

## ATC TOWER SIMULATOR: TACUND

Henry Hexmoor and Tim Heng  
University of North Dakota  
Grand Forks, North Dakota, USA  
{hexmoor,heng}@cs.und.edu

### Summary

We have implemented an air traffic control tower simulation program. This program allows an Air Traffic Controller (ATC) to use our program to control simulated flights in the vicinity of their tower post. Our program is designed as a training tool for students of air traffic control. We call our program Tower Air traffic Control at UND (TACUND). TACUND simulates flight of up to six planes simultaneously. These planes are different models of Piper (Cadet, Seminole, and Aero) and Boeing (737, 747). Planes fly along simulated paths. After receiving clearing for landing, planes enter a pattern of descent and the ATC can see the landing plane in its view over the background of our airport runways. Our simulator shows pictures of Grand Forks airport taken from the tower of this airport. The simulator is implemented in Java 1.1 and uses a viewing screen of size 800 by 600 or larger.

### Background and Related work

Many air traffic simulators exist. However, they either tend to be of high fidelity and very expensive software or of training variety [National Research Council, 98]. We wanted to build a simple system that can be used for training and research in fundamental concepts of intelligent agency. For instance, CTAS [Erzberger, 92] is a physical building in Ames the size of a real tower fully equipped with the latest graphics simulators. They can model all airports including weather conditions. ATC and pilots sit in their positions as they would in the real world and run through simulated scenarios. On the other hand, you can buy a training CD [BAO, 1992] that simulates simple commands like our own simulator. However, this system is a commercial product and we cannot experiment with it or extend it.

### Modeling

**Planes.** The space is modeled by a Cartesian and a polar coordinate system centered at the tower. With the Cartesian coordinate system, the center of the world (0,0) is the tower. In the polar system, North is zero and the angle formed between line connecting the plane position to tower and North is measured

clockwise. Each plane has a heading with respect to tower, an angle of pan, an angle of tilt, and a current speed. Location of plane is updated once a second. That is done by the main program using a thread that will tell the plane to update itself and then sleeps for a second. Each plane type has its flight characteristics that are used in simulation. Some of the characteristics of the plane are minimum speed, maximum speed, acceleration rate, deceleration rate, and the course that it will follow. The course is a stack that holds increments of changes in the speed, the direction, and the ascent of the plane. Each entry in the stack will tell the plane how to change its parameters every second. There is currently a maximum and a minimum set on each of the increments. On the direction and ascent there is a six degree maximum set. With the acceleration rate there is a minimum and maximum set for each type of plane. There are many functions that are used to put together a course for the plane. A few functions that will assemble the course are ascend/descend, and clear to land. Once the function puts together the course information, the plane will start using it the next time it updates its parameters.

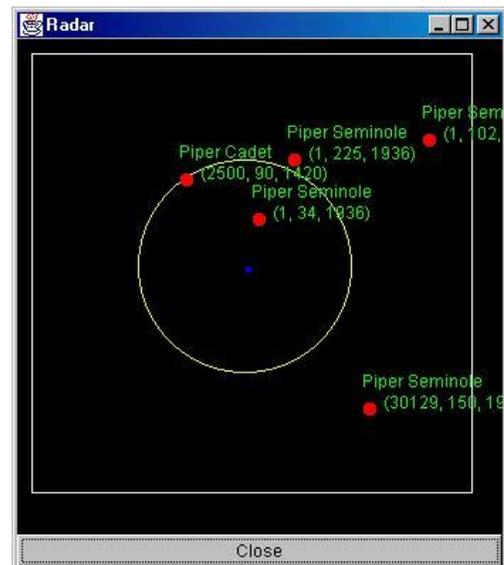


Figure 1 Radar Screen (BRITE)

**BRITE.** ATC is provided with a graphics device that simulates the Bright Radar Indicator Tower Equipment (BRITE). In the TACUND program, the BRITE is called a radar screen (Figure 1). The radar shows a 10-

mile by a 10-mile area of airspace. On the radar screen you can see the planes that are in that area with important data displayed near the plane. You will see important information, like name, call number, height, speed and direction of the plane. The radar gets all the information that it needs to display everything directly from the plane. The viewing area of concern to ATC is limited to 5 miles, which is shown by a yellow circle on the radar screen. The BRITE is updated every second as the plane parameters are updated.



Figure 2: Screen Shot

**Tower view and compass.** On the main screen of TACUND, ATC is provided with a view out of the control tower (Figure 2). The user can manipulate a compass graphics device to view flying planes from 24 different views (Figure 3). On the main screen, you will also be able to see planes are flying though that area. The simulator changes the location and appearance of the plane in flight and displays plane on the screen with appropriate changes in size with respect to the viewer's position. The first item that the TACUND does is to determine if the plane is flying though the displayed airspace. To do that, the program looks at the planes polar coordinate, then it compares it to the polar coordinate of the view that is currently shown on the main screen. If the plane falls in the area of the main screen view, the plane is displayed. The polar coordinate is also used to determine where the plane should be placed on the main screen. Next, the program has to determine the appearance of the plane, which depends on a few items. One of the parameters is the type of the plane. The TACUND can display five different types of plane pictures. Another parameter for consideration is the direction the plane should be

flying through screen. With every plane type, there are eight different plane views corresponding to different sizes and orientations with respect to the viewer. The program decides on a plane using the angle with respect to the screen normal and the direction the plane flight. Then the program has to figure out how big the plane should look on the screen. That is determined by taking the distance of the plane from the tower to compute a scaling factor. Using the scaling factor, the picture of the plane is shrunk by that factor.

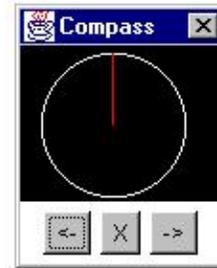


Figure 3. Compass

### ATC Commands

The ATC decides on a choice of a runway and a strategy for landing the planes. Commonly used ATC commands in our system are the following commands to pilots: "touch and go", "enter holding pattern", "ascend", "descend", "left 180", "right 180", "left 360", "right 360", and "clear to land". Commands to pilots and pilot responses to these commands are displayed to provide feedback for the ATC. Interactions between ATC and each plane are kept in separate viewing areas.

**Ascend/Descend.** This is a pair of commands. Each issuance of this command is for 1000 feet and it is used to separate the space between planes. The simulator starts a loop that computes the angle of plane height change based on the plane speed and distance of height adjustment remaining. If this angle is larger than 6 degrees, it changes the plane's tilt angle to 6 degrees. Otherwise, that angle is adopted as the plane's tilt angle. The tilt angles will gradually get smaller and will not always reach perfect 1000 feet change. To avoid indefinite changes in angles, the simulator stops executing this command when the plane reaches within about 50 feet of target height.

**Left/right 180/360.** Similar to Ascend/Descend, each of these four commands is used to move the planes around in the air space. The simulator starts a loop that computes the angle of pan 6 degrees at a time until the plane reaches 180/360 change in direction. Once it

reached its new heading, the plane will maintain that course and begin to fly straight-ahead.

**Enter holding pattern.** ATC uses this command to separate planes at different heights before they can be given permission to land. The default altitude is 2600 feet. Other holding patterns are at successive 1000 feet, e.g., 3600 feet, 4600 feet so on until 7600 feet. The plane will first point towards tower and decrease its speed. Once it gets to the area of the control tower, the plane will go into a loop to circle the tower. The plane will pan 6 degree to the right and continue turning until another command is given by the air traffic controller.

**Clear to land.** This command is issued to a plane and gives it permission to land. First the plane will fly out to an area about 10 miles southeast of the airport on a heading of 170 degrees, this is shown as point 1 in Figure 4. When it reaches that point, the plane will be at 2600 feet and will start to slow down to its minimum speed. Once it has reached the point it will change the direction to go to a point that is about 10 miles south of the airport, this is point 2 of Figure 4. After it gets there, the plane will be on a heading of 350 degrees, which will lead it to the airport. Once it reached a point about 5 of a mile away from the runway, point 3 of Figure 4, the plane will start descending to the ground. Once the plane is on the runway, then it will apply the breaks and well come to a stop on the runway.

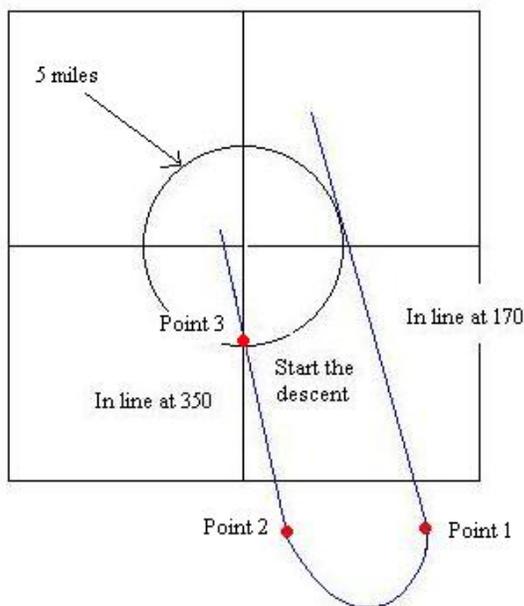


Figure 4 landing diagram

**Touch and go.** This command is issued when the plane lands but without powering down, the plane takes off again. This is used for practice runs. The plane will do the same exact method as if it was planing to land. It will line up with all the same points as before but once it hits the runway, it will not slow down to a stop, but it will maintain power and return to the height of 2600 feet.

## Prospects

TACUND is a simple tower simulator that can be used by an ATC student. But beyond the student use, we want to automate the functions of ATC. The human operator detects urgencies in landing planes, determines operations for safely separating planes, cascades multiple plane landing, and geometrically guides planes to near runways. We will develop an expert system that can perform these tasks. The automated ATC will enable experiments with human cognitive load and will help us design automated tools for ATC use. Another area of advancement is design of pilot agents. This will focus on communication between pilot agents and ATC.

**Acknowledgement:** Parts of this simulator was implemented by many students of Simulation and Math-modeling course taught by Dr. Hexmoor. We want to acknowledge Tanim Ahmed, Daniel Boardman, Xin Ding, Scott Fowler, Dongqing Lin, Vaughn Nerdahl, Kirk Ogaard, and Chris Snow. Our motivation for developing this program in the Aerospace college at the University of North Dakota was to have a program that ATC trainers could use to teach the air traffic controller students approached the computer science department at the UND. We thank Gary Bartelson of UND Aerospace for discussing related issues with us.

## References

- Erzberger, H., 1992. CTAS: Computer Intelligence for Air Traffic Control in the Terminal Area, NASA TM 103959, Ames Research Center.
- National Research Council, 1998. **The Future of Air Traffic Control: Human Operators and Automation**, National Academy Press.
- BAO. Ltd, 1992. TOWER: Real-Time Control Tower Simulation. Champaign, IL.