor Driven Airborne Replanner: The System Flying a Mission Under Simulation

Richard C. Wagner
Aerospace Engineer
Advanced Processors Branch, Code 5051
Naval Air Warfare Center, Aircraft Division
Warminster, PA 18974-5000

Introduction

This paper illustrates the U.S. Navy's Sensor Driven Airborne Replanner (SDAR--pronounced, "ess'dar") flying a mission under simulation. SDAR[1] is an autonomous control system for Unmanned Aerial Vehicles (UAVs). The SDAR system employs artificial intelligence techniques to impress the interests and intentions of a human mission planner onto the real-time behavior of a fully autonomous surveillance UAV. The system is developed and tested using a PC-based, high-fidelity environment simulation[2]. The simulation allows SDAR to control a simulated UAV and camera, and models the entire data environment in which the system is immersed while in flight. The heart of the simulation is a six degree-of-freedom dynamic model of an aircraft. This is the same kind of model that is used to drive manned flight simulators. The SDAR simulation's model has been validated against a number of aircraft, and provides high correlation and fidelity. Over the flight dynamic model, there are realistic models of the aircraft's sensors and effectors. On the sensor side, there are models of the air-data sensors, an Inertial Measurement Unit, a GPS receiver, a camera and its gimbal, a vision processing system, and a COM/ESM receiver. The effectors include models of the aircraft's control surface and throttle servos, the gimbal drive motors and the camera lens' zoom drive motor. Together, these models form what SDAR believes to be a real UAV. The simulation also contains models of ships on the ocean surface, giving SDAR targets to prosecute, and wind, to make the environment complete. During development and testing, the SDAR system, running on one of its RTX2000-based airborne units, is connected to the simulation PC through a serial link. SDAR controls the aircraft, camera and gimbal through the link and receives data from each of the sensors over the same link. The images presented in this paper are generated by the simulation, and show the mission as it progresses.

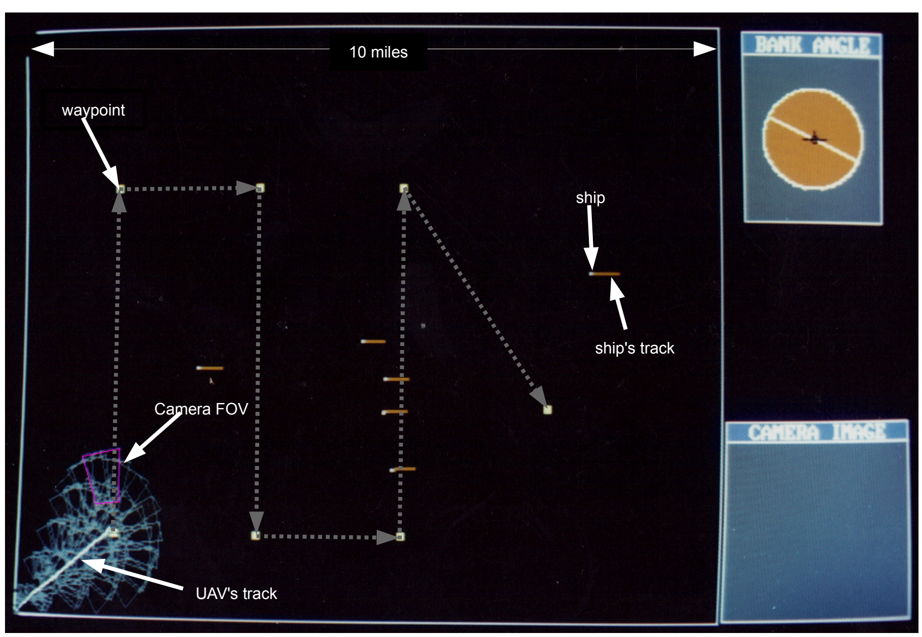

Figure 1: The simulation display

Figure 1 is a view of the simulation display. You, as a viewer, are oriented above the ocean surface looking straight down. The area enclosed by the white frame is about 10 x 10 statute miles. The large white squares represent waypoints, assigned latitude, longitude and altitude coordinates that the mission planner has programmed into the SDAR system as a provisional search path. The waypoints for this mission form a "ladder search", and the search path is indicated by the gray lines connecting them. The white track at the lower left-hand corner is that of the UAV. The UAV itself is at the very head of this track. The magenta trapezoid is the projection of the camera's field of view (FOV) down onto the ocean's surface. The camera sees whatever is within the FOV, and the simulated camera image is displayed in the window on the lower-right of the display. SDAR moves the camera in increments as it scans the ocean surface, and after each movement, processes the image looking for ships. This is done once per second. On the simulation display, old FOV footprints are colored gray so that the developers can observe how thoroughly the search area was viewed by the system. The bank angle display at the upper-right-hand corner shows the attitude of the UAV. Last, the brown lines are the tracks of ships moving on the ocean surface. The white dot at the head of each track is the ship itself. Each ship has its own course and speed.

<u>The Mission</u>

In Figure 1, the aircraft has recently been launched and is climbing on course. Note how the size of the camera FOV footprints is increasing as the UAV gains altitude. The UAV has just reached its first waypoint and is currently in a left bank, turning for the second waypoint. The mission planner has set the altitude of each waypoint at 5,000 feet MSL, and the UAV is nearly at that altitude. The mission planner has told the SDAR system that he is interested in any ships over 200 feet long, going any speed, on any course. SDAR is currently in its "search" state, flying the provisional search path while scanning the surface for ships and listening for transmissions.

In Figure 2 SDAR has encountered its first ship. SDAR always attempts to make an aft approach to any ship, but in this encounter it spotted the ship when it was directly off of the bow. Immediately following the initial sighting, SDAR entered its "*verify*" state, aiming the camera at the ship for two seconds and zooming in the camera in order to get accurate size, position, course and speed data. Those data were then inserted into SDAR's ship database. SDAR compared the database entries to the mission planner's interests and found this ship to be interesting. The system then transitioned into its "*pursue*" state, flying a course directly at the ship while scanning the camera. When it reached the "avoid radius", the radius from the ship that it must never enter unless in pursuit, SDAR transitioned into its "*ingress*" state, flying the UAV on a course taking it to a tangent with an even closer "orbit radius". Upon reaching the orbit radius, SDAR transitioned into "*orbit*" state, aiming the camera at the ship and flying around the orbit to the ship's beam--where we see it here. SDAR is now in "*photo*" state, indicated by the white

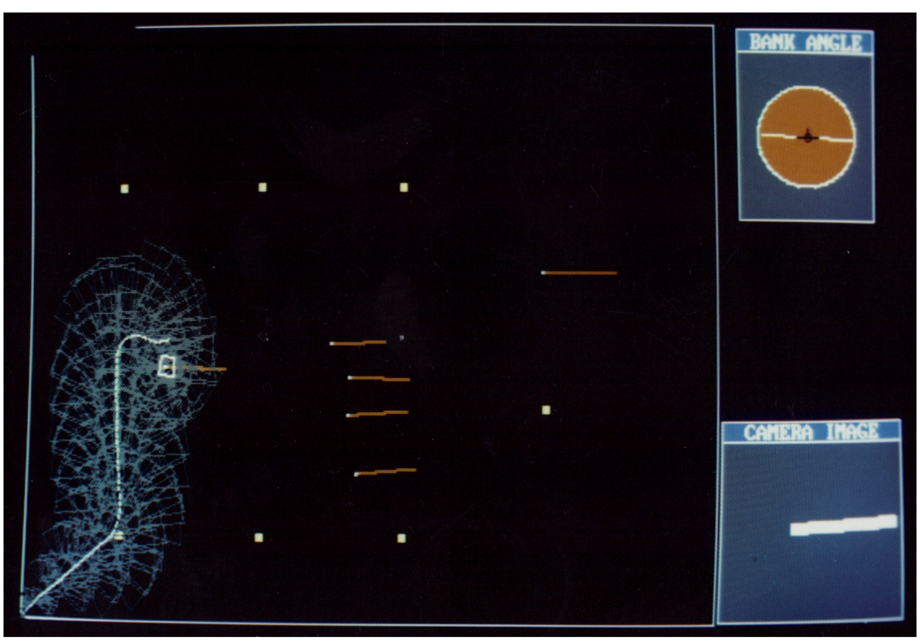

Figure 2: The first encounter

camera FOV. It has zoomed the camera in to get a good picture of the ship and is now recording images of the ship. The ship is visible in the simulated camera image. With each image recorded, the ship's interest level is reduced. And so after a few shots, the ship becomes uninteresting. At that point, SDAR will transition into its "*egress*" state, flying directly away from the ship to the avoid radius. There, it will transition back into the "*search*" state, returning to the waypoint search. This ship's entry in SDAR's ship database will be maintained for some time, with its position constantly updated based on its course and speed. Should SDAR encounter this ship again, it's actual position will likely correlate with its predicted position, indicating that it already resides in the SDAR database. SDAR will know that the ship is uninteresting and ignore it, or avoid it, if it is in the way.

In Figure 3, by observing the track, you can see SDAR's egress away from the first ship and its return to the waypoint search. It has made the second and third waypoints and is now headed for waypoint 4. Note the double UAV track exiting the third waypoint. This is a GPS dropout. The GPS signal has failed and SDAR has reverted to its Inertial Navigation System (INS). The INS system isn't perfectly accurate; its position estimate contains an ever-increasing drift. The white track is the actual position of the UAV. The cyan track is where it thinks it is. The GPS signal is reestablished where the cyan track ends. Flying from waypoint 3 to waypoint 4 in its "*search*" state, SDAR has encountered a group of four close ships. After sighting the first (top) ship, SDAR verified it, aiming and zooming-in the camera. In the process, the second ship fell into the camera FOV. SDAR then verified that ship, sighting the third and then the fourth. It now has accurate data on each ship and has inserted them into its database. The first ship is interesting, and so SDAR must prosecute it. However, the first and second ship's avoid radii overlap. There isn't enough space between the ships for SDAR to do its job. So SDAR has transitioned into its "*escape*" state. In this state, SDAR will fly the UAV directly away from the two ships while taking images of each of them. (The second ship is also interesting.) These images will be sub-optimal, but SDAR is making the best of a bad situation. It will avoid overflying either ship while recording images for the mission planner, and that is successful, in this case. In this image, SDAR has just finished recording the shots. These two ships are now uninteresting. The third ship is interesting, and so SDAR must still prosecute it.

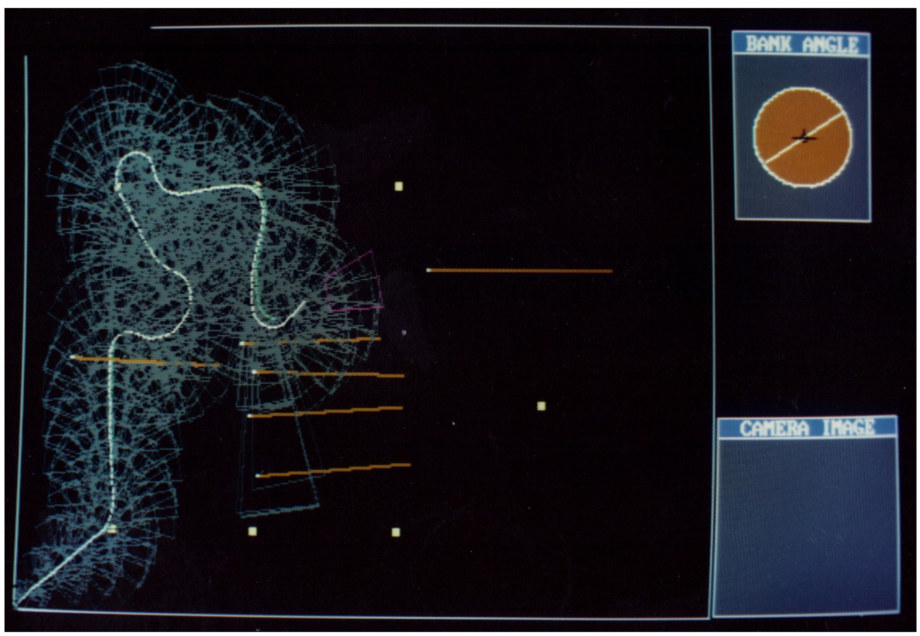

Figure 3: A multi-ship encounter

In Figure 4, after escaping from the first two ships, SDAR entered "*pursue*" state, and turned the UAV around to fly directly toward the third ship. But its avoid radius overlapped with those from both the second and fourth ships. SDAR entered the "*escape*" state again, flying away from the third and fourth ships, but only taking pictures of the third, as the fourth ship is uninteresting (less than 200 feet long). SDAR then made the fourth waypoint, flew to the fifth, and is now flying to the sixth. The

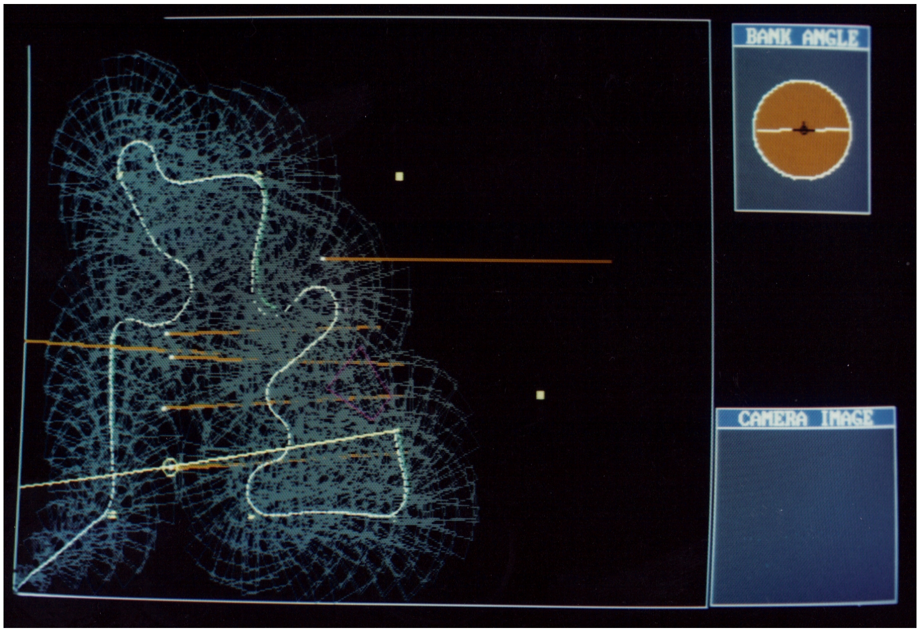

Figure 4: An uninteresting COM/ESM encounter

fourth ship of the previous encounter has just made a communications frequency transmission and SDAR's COM/ESM receiver has picked it up. The yellow line indicates the bearing from the UAV to the transmitter. The COM/ESM receiver can identify different transmitter types, and this type is uninteresting to the mission planner. SDAR will ignore it.

In Figure 5 SDAR is continuing its progress toward the sixth waypoint. The first ship from the previous encounter has just transmitted. The bearing from the UAV to the transmitter is indicated by the yellow line. Comparing this transmitter type to the transmitter interest matrix, SDAR has found this transmitter to be interesting to the mission planner. It has entered its "*vertical pan*" state, continuing to fly the search path, turning the camera to the transmitter bearing, and then panning the camera vertically, to the horizon. In this image, it has just sighted the first two ships from the previous encounter. The ships are in the upper-left of the camera FOV, and are visible in the camera image. SDAR just calculated position, course and speed data for these ships and has compared their data to the entries in the ship database. These ships correlate with

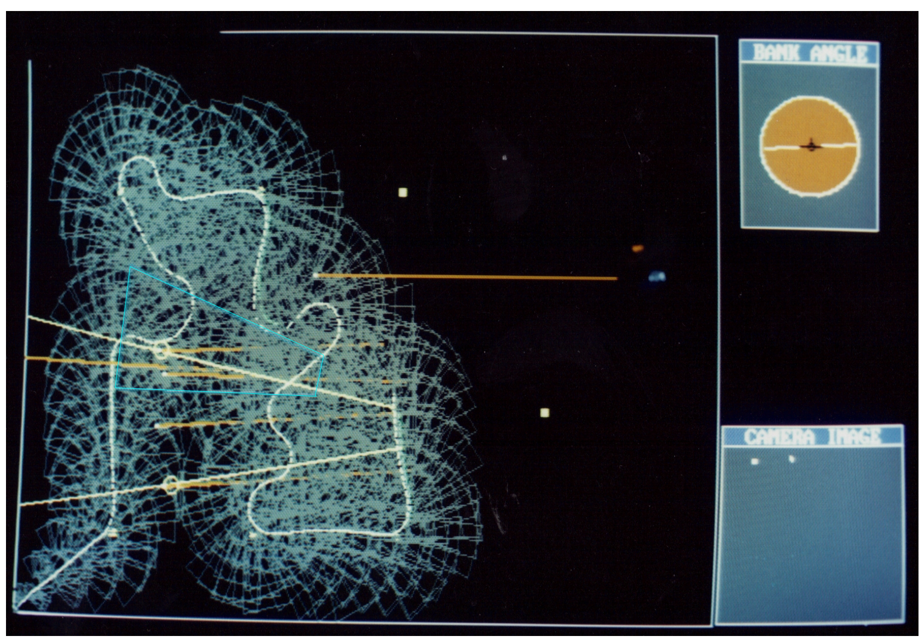

Figure 5: An interesting COM/ESM encounter

existing entries, and so SDAR knows it has seen these ships before. They are both uninteresting because they have already been prosecuted. SDAR will ignore them.

In Figure 6 another ship has just transmitted. This transmitter is interesting and so SDAR will enter its "*vertical pan*" state to look for the ship. In this image, the camera has just been turned to the bearing of the transmitter.

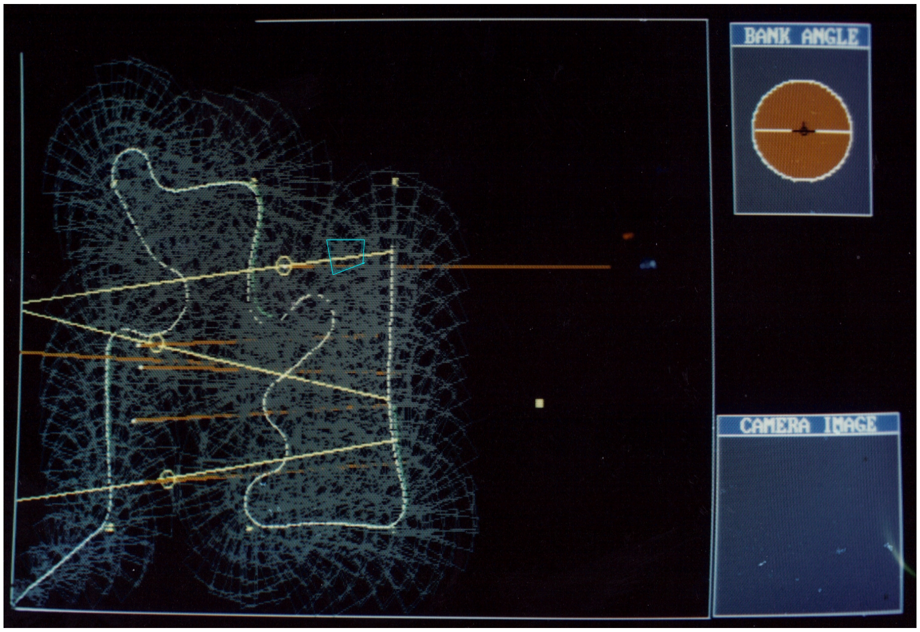

Figure 6: Vertical pan maneuver beginning

In Figure 7, SDAR has vertically panned the camera and the ship is now within the FOV. The ship, as SDAR sees it, is visible in the camera image window. SDAR has entered its "*verify*" state and is now getting accurate data on the ship. This ship is interesting to the mission planner and hasn't been prosecuted yet.

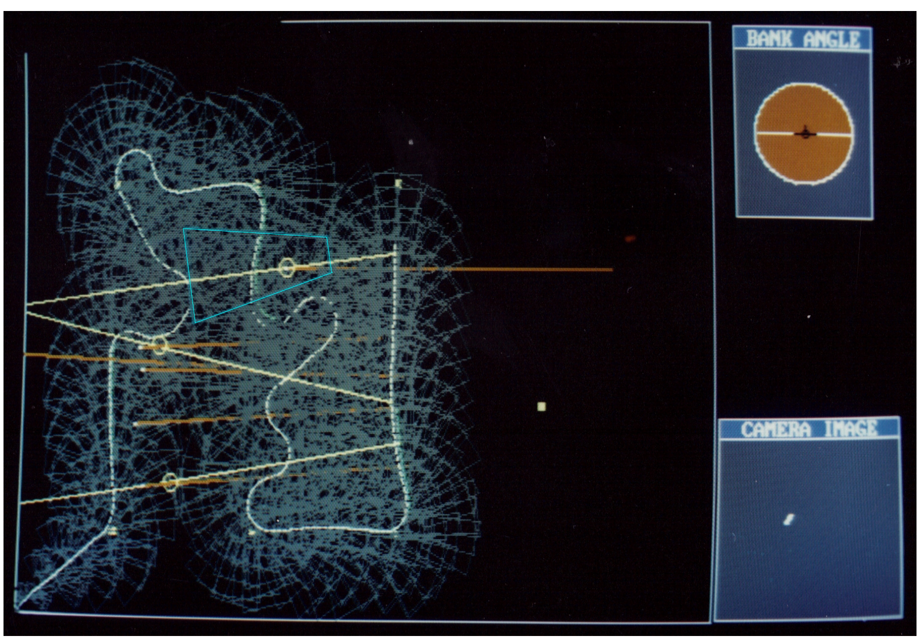

Figure 7: Tallyho

Figure 8 shows a classic encounter. Through sensor cross-cueing, the SDAR system has spotted this interesting ship and entered the ship's data into its ship database. It then went through the usual progression of states: "*pursue*", flying directly at the ship using a stern approach; "*ingress*", taking it from the avoid radius to the orbit radius; and then into the "*orbit*" state. It's now flying at the orbit radius at the beam of the ship and has transitioned to its "*photo*" state. It's aimed the camera at the ship, zooomed in, and is now recording images of the ship. The current image is visible in the camera image window. This is the way single-ship encounters usually progress.

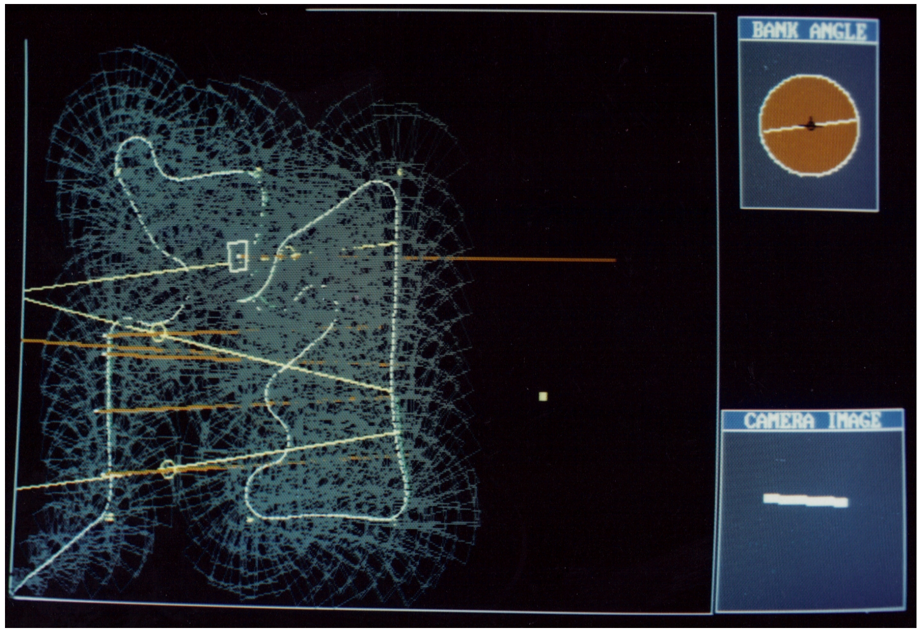

Figure 8: A classic encounter

In Figure 9 SDAR has taken enough shots of the ship to make the it uninteresting. It has egressed from the ship and is now back in "*search*" state. During the encounter with the ship, it came close enough to the sixth waypoint to consider that waypoint made. So it is now continuing on to the seventh waypoint, the last waypoint listed by the mission planner. When the UAV reaches that location, SDAR will orbit that point, awaiting rendezvous with its host ship. Then, a ship-based, human pilot will take control of the system remotely and land the UAV into a net--the end of the mission.

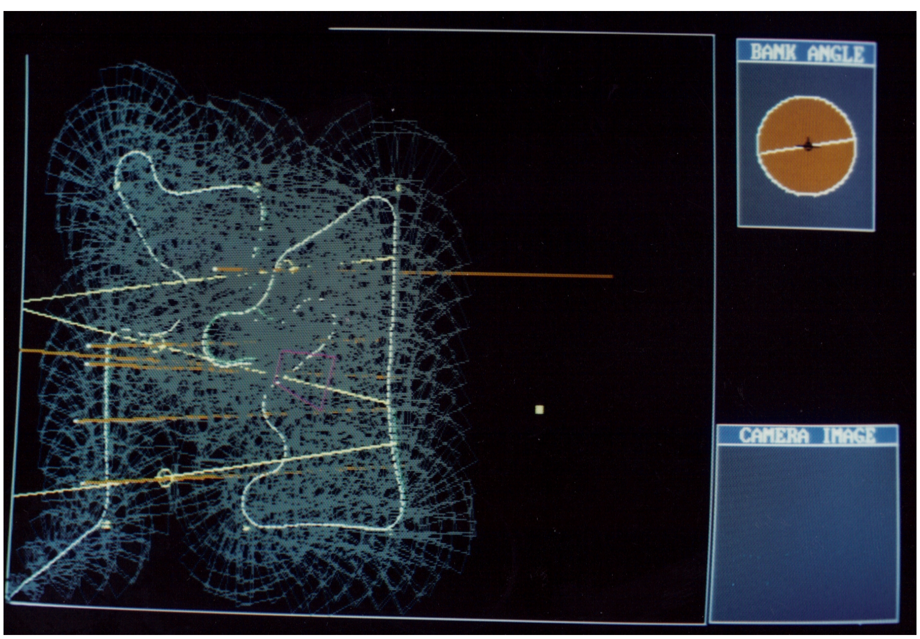

Figure 9: Exiting the encounter

Last, Figure 10 shows one of the SDAR system's RTX2000-based airborne computers. These powerful machines operate at a clock rate of 10 MHz, but can process more than one high-level *software* (not hardware) instruction per clock cycle. They provide bursts of up to 40 MIPS and operate at a measured average speed, processing the SDAR software, of about 15 MIPS. The SDAR artificial intelligence and control algorithms are required to operate at one processing cycle per second to control a UAV. But on these machines, they are capable of operating at 650 processing cycles per second, easily fast enough to control a missile.

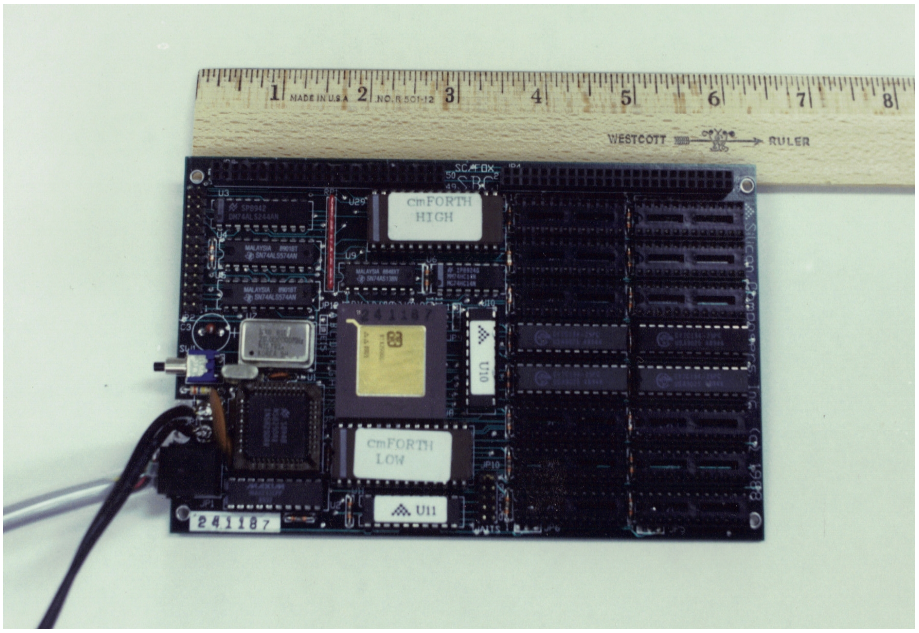

Figure 10: The SDAR flight hardware

References

1. Wagner, Richard C.: *The Sensor Driven Airborne Replanner: Artificial Intelligence for Unmanned Air Vehicles*, Proceedings of the 1992 Rochester Forth Conference, June 17-21, 1992, University of Rochester, Rochester, NY, ISBN 0-914593-12-9

2. Wagner, Richard C.: *The Sensor Driven Airborne Replanner Simulation: Building the World in a Box*, Proceedings of the 1996 Rochester Forth Conference, June 19-22, 1996, Ryerson Polytechnic University, Toronto, Ontario, Canada, ISBN 0-914593-16-1