



A THREE-DIMENSIONAL HEADS-UP PRIMARY  
NAVIGATION REFERENCE DISPLAY FOR  
PARATROOPERS PERFORMING HIGH  
ALTITUDE HIGH OPEN JUMPS

THESIS

Brian A. Balazs, First Lieutenant, USAF  
AFIT/GCS/ENG/03-02

DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY

---

---

*AIR FORCE INSTITUTE OF TECHNOLOGY*

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense or United States Government.

A THREE-DIMENSIONAL HEADS-UP PRIMARY  
NAVIGATION REFERENCE DISPLAY FOR  
PARATROOPERS PERFORMING HIGH ALTITUDE HIGH  
OPEN JUMPS

THESIS

Presented to the Faculty

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Computer Science

Brian A. Balazs, B S

First Lieutenant, USAF

March 2003

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

A THREE-DIMENSIONAL HEADS-UP PRIMARY  
NAVIGATION REFERENCE DISPLAY FOR  
PARATROOPERS PERFORMING HIGH ALTITUDE HIGH  
OPEN JUMPS


THESIS

Brian A. Balazs, B. S.  
First Lieutenant, USAF

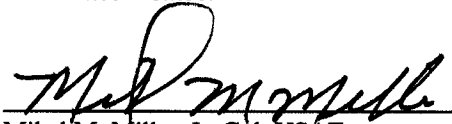
Approved:

  
\_\_\_\_\_  
Timothy M. Jacobs, Lt Col, USAF  
Chairman

13 Mar 03  
Date

  
\_\_\_\_\_  
Karl S. Mathias, Lt Col, USAF  
Committee member

13 MAR 03  
Date

  
\_\_\_\_\_  
Mikel M. Miller, Lt Col, USAF  
Committee member

13 March 03  
Date

## ACKNOWLEDGEMENTS

Committee members: Lt Col Timothy Jacobs, Lt Col Mikel Miller, and Lt Col Karl Mathias.

Fellow SIEVErs: Brian Bakshas, Carolyn Bartley (mouse clicker at my defense too), Rich Gonzalez, Eric Hunt, Andy Kowalchuk, Ben Musial, Rick Roell, and Ken Wilson.

People I bothered for information: Professor William Baker, Mr. Jeffery Craig, Mr. Jerry McDonald, Maj John Raquet, SSgt. Joshua Smith, Mr. Rob Stephens, Mr. Allen Page.

Research free-jazz fix provider: Ken Vandermark.

One person *can* do something: Phil Elvrum a.k.a. the Microphones.

Friendly/Funny/Lively mathematician: Dr. Potoczny

People who kept me from doing research: Mostly Larissa O'Brien, but also Clint and Madelyn Carr (not the Carrs), Chris Clark, Danielle Burrows, Chris and Katie Blanchette, Maria Bavaro, occasionally Mark and Maria Reith, fellow students, John Lily (from language classes – former Hungarian fighter pilot turned dirty old man), my dogs Gluten and Mentos, and the two bartenders at Walnut Hills that always pushed their cheap beer and free pizza (Fridays from five to eight) on me.

One of the few people to lay eyes on every page of this paper: LaVonne Allen.

Craziest person I ever met: Matt (Some guy I met on New Year's Eve 2003. Although he didn't help me, I think his craziness (which far surpasses the "Navy seals!" guy and the guy who was abducted by aliens) is worth acknowledgement).

People supported me: Larissa O'Brien, and my brother, sister, and parents (if I put one I have to put them all, you know how it is...).

# TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	v
TABLE OF CONTENTS .....	vi
LIST OF TABLES .....	ix
ABSTRACT .....	x
1. Research Introduction.....	1
1.1 Introduction .....	1
1.2 Background.....	2
1.3 Research Focus .....	4
1.3.1 Objectives .....	5
1.3.2 Assumptions .....	5
1.3.3 Approach .....	6
1.4 Summary.....	7
1.4.1 Document Overview .....	7
2. Background.....	9
2.1 Introduction .....	9
2.2 Navigation via GPS .....	9
2.2.1 Related Research in GPS Navigation .....	13
2.3 Parachute Travel.....	14
2.3.1 Related Work in Parachute Travel.....	18
2.3.1.1 Sherpa and Mission Planner .....	19
2.3.1.2 OPANAS .....	20
2.4 Graphical Navigation Displays.....	23
2.4.1 Related Work in Graphical Navigation Displays.....	23
2.4.1.1 Current Military Interface Standard.....	24
2.4.1.2 Highway-in-the-Sky Technique.....	25
2.4.1.3 Tunnel-in-the-Sky Technique .....	26
2.5 Three-Dimensional Graphics.....	28
2.5.1 Three-Dimensional Graphics in Navigation Displays .....	30
2.6 Summary.....	32
3. Requirements and Methodology.....	33
3.1 Introduction .....	33
3.2 Research Objectives .....	33
3.2.1 Three-dimensional Primary Navigation Reference Display .....	33
3.2.1.1 Elements Essential to HAHO Navigation Display .....	34
3.2.2 Night Vision Goggle Integration .....	38
3.2.2.1 Heads-Up Display Constraints .....	39
3.2.2.2 Hardware Constraints .....	40
3.3 Research Challenges .....	42
3.3.1 GPS Precision and Update Frequency .....	42

3.3.2	Parachute Canopy Performance Differences .....	43
3.4	Summary.....	44
4.	Design and Implementation .....	45
4.1	Introduction .....	45
4.2	Software Design .....	45
4.2.1	Data Section.....	46
4.2.2	Display Section.....	47
4.2.2.1	Java3D .....	48
4.2.2.2	Display Section Classes .....	49
4.2.3	Information Gathering Section .....	51
4.3	Software Implementation.....	52
4.3.1	Implementation of Classes.....	53
4.3.1.1	MyScene .....	54
4.3.1.2	Tunnel.....	55
4.3.1.3	Edge.....	58
4.3.1.4	MyView .....	58
4.3.1.5	SerialReader .....	61
4.3.1.6	GetInfoFrame.....	62
4.3.1.7	SmoothPosition.....	63
4.3.1.8	Queue.....	66
4.3.1.9	MPosition .....	66
4.3.2	Interaction of Classes.....	66
4.4	Summary.....	68
5.	Evaluation and Results .....	69
5.1	Introduction .....	69
5.2	Evaluation Process.....	69
5.2.1	Modified Evaluation System .....	70
5.3	Evaluation.....	71
5.3.1	Simulation Testing.....	71
5.3.1.1	Results .....	73
5.3.1.2	Test Environment .....	80
5.3.1.3	Test Limitations.....	81
5.3.2	Other Test Notes.....	81
5.4	Summary.....	82
6.	Conclusions and Recommendations .....	83
6.1	Introduction .....	83
6.2	Research Impact .....	83
6.3	Outline of Future Work .....	83
6.4	Summary.....	84
	BIBLIOGRAPHY .....	86
	VITA .....	88

## LIST OF FIGURES

Figure 1. Heads-Up Display Integrated with NVGs [STS03] .....	4
Figure 2. GPS Space Segment [MIS01] .....	10
Figure 3. GPS Control Segment [MIS01].....	12
Figure 4. Overhead View of Flight Data [BAR00] .....	14
Figure 5. Parachute Canopy Characteristics [BER02].....	15
Figure 6. Overhead View of Parachute Travel .....	17
Figure 7. Cross-Section of Parachute Travel .....	18
Figure 8. MMIST Mission Planner Software [MMI03] .....	20
Figure 9. OPANAS In-Flight Display [OPA03].....	22
Figure 10. Display Following Current Standard [BAR97].....	25
Figure 11. Typical Highway-in-the-Sky Display [SNO99].....	26
Figure 12. Example Tunnel-in-the-Sky Display [BAR97].....	27
Figure 13. Basics of Three-Dimensional Graphics [JAC02] .....	29
Figure 14. Clipping in Three-Dimensional Graphics [JAC02].....	30
Figure 15. Inclusion of Heads-Up Display .....	39
Figure 16. Example Wind Forecast Data.....	47
Figure 17. Basic Java3D Scene Graph [J3D99] .....	48
Figure 18. Tunnel Composed of Edge Objects.....	50
Figure 19. Simplified Class Diagram of Reference System .....	54
Figure 20. Greedy Algorithm Used to Determine the Best Path .....	56
Figure 21. Waypoints Define Path Shape.....	56
Figure 22. Effect of Speed on Separation of Tunnel Edges.....	57
Figure 23. Four Objects (left) Used to Create One Rectangle (right).....	58
Figure 24. Navigation Reference with Textual and Graphical Information .....	59
Figure 25. Screen Shot of GetInfoFrame Window .....	63
Figure 26. Visual Demonstration of Algorithm Results .....	65
Figure 27. Average Distance by User. Only Two Users Exceeded Two Meters.....	74
Figure 28. Average Distance and Standard Deviation of Subjects by Jump .....	75
Figure 29. Maximum Distances from Desired Path, by Subject.....	76
Figure 30. Average Distance of Subjects from Desired Path .....	77
Figure 31. Subject 21's Difference from the Desired Path.....	78
Figure 32. Distance from Path for Jump 1. Turn Indicated with Vertical Line .....	79
Figure 33. Distance from Path for Jump 2. Turns Indicated with Vertical Lines .....	80



## LIST OF TABLES

Table 1. Conversion of Geodetic Coordinates to Distance.....	61
Table 2. Smoothing Algorithm.....	64

## ABSTRACT

The Department of Defense (DoD) relies on the para-dropping of resources to meet different objectives in order to accomplish missions during peace-time, war-time, or military operations other than war. The resources dropped to the ground via parachute range from supplies and equipment to the most valued asset, people. Tactics have been developed to increase the safety of troops parachuting into areas of conflict. These tactics include high-altitude high-opening (HAHO) jumping and night jumping.

HAHO jumping allows paratroopers to travel large distances in the air away from the path of the delivering aircraft. While night jumping, done with the aid of night vision goggles (NVGs), provides paratroopers with the cover of night. Both of these tactics aid in avoiding detection. These techniques, however, have their drawback: low cloud cover and fog can often delay mission accomplishment due to a lack of visibility. However, low cloud cover and foggy conditions also provide a tremendous aid in covert insertion missions and enhance the element of surprise.

This research introduces a novel application combining three-dimensional graphics and GPS for a primary navigation reference for paratroopers. It uses three-dimensional graphics to realistically portray a paratrooper's movement in the physical world, measured by GPS, as movement in a computer generated scene. This reference, presented as a heads-up display on the NVGs paratroopers already wear, facilitates mission accomplishment in cloudy and foggy conditions. Evaluation of a prototype system validates the effectiveness of such a three-dimensional navigation reference for paratroopers.

# A THREE-DIMENSIONAL HEADS-UP PRIMARY NAVIGATION REFERENCE DISPLAY FOR PARATROOPERS PERFORMING HIGH ALTITUDE HIGH OPEN JUMPS

## *1. Research Introduction*

### **1.1 Introduction**

The Department of Defense (DoD) relies on the para-dropping of resources to meet different objectives in order to accomplish missions during peace-time, war-time, or military operations other than war. The resources dropped to the ground via parachute range from supplies and equipment to the most valued asset, people [LOP02].

People are used effectively in combat, however, casualties and loss of life frequently occur. While losing troops in combat is always considered tragic, it is accepted as part of war. Likewise losing troops en route to combat is tragic, it is however, considered unacceptable. Therefore, the safe and quick transport of troops to the area of conflict to perform their jobs of support and execution of combat is essential. Combat is, by nature, dangerous, and even literally “dropping” troops into place can be hazardous.

Given that people are the military’s most valued asset, and that they are more valuable in combat than in transport, it follows that effort be to made to move troops as safely as possible. Tactics have been developed to increase the safety of troops parachuting into areas of conflict. These tactics include high-altitude high-opening

(HAHO) jumping and night jumping. HAHO jumping is given its name because of the height from which the paratrooper is released, a high altitude of 25,000 feet or more, and a similarly high parachute canopy opening. This technique allows paratroopers to travel large distances in the air away from the path of the delivering aircraft. Night jumping, done with the aid of night vision goggles (NVGs), provides paratroopers with the cover of night. Both of these tactics aid in avoiding detection.

## **1.2 Background**

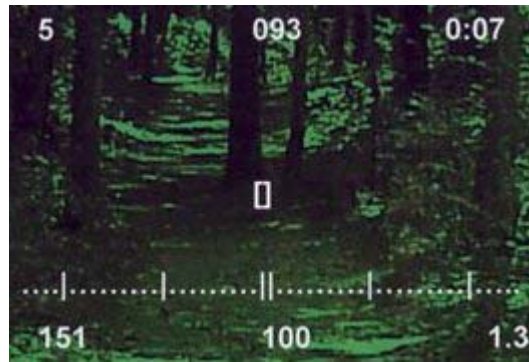
Both HAHO jumping and night jumping with NVGs have taken place for years. Together these tactics have decreased a paratrooper's chance of detection and overcome the formerly limiting characteristic of darkness. In the past darkness regularly restricted the hours during which paratroopers could operate. Though this is no more, there still are naturally occurring events like clouds, fog, and wind that preclude parachute operations. Some can be overcome with technology, while wind cannot. Parachute travel is highly reliant on wind because it directly affects a parachute's path of travel. As a result extremely high winds, which make parachuting unpredictable, and therefore unsafe, will continue to rule out operations.

Unlike wind, clouds and fog do not directly affect a parachute's path of travel; it is independent of cloud cover. Clouds and fog affect a paratrooper's ability to navigate by making it difficult or impossible for him to see terrain features, even with NVGs. Furthermore, immersion in extremely thick clouds or fog can result in disorientation due to the homogeneity of appearance. Unfortunately, the very issues that make parachute operations infeasible during these conditions prove advantageous if overcome.

The ability to operate parachute missions unhampered by such weather conditions can be quite advantageous militarily. First of all, inclement weather provides extremely good cover for paratroopers, because, just as they cannot see terrain features, individuals on the ground cannot see the paratroopers. Furthermore, its current regard as infeasible implies the enemy is unlikely to expect a parachute mission, which adds to the element of surprise. Furthermore, it is human nature to withdraw to shelter during bad weather, making it easier to take advantage of the situation.

The higher the altitude of a troop's release, the higher the probability of inclement weather existing between the release and the drop points. This is simply because there is more physical space in which foul weather could exist. As such, HAHO jumping is heavily subject to weather conditions. If a HAHO paratrooper were given the ability to navigate to his target no matter the weather conditions, the weather would fail to hamper parachute missions, resulting in a military advantage.

Current attempts to overcome adverse weather conditions often include the use of heads-up displays. These displays present position and velocity information produced by a Global Positioning System (GPS) receiver attached to the paratrooper's person and are integrated into NVGs or helmets already worn by paratroopers. They are simple, often text-based and not very intuitive. Figure 1, a navigational display, is one example of a heads-up display integrated with NVGs currently on the market. It presents the user's active waypoint, bearing, time to go, distance to waypoint, track-made-good, and ground speed information, as well as a course tracking bar, though one may not know it at first glance.



**Figure 1. Heads-Up Display Integrated with NVGs [STS03]**

This un-intuitive nature is a major problem with current navigation references for paratroopers. Furthermore, they are, for the most part, real-time displays that display position information as it is received. Instead of giving a paratrooper a predictive display which gives him a glimpse at his future path, these real-time displays only present current information. Although a real-time display can be effective, it provides only a limited amount of information to a paratrooper. Display techniques are discussed further in chapter two.

### **1.3 Research Focus**

The primary focus of this research is the design, implementation, and evaluation of an intuitive three-dimensional display to serve as a primary navigation reference for paratroopers. This navigation reference is intended to serve as a heads-up display integrated into night vision goggles used by paratroopers with high altitude high opening parachutes during inclement weather. A significant portion of the research involves the

development of a software system to compute and display the optimal path to take a paratrooper from his release point to his target.

### **1.3.1 Objectives**

This research has three main objectives. The first objective is the development of a three-dimensional navigation display. The second is to convincingly represent the user as traveling through that three-dimensional space. The final objective is the development of that display in order for it to serve as a heads-up display with NVGs.

The first objective involves the gathering of information necessary to determine the optimal path, then determining the optimal path, including heading and waypoints, and displaying the path in a manner that is intuitive and easy to follow for even untrained users. A three-dimensional space representing the space the paratrooper must travel through is built to accomplish this. The second objective is accomplished by echoing the user's gross movements, or position changes in the physical world, as movements within the display's three-dimensional space. This is done with the help of a GPS receiver. The final objective, development of the display for it to serve as a heads-up display with NVGs, involves ensuring that the display already developed performs effectively given the constraints inherent with the integration of the display with NVGs and the fact that the platform must be mobile enough to travel with a paratrooper both in an aircraft and under canopy.

### **1.3.2 Assumptions**

Due to the wide variety of situations in which a paratrooper may use a navigation reference some assumptions have been made. First of all, it is assumed that the user's

designated release point is that point at which his parachute is open and he begins to fly under its aid. It is also assumed that the wind speed and direction information will be gathered from forecasts, and that information is accurate enough to meet the needs of a paratrooper's primary navigation reference. The final assumption is that a GPS receiver is adequate to realistically present the user's movements to him.

### **1.3.3 Approach**

The software application portion of this research consists of several finite steps which must be accomplished in a pre-defined sequence in order for the display to work correctly. Initially the necessary information is gathered from the user via a graphical user interface (GUI). This GUI is presented to the user at the activation of the program. It gathers information from the user regarding the physical location of his release point and target, the characteristics of his canopy, and the forecasted winds.

This information is used to determine the heading, or series of headings, the user must take, and the position at which he must obtain each heading, in order to arrive at his target. The positions at which the user must obtain certain headings are considered waypoints. The unique path that connects all the waypoints is considered the path that the user must maintain, and is the path displayed to the user. The path is then displayed to the user within a three-dimensional space representing the physical world. The user's position within that space is determined and he is placed accordingly.

The user's position changes within the physical world are reflected by the changing of his position within the display. The user's changes in position are known through the use of a GPS receiver attached to his person. When the GPS receiver



indicates the user has changed position the corresponding view of the display presented to the user is also changed accordingly. This presents a realistic view of the world to the user by which he can navigate instead of navigating with physical world references.

## **1.4 Summary**

The primary objective of this research is the design, implementation, and evaluation of an intuitive three-dimensional display to serve as a primary navigation reference for paratroopers. During the accomplishment of this objective several other objectives were met. These objectives include the development of a three-dimensional navigation display, representing the user as traveling within the display's three-dimensional space, and production of a display with the ability to serve as a heads-up display with NVGs.

Although this research does not provide a fully operational version of a primary three-dimensional heads-up primary navigation reference display for paratroopers using HAHO parachutes, it does provide the groundwork for such an application both in terms of software engineering, and visual display.

### **1.4.1 Document Overview**

The remainder of this paper is divided into five chapters (chapters two through six). Chapter two provides a brief look at background fields pertinent to the problem of developing a three-dimensional heads up display for paratroopers performing HAHO jumps. A description of related work done in each field is given. Chapter three provides the requirements of a paratrooper's primary navigational reference and issues in the design and implementation of such a system. Chapter four provides the detailed design

and implementation of the system developed for this thesis. Chapter five describes the evaluation techniques employed during for this thesis. It also presents and interprets the results of aforementioned evaluations. Chapter six, provides a conclusion and discusses options for future work.

## ***2. Background***

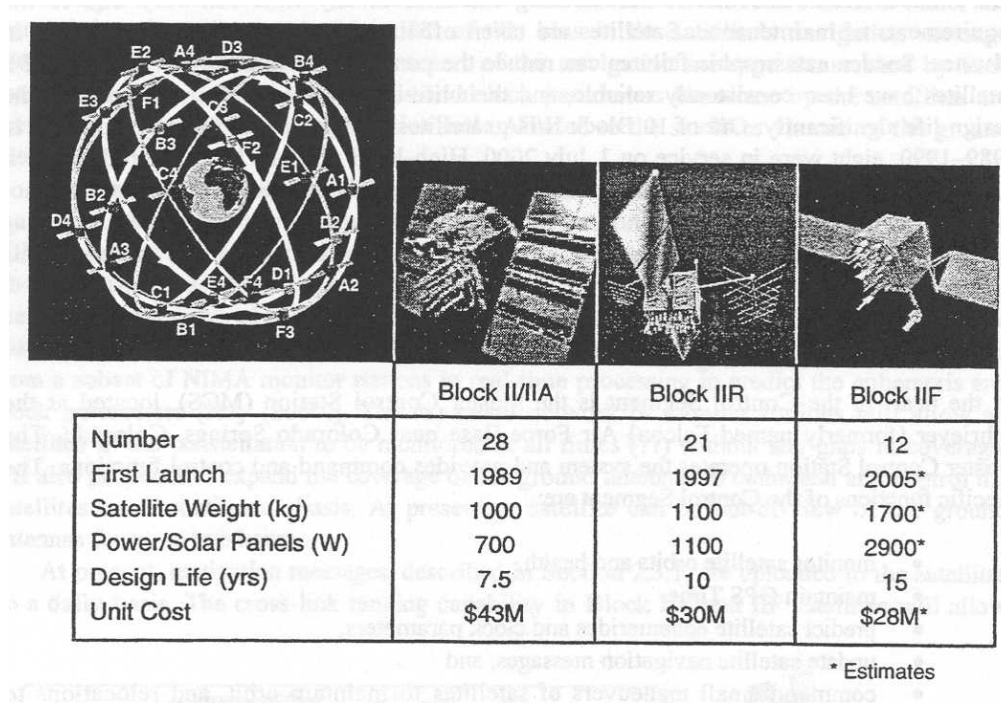
### **2.1 Introduction**

The design and implementation of a primary navigation reference for paratroopers using HAHO parachutes required understanding of four distinct areas. The areas are navigation via the global positioning system, parachute travel, graphical navigation displays, and three-dimensional graphics. These areas, and related work or research in each, are described in this chapter.

### **2.2 Navigation via GPS**

The global positioning system is a DoD system designed to offer the U.S. military extremely accurate three-dimensional location information (latitude, longitude and altitude), velocity and precise time [FOR03]. The system, first declared operational in 1995, consists of three segments: space, control, and user. The DoD is responsible for the space and control segments, while the user segment consists of both military and civilian user equipment.

The space segment is comprised of a satellite constellation with a baseline of twenty-four satellites in circular orbits of 25560 kilometers, a period of nearly 12 hours, and stationary ground tracks [MIS01]. The constellation, shown in Fig. 2, provides users with a clear view of the sky access to at least four satellites at a time from almost anywhere in the world. Each satellite in the constellation continually transmits signals using two frequency bands which are used by the user segment to determine position, velocity, and time.



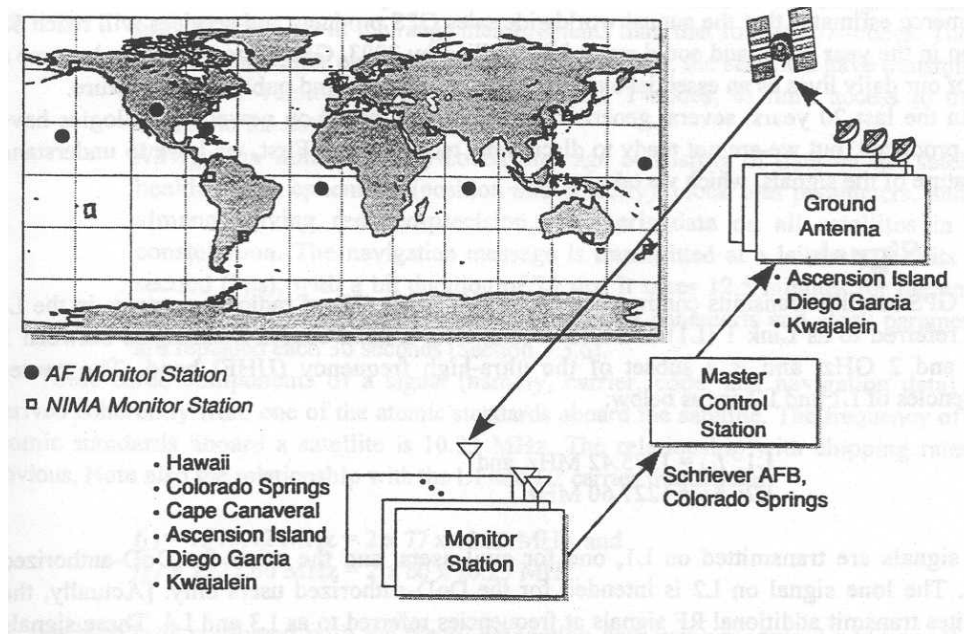
**Figure 2. GPS Space Segment [MIS01]**

The signals transmitted by the satellites consist of three components. The first component is the carrier, a radio frequency sinusoidal signal. The second is called the ranging code. The ranging code is a unique sequence of bits assigned to each satellite which allows the user segment to determine signal transmit time instantaneously [MIS01]. The sequences, called pseudo-random noise sequences are generated in order to allow all the satellites to transmit at the same frequency without interfering with each other [MIS01]. The final component of the signal is the navigation data. The navigation data is a binary message consisting of, most importantly, the satellite's ephemeris (position and velocity) and clock parameters [MIS01].

The GPS control segment, operated by the United States Air Force (USAF), is centered at the Master Control Station (MCS) at Shriever Air Force Base. The specific functions of the control segment are to:

- monitor satellite orbits and health,
- maintain GPS Time,
- predict satellite ephemerides and clock parameters,
- update satellite navigation messages, and
- command small maneuvers of satellites to maintain orbit, and relocations to compensate for failures, as needed [MIS01].

To accomplish this, the MCS is augmented by five monitoring stations spread longitudinally across the globe. The five monitoring stations, located in Ascension Island, Diego Garcia, Kwajalein, Hawaii, and Colorado Springs are unmanned satellite tracking facilities operated remotely from the MCS. The data gathered from the facilities is used to estimate and predict the satellite orbits and clock biases [MIS01]. Co-located at three of the stations are antennas for communicating with the satellites. Communication with the satellites is necessary for the synchronization of the satellite clocks to maintain GPS Time.



**Figure 3. GPS Control Segment [MIS01]**

The final segment of GPS is the user segment. While the aforementioned segments are concretely defined and maintained, the user segment is not. The user segment of GPS is more nebulous than the other segments because it is the equipment used to capitalize on GPS. This equipment ranges from hand-held civilian receivers to the OnStar system in automobiles [ONS03], to the guidance systems in precision bombs. In all cases the user equipment receives signals transmitted by GPS satellites. They then perform measurements of signal transit time and Doppler shift, decode the navigation message to determine satellite position, velocity, and clock parameters. That information is used finally to estimate the user position, velocity and time.

### **2.2.1 Related Research in GPS Navigation**

The benefits of GPS and GPS aided navigation are well known. The corporate, academic, and military communities worldwide are continually enhancing products and processes developing new products built around GPS. One such GPS based product is an aircraft primary flight display developed by the GPS Lab at Stanford University. The display presents a three-dimensional depiction of the world with the aircraft's desired path shown as a series of hoops [BAR00]. The aircraft's position in the physical world, obtained via three GPS receivers, is presented in real-time within the display's three-dimensional world. This scheme provides intuitive guidance with the added advantages of trajectory preview and enhanced special awareness [BAR00].

During field tests the GPS flight guidance system aided pilots in achieving an order of magnitude reduction in cumulative flight technical error on approaches over the conventional localizer needle display [BAR00]. The GPS flight guidance system is based on known position data (such as the position of an airport's runway) and the information gathered from GPS receivers mounted on the aircraft. The conventional localizer needle display, on the other hand, is an analog display representing an aircraft's distance from an airport's localizer beam during approach. Figure 4 shows an overhead view of the paths taken by test aircraft. The thick dark line in the figure represents the approach made using the localizer beam and localizer needle display. Although it is almost entirely within a one-dot range (within two degrees of desired path) of the course deviation indicator needle, and meets the highest standards for certification as an Airline Transport

Pilot, one can easily tell the marked improvement during three approaches when the pilot used the GPS based guidance system, which are represented by thinner lines [BAR00].

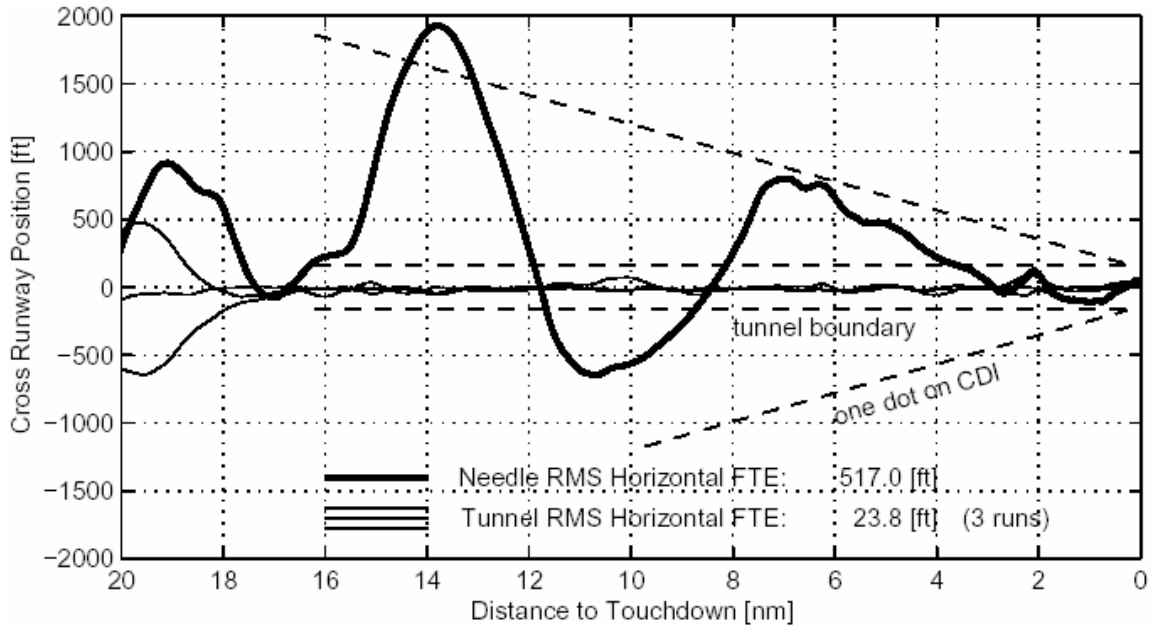


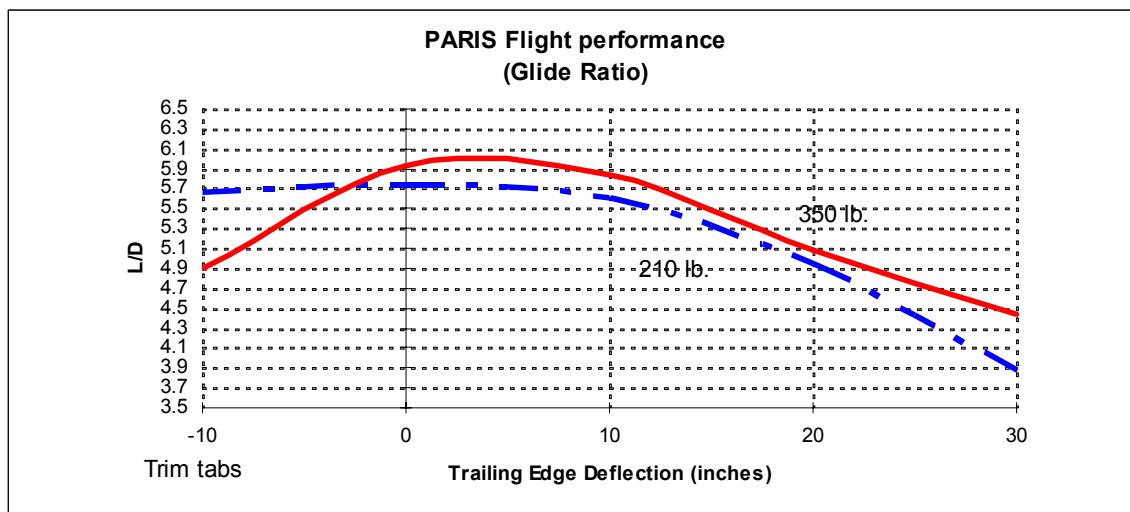
Figure 4. Overhead View of Flight Data [BAR00]

### 2.3 Parachute Travel

The second area of knowledge required for the design and implementation of a primary navigation reference for paratroopers using HAHO parachutes is parachute travel. An understanding of this area is essential to the display's development because a paratrooper's movement is limited by, and dictated by, the parachute canopy the paratrooper is using. Furthermore, different canopy designs perform differently. A round canopy for instance greatly slows an object's descent, but horizontal movement is limited. A square canopy, such as the HAHO parachutes used by paratroopers, allow for traveling great distances horizontally.



Since each model of parachute canopy has distinct performance characteristics a predictive primary navigation reference for paratroopers needs to be configurable. It must be configurable in order to predict the paratrooper's possible path. Such a display would be useless if it predicted the user traveling faster or farther than his physical limits. The weight of the object suspended from a parachute canopy also must be accounted for because it affects a parachute's performance characteristics, such as glide ratio. Glide ratio is the ratio of distance traveled horizontally to distance traveled vertically. Figure 5 shows the relationship between trailing edge deflection and glide ratio for two weights suspended from one canopy (PARIS by Para-Flite) [BER02].



**Figure 5. Parachute Canopy Characteristics [BER02]**

Wind also greatly affects parachute travel, affecting all canopies in the same manner. Wind pushes the parachutes as if the canopies are attached to the wind. For example, if the wind is blowing from the south at 100 meters per second it will push a

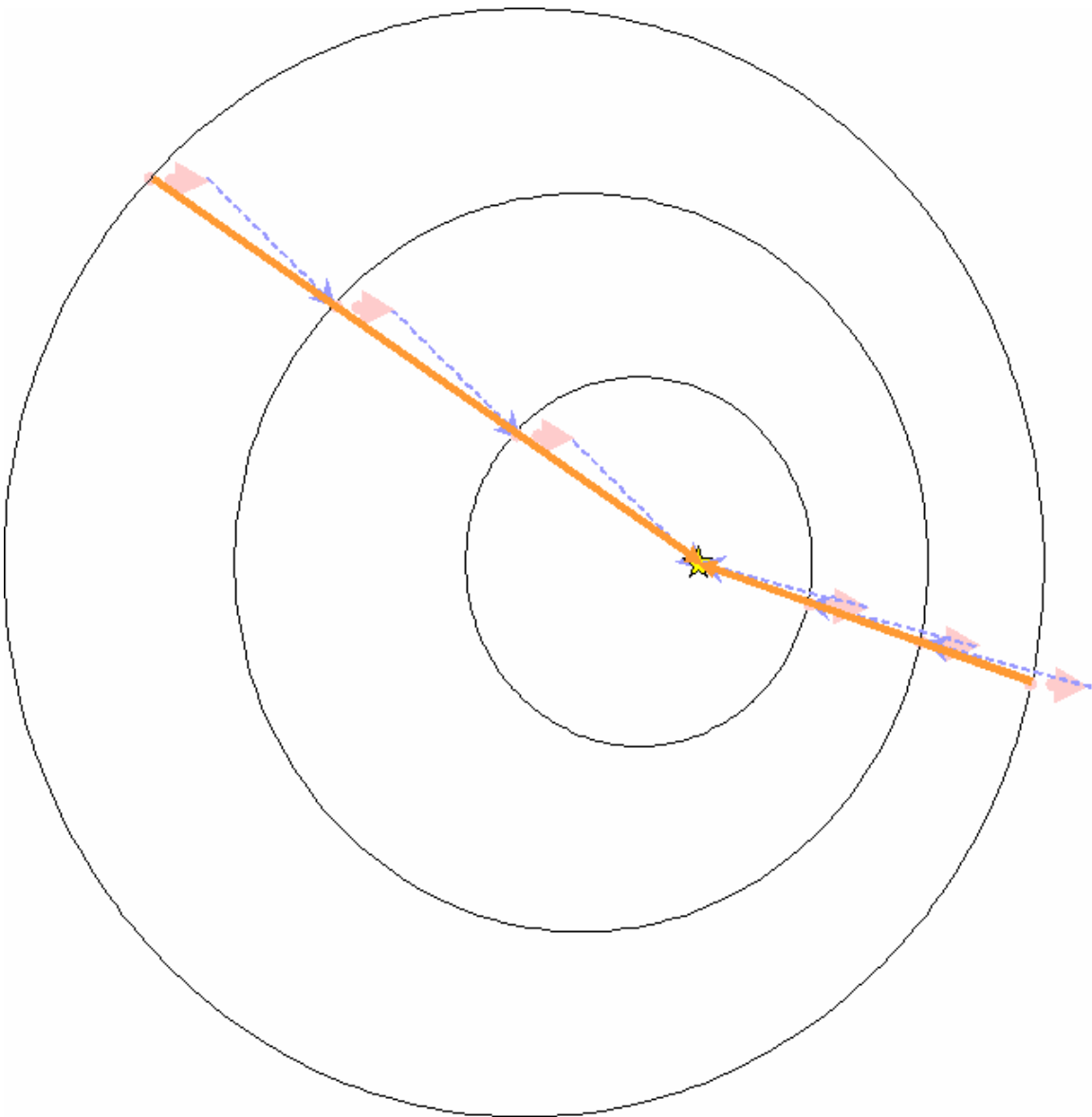
canopy northward at 100 meters per second. The paratrooper can alter his direction of travel by steering his parachute, which constantly travels forward through the air. This does not necessarily mean, however, that the parachute constantly travels forward with respect to the ground. If a paratrooper is using a canopy that travels forward at ninety meters per second, he would be traveling backward at a rate of ten meters per second north if he is facing due south in the wind described above. If, however, he were to face due north he would travel through the air at ninety meters per second and be pushed by the wind at 100 meters per second resulting in a total rate of 190 meters per second north.

The effect of wind on parachute travel is described in Figure 6. The figure shows a target in the middle of the smallest circle. The dark circles represent positions at which a paratrooper following a constant heading can be at and still arrive at the target. The positions represented by each circle symbolize a unique altitude, so that every point on a circle has the same altitude with an inner circle having a lower altitude than an outer circle.

The circles are derived from the target out with a constant (both speed and direction) wind, a constant parachute speed, and a constant heading by the paratrooper. There are two circles at each time interval, the lighter ones representing the distance a paratrooper can travel. The positions of the lighter circles are then shifted by the wind resulting in the darker circles. Although the two steps are performed serially here, the effect is the same as the two done in tandem as in the physical world.

To further demonstrate the effect of wind on parachute travel two constant headings have been displayed in Figure 6. Beginning travel on the left side of the figure

causes a paratrooper to travel with the wind, while beginning on the right forces a paratrooper to travel against the wind. A decomposition of the heading of travel (longer arrows) reveals two components: wind heading (large arrowhead), and user heading (dashed arrow). The wind heading is the same for both headings, only the user heading varies.



**Figure 6. Overhead View of Parachute Travel**

A paratrooper can vary the speed at which he travels using controls on his parachute called brakes. By applying the brakes he slows himself down, but does not change the angle at which he descends. This is significant because it means it is possible for a paratrooper to regain a full speed path if he finds himself below it. An example of this is presented in Figure 7.

### Figure 7. Cross-Section of Parachute Travel

### 2.3.1 Related Work in Parachute Travel

There currently are products, already developed, that fuse parachute travel and navigation. Mist Mobility Integrated System Technology Inc. (MMIST), a private company located in Canada produces a system called the Sherpa, and another called the Mission Planner. The Operational Paratroopers Navigation System™ (OPANAS) was

developed upon request of the French Ministry of defense, and was enhanced by the American company SSK Industries Inc.

#### **2.3.1.1 Sherpa and Mission Planner**

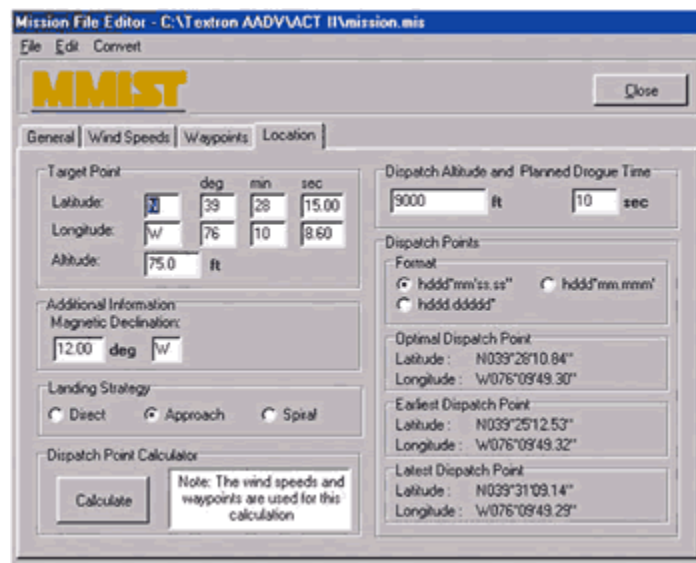
The Sherpa Hi Glide Ram Air Parachute Delivery System is designed to deliver payloads to targets via parachute. The system attaches to parachute controls and steers the parachute to its target. The unit steers itself by either homing in on an ultra high frequency radio beacon or through the use of GPS data [MMI03]. The GPS data is gathered in two ways. First of all, a user must enter target coordinates into the system. Second, the Sherpa unit contains a GPS receiver which it can gather data from on the fly. The system is also able to compute the optimal release point of the payload [MMI03].

The Sherpa, however, contains no display. The system automatically flies its payload to a given target without presenting any reasoning to the payload. This is fine for equipment, but humans often feel uneasy about having no control over their destiny. This is especially true during life-threatening situations, such as falling from the sky.

The Mission Planner is the software application behind the Sherpa system [MIS03]. The Mission Planner is a portion of the Sherpa system that determines the best path and waypoints, it does not, however, perform any steering operations. It is a Microsoft® Windows™-based program that allows the user to enter target location, planned release altitude, and wind speeds, among other information. The Mission Planner uses that information to calculate an optimal, earliest and latest release point, as well as waypoints. It then displays the release and waypoints graphically, and allows for

the waypoints to be edited [MIS03]. Although the Mission Planner can record data during a jump, it cannot display data in real-time.

The Mission Planner is essentially the portion of the Sherpa system responsible for mission planning. It can be used as a stand-alone system, but the system does not aid the user during a mission. One screen shot of the application is presented in Figure 8.



**Figure 8. MMIST Mission Planner Software [MMI03]**

### 2.3.1.2 OPANAS

OPANAS is a parachute navigation system developed by Navocap Ingenierie Electronique, a French company. The system, used by the French Army Agency, Singaporean Special Forces, U.S. Navy, and Russian Forces [NAV03], is designed to aid paratroopers reach their targets during HAHO jumps. OPANAS is a system that can be used during mission planning, training, and execution.

OPANAS is a chest mounted unit that can be used by a paratrooper while wearing NVGs and an oxygen mask [NAV03]. The unit consists of a GPS receiver, compass, backlit graphical display, magnetometer, barometric altimeter, and temperature sensors [NAV03]. The graphical display shows a moving map with wind data, and magnetic heading [NAV03], which is measured in degrees increasing in the clockwise direction with due north as 0/360 [STE44].

The system is programmed before a mission with three-dimensional release, target, and waypoints. Unlike the Mission Planner software application described above, the OPANAS user must manually compute these points based on weather forecast information. The user may, however, adjust the waypoints during flight based on position and wind parameters determined by means of GPS data, the magnetometer, the barometric altimeter, and preprogrammed canopy performance characteristics [OPA03].

During flight OPANAS uses GPS signals and pressure sensors to provide three-dimensional positioning information, and a magnetometer to provide heading information [OPA03]. This information is then used, along with the preprogrammed course data to display the coordinates and displacement of the paratrooper from the planned trajectory [OPA03].

OPANAS's default in-flight display is shown in Figure 9. The display contains six windows labeled A1-A6. Window A1 displays the heading the user should hold on the unit's compass. A2 indicates the horizontal distance between the user and his target point. Window A3 indicates the shortest route to the desired target. A4 displays the current wind direction in twenty-five degree steps (the circle represents the user).

Window A5 indicates the wind speed and the wind direction again. Finally the large window, A6, represents the relative position and orientation of the user as an arrow. The other contents of the window are displayed relative to the user, so that if he turned to the right the contents would move to the left on the display [OPA03].

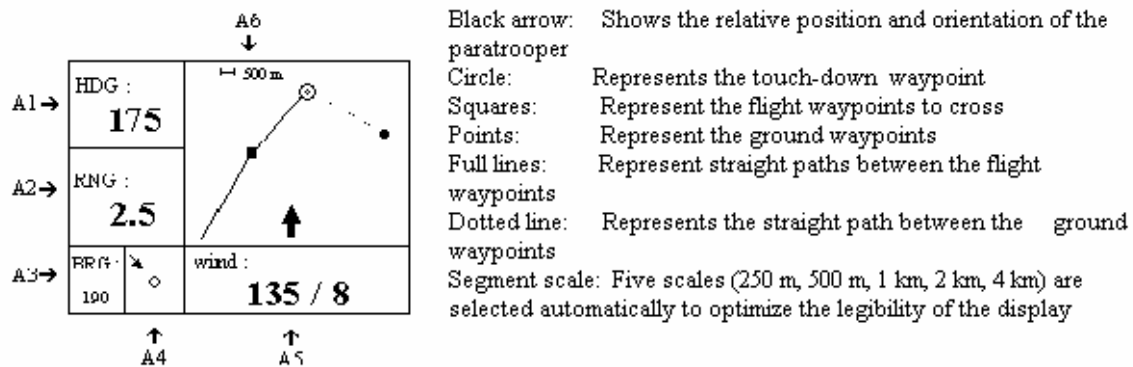


Figure 9. OPANAS In-Flight Display [OPA03]

SSK Industries, Inc, of Lebanon, Ohio augmented OPANAS resulting in a system they call On-Target. On-Target is a mission training, planning and simulation system that integrates OPANAS, Mission Management Planner™ (MMP), a PC-based mission planning application, and ParaSim™, a PC-based virtual reality parachute simulator.

MMP is used as a control panel during simulated training missions [ONT99]. It takes as input target, weather forecast, user, and canopy information and determines positions the user can start at to reach the desired target with a constant heading. It allows for an instructor to adjust conditions to add realism to simulated jumps [ONT99]. Once planning is completed the instructor feeds the information to the ParaSim program,



which also receives input from a user through ripcord, riser and steering toggle sensors [ONT99]. The result is presented to the user via virtual reality goggles.

## **2.4 Graphical Navigation Displays**

Knowledge of graphical navigation displays is necessary to the design and implementation of a primary navigation reference for paratroopers using HAHO parachutes, because the display must convey enough information to the user for him to make correct decisions. A primary navigation reference for paratroopers is directly equivalent to an aircraft's primary flight reference (PFR) which provides enough information to a pilot for her to fly an aircraft by instruments alone. So, it follows that a primary navigation reference for paratroopers expresses enough information for a paratrooper to land safely on target. This requires the display include not only the correct information, but that it correctly conveys it to the user.

The concept of a three-dimensional flight path for aircraft navigation dates back to the Army Navy Instrumentation Program during the 1950s, but until only recently was considered too expensive to field [WIL00]. With the advent of GPS technology and inexpensive, yet powerful, graphical displays three-dimensional displays are becoming more commonplace. Two-dimensional displays, both mechanical and optical, are, however, not uncommon.

### **2.4.1 Related Work in Graphical Navigation Displays**

This paper has already discussed several navigation products, and navigation displays, namely the aircraft flight GPS navigation system developed at Stanford, and the paratrooper specific OPANAS system. The navigation system from Stanford was

accomplished through the use of a three-dimensional tunnel-in-the-sky technique [BAR00], while the OPANAS system applied simpler, two-dimensional text and symbol-based technique. A simple two-dimensional technique, embodied by the Military Interface Standard for Aircraft Symbology, as well as two three-dimensional techniques, highway (or pathway)-in-the-sky, and the related tunnel-in-the-sky are discussed here.

#### **2.4.1.1 Current Military Interface Standard**

Current display standards are a result of a direct conversion of the standards for electro-mechanical instruments to standards for electro-optical displays. Consequently, the current standard is to have digital displays appear very similar to the old standard's analog instruments [SNO99], and have been fairly static over the past fifty years [BAR97], see Figure 10. The result is a display that does not take advantage of the data processing power of today's automated data processing equipment. These displays are quantitative, not qualitative, in that they show raw data, and are non-predictive, in that they display that data in real-time only.

The current military standard for aircraft display symbology, outlined in *MIL-STD-1787B*, is such a standard. Although *MIL-STD-1787B* applies only to aircraft, it is representative of current standards for navigational displays. Direct adaptation of these standards to a paratrooper's navigation display would result in a less than optimal display of little use to paratroopers. The display would convey data in real time, but leave all interpretation of the data to the paratrooper. The approach also forces users to focus on the display more frequently to read data than the two approaches discussed below

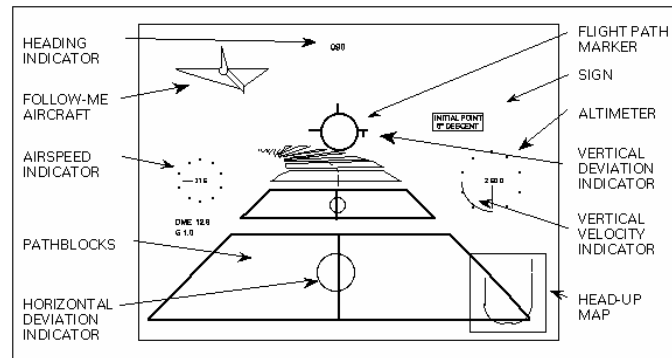
[SNO99]. Furthermore, in high stress situations, the standard display can lead to misreading errors, making it a technique to avoid.



Figure 10. Display Following Current Standard [BAR97]

#### 2.4.1.2 Highway-in-the-Sky Technique

The highway-in-the-sky is a three-dimensional perspective display technique that displays the user's intended route up to forty five seconds into the future [SNO99], see Figure 11. It consists of a string of path blocks drawn in perspective describing the intended route. The path blocks representing a highway emanate from the user's present position. The display can also incorporate *road signs* to alert users of important information [SNO99], as well as a limited amount of two-dimensional data.



**Figure 11. Typical Highway-in-the-Sky Display [SNO99]**

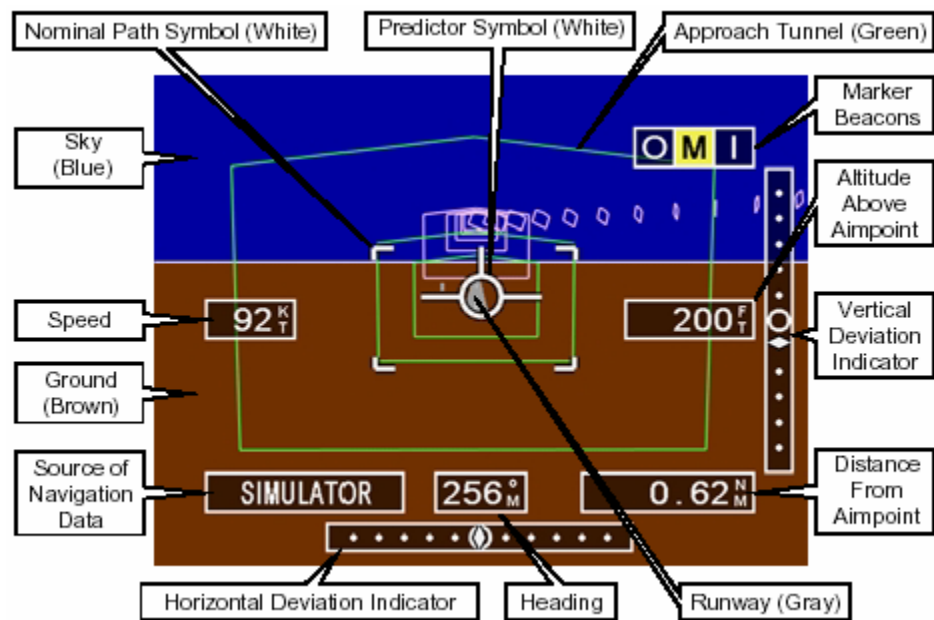
The predictive nature of the highway-in-the-sky allows users to see not only the desired route in real time, but also a visual path to follow. In a study, Snow *et al* found that pilots unanimously preferred this technique to the current military standard [SNO99]. Furthermore, using the technique pilots flying curved, precision, and complex approaches to landing are better able to maintain commanded airspeed, altitude, and heading than using the military standard technique [REI00]. From this we can extrapolate that paratroopers would also prefer, and operate more effectively, using a highway-in-the-sky-type display rather than using a display following the current military standard.

#### **2.4.1.3 Tunnel-in-the-Sky Technique**

The tunnel-in-the-sky is a derivative of the highway-in-the-sky. Like the highway-in-the-sky, the tunnel-in-the-sky is predictive three-dimensional perspective display showing the user's intended path. The main difference between the two techniques is that the highway-in-the-sky technique presents a path on which the user

should travel, while the tunnel-in-the-sky technique presents the user with a tunnel in which he should travel.

The tunnel-in-the-sky technique presents a three-dimensional view of the world (see Figure 12) and the desired flight path enabling users to follow a precise path [BAR99]. This approach gives users pre-cognitive visual boundaries in which they must stay. Giving paratroopers an area in which they should stay is appropriate considering the difficulty a paratrooper has regaining a lost path, especially if that path is above him. This is because a paratrooper is at the mercy of both wind and gravity.



**Figure 12. Example Tunnel-in-the-Sky Display [BAR97]**

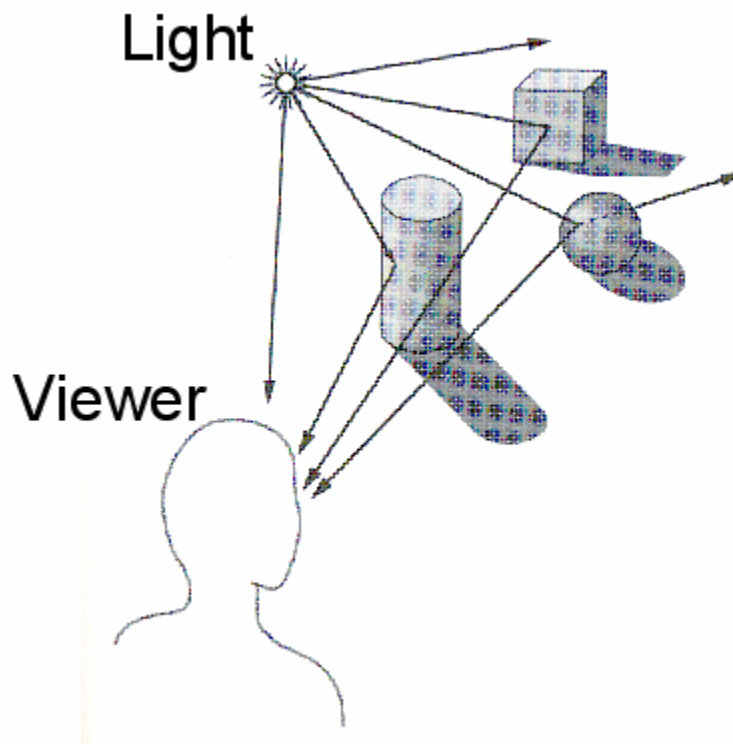
The Stanford and Snow studies described above are just two of many studies that resulted in not only a better performance with, but a preference for, perspective displays, like the tunnel-in-the-sky. The National Aeronautics and Space Administration (NASA)

has recognized the ability of such displays to give aircraft crews clear views of their surroundings in bad weather and in darkness, which could help prevent deadly aviation accidents [BRA99].

## **2.5 Three-Dimensional Graphics**

An understanding of three-dimensional graphics is necessary for the design and implementation of a primary navigation reference for paratroopers using HAHO parachutes. Knowledge of this area is important because an intuitive predictive display, like the tunnel-in-the-sky described above, requires three-dimensional design. The use of the tunnel-in-the-sky technique is desired because it has been proven to be effective for navigation. Furthermore, the use of three-dimensional graphics allows for the building of realistic, perspective-based scenes familiar to users.

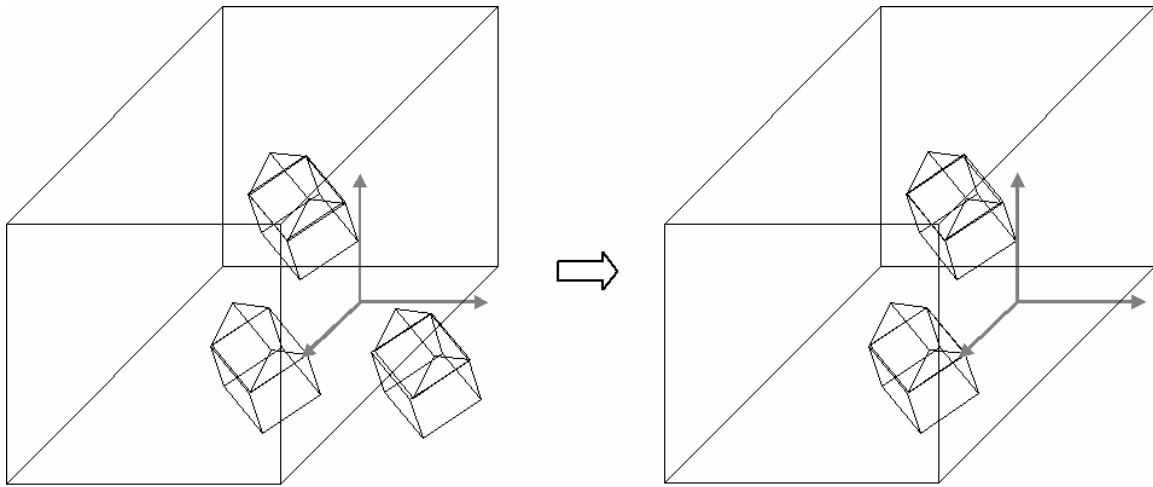
The display of three-dimensional graphics is a computationally intensive process. Three-dimensional graphics are a result of the conversion of geometric models into a two-dimensional pixel array (image buffer), a process called rasterization. The pixel information in the image buffer can only be displayed to the screen after the rasterization process. This process is highly dependent on the shape, placement, and color of the geometric models, as well as the placement of lighting and viewers, among other variables (see Figure 13). It is the consideration of all this information that requires a great deal of computation.



**Figure 13. Basics of Three-Dimensional Graphics [JAC02]**

Since rasterization is so computationally expensive, it is beneficial to perform it only on the portion of the three-dimensional scene that will be displayed. To do this, initial calculations must be performed. These initial calculations, called clipping and projection, take into account the position of the viewer, and the viewer's angle of focus, the angle of view, the maximum distance models remain visible to the viewer, and of course the position of the objects. Then the scene is pared down to the objects visible to the viewer, limiting the amount of calculation that must be performed during rasterization, because it is worthless to compute the appearance of objects that do not appear on screen.

Clipping is pictured below. In Figure 14 there are three houses in the entire scene on the left. The boxed area represents the area visible to the viewer. Only two of the houses are visible, so the house that is not visible is clipped. The result is the portion of the scene that will be rasterized and appear on screen.



**Figure 14. Clipping in Three-Dimensional Graphics [JAC02]**

One technique during this process worth mentioning is the use of the z-buffer. The z-buffer is used to keep track of the placement of objects' depths. If one object is in front of another, closer to the viewer, the z-buffer is used to determine if the closer object blocks the farther object's visibility in whole or in part. The use of the z-buffer facilitates the exclusion from rasterization of not only objects outside of the area of visibility, but also object's whose visibility is blocked.

### **2.5.1 Three-Dimensional Graphics in Navigation Displays**

Three-dimensional graphics are widely used in perspective navigation displays, such as highway-in-the-sky and tunnel-in-the-sky. A prototype tunnel-in-the-sky display



was developed at Stanford that has been used to evaluate operational issues through piloted simulation and flight testing [BAR97]. This display, already discussed above, will also be discussed here with respect to three-dimensional graphics.

The three-dimensional display makes flying by instrument reference safer and easier by presenting an “out the window” view of the world allowing the horizon, runway, and desired path to be seen even when flying in clouds [BAR97]. This display, made possible with the aid of GPS, was kept simple to minimize computational requirements and enhance ease of use [BAR97].

The tunnel was made of simple *hoops* 100 meters wide spaced 500 meters apart on straight segments. The distance between the hoops was reduced to 200 meters on turns to allow the pilot to better see the tunnel, which curved out the side of the display when the aircraft was in a turn [BAR97]. A high contrast between the tunnel and the sky and ground was established to enhance readability. Furthermore, there was a predictor symbol, an abstraction of an aircraft, to help the user determine direction.

The results of the piloted tests of the three-dimensional display proved very positive. The tunnel display was found to have significant advantages over conventional aircraft instrumentation, in particular horizontal and vertical precision and in workload reduction [BAR97]. Pilots were also able to quickly learn to use the tunnel display to fly complex flight trajectories, and were able to make tactical deviations and smoothly rejoin the desired path [BAR97].

## 2.6 Summary

The design, implementation and testing of a primary navigation reference for paratroopers using HAHO parachutes requires the understanding of four distinct areas: navigation via the global positioning system, parachute travel, graphical navigation displays, and three-dimensional graphics.

GPS satellites broadcast complex messages to the earth which can be received anywhere. By performing calculations on the signals from at least four satellites a GPS receiver can precisely and accurately determine its location. This is useful to navigation, because it allows one to determine location, velocity, and time. GPS navigation can be applied to parachute travel as evidenced by the Sherpa, Mission Planner, and OPANAS.

Parachute travel is highly dependent on wind conditions. Both wind speed and heading greatly affect parachute travel. All parachutes respond to wind the same way. Some, however, have the ability to easily change speed and direction. The parachute canopies used by military HAHO paratroopers are examples of the type of canopies that do this. In fact it is almost necessary for paratroopers to steer their canopies when attempting to reach their target. To do this some sort of navigation system is needed.

A graphical navigation display can be designed in a number of ways; however it has been proven in several studies that three-dimensional predictive displays make flying paths in aircraft much easier. One can extrapolate from this that they can do the same for parachute navigation, since parachutes also travel curved, three-dimensional paths. The tunnel-in-the-sky technique has demonstrated itself to be best better suited for aerial navigation than similar three-dimensional perspective-based displays.

### ***3. Requirements and Methodology***

#### **3.1 Introduction**

This chapter discusses specific research objectives and outlines the solution methodology. First, the research objectives are discussed in detail including factors that must be considered during development. In addition, technical issues that surfaced during design are presented.

#### **3.2 Research Objectives**

The primary focus of this research is the design, implementation, and evaluation of a three-dimensional display that can fulfill the role of a primary navigation reference for paratroopers using HAHO parachutes in the navigation to their targets. The objectives of this research include design and development of a three-dimensional navigation display, representing the user as traveling through that three-dimensional space. A secondary objective is the integration of the display with appropriate hardware, rendering it operationally effective as a heads-up display on NVGs. Three-dimensional navigational imaging, GPS positioning, heads-up display design techniques, and software engineering techniques are integral parts of this research.

##### **3.2.1 Three-dimensional Primary Navigation Reference Display**

An aircraft's primary flight reference, as defined in *MIL-STD-1787B*, is "any display or suite of displays or instruments that provide all required information for instrumented flight." One can think of a primary navigation reference for paratroopers as the paratroopers' equivalent of a PFR and extrapolate the requirements of such a

reference from those of a PFR. Therefore, it follows that a paratrooper's navigation reference is a display that provides all required information for instrumented parachuting.

Since a paratrooper's navigation reference must provide all required information for instrumented parachuting it must be both easy to use and informative enough to guide a user to his target without the aid of any other instruments or displays. Optimally, such a reference display could guide a paratrooper to his target without him viewing the physical world. This requires that the display provide sufficient information in an intuitive manner that is easily read in high stress situations, and that the underlying software is robust enough to handle operational conditions.

#### **3.2.1.1 Elements Essential to HAHO Navigation Display**

Paratroopers must be aware of several elements in order to reach the ground and their targets safely. These elements, which are usually monitored manually or by separate instruments, include physical location (latitude, longitude, and altitude) of self and target, wind speed, user weight, and canopy size and shape. An effective primary navigation reference must incorporate at least these essential elements, as well as display the following information: altitude (of user), distance to target, bearing, and waypoints [SMI02].

##### ***3.2.1.1.1 Evaluation of the Tunnel-in-the-Sky Technique***

The tunnel-in-the-sky display approach was chosen over a simple military standard display and a highway-in-the-sky display because it is both predictive and it specifies an area in which the user must travel. Defining, in three dimensions, the area in

which a paratrooper can travel is important because he can travel in any dimension, albeit limited in the upward direction.

The military standard display is not predictive, nor does it provide an intuitive way to define an acceptable area in which the user can travel. It provides only a static, real-time view of position without putting that position in context. The user must then use this static information to travel to predefined waypoints. This approach forces the user to continually monitor position in order to maintain a path, defined by the waypoints, which he cannot see.

Like the tunnel-in-the-sky technique, the highway-in-the-sky technique provides the user with predictive flight path information. It is, however, less graphically descriptive, and therefore less useful. It does show the user the path that he must follow, allowing him to visually navigate the changing path. The highway-in-the-sky technique, however, provides only an optimal path and indicates graphically to the user whether he is away from the path. It does not graphically indicate to the user whether his distance from the path is so great as to keep him from re-attaining the optimal path, which can happen in the field of parachuting.

The tunnel-in-the-sky technique described in chapter two provides a three-dimensional bound for the user. This bound is in the shape of a tunnel defining the user's intended flight path. It shows the user's position in a qualitative, not quantitative, manner. Since a paratrooper's navigational display is such a life-critical application, the tunnel-in-the-sky technique must be augmented with quantitative data. This provides a second way for users to see data to avoid errors in judgement. This thesis augments a

tunnel with quantitative data because even a small misjudgment in the distance from the ground can be life-threatening to the user.

The tunnel in the navigation reference must lead the user to his target; therefore, an effective reference's tunnel shape is reliant on the user's weight (including the weight of his gear), canopy type (size and shape), wind speed and direction, and release and target position. First of all, the user's weight and the canopy type determine the speed at which the user may move. While a parachute's glide slope (the number of feet a parachute can move forward for every foot it descends) remains constant, its forward speed is dependent on the user's weight. While the user and canopy type determine how a parachute can move, wind speed and direction affect how it does move. Wind affects a paratrooper while airborne by carrying him with it. So it is essential to determine the effect of the wind on the paratrooper's flight path. The final variables, release and target position, do not directly affect the shape of the flight tunnel, but they do affect the tunnel's placement. They simply represent two endpoints between which the user must travel.

These variables, wind, canopy type, and user weight, are used to determine the shape of a static tunnel through which the user must travel. This tunnel is used to guide the user between two specified endpoints, his release point and his target point. In this research, once the tunnel's shape has been determined, it is static. The tunnel does not change shape based on the user's position once he departs the release position. The user's view of the tunnel is, however, determined by his position.

The tunnel's shape is not dependent upon the user's position because it is intended to be like a road that will take him from one point to another. Only instead of a road he must travel on, it is a tunnel he must travel through. The desired release and target points are limited to the target defined by the user. The pertinent factors, listed above, are then used to determine an optimal path from the release point to as close as possible to the target point. The path indicates the path the user should take to his target. A tunnel is then built around that path, resulting in an optimal tunnel. This approach provides a simple, intuitive, visual cue to the user to guide the user to his target. It has been demonstrated through extensive operational experience that aircraft pilots learned to fly the tunnel, in tunnel-in-the sky displays, with minimal training [JEN00].

This static optimal tunnel approach was chosen over two other tunnel approaches favored earlier in the research portion of this thesis. In the early stages of the thesis an acceptable/unacceptable tunnel seemed favorable. The premise of this tunnel is that while the user is inside the tunnel he is able to land at his target, while straying outside the target precludes a proper landing. It was found, however, that at higher altitudes, farther from the landing point, an acceptable tunnel would be too large to display relevantly, and the user could easily become lost in the tunnel. Furthermore, it has been noted that users may not be able to always travel within the tunnel – that straying is inevitable. This sentiment was expressed concisely by an American Airlines pilot after seeing a tunnel-in-the-sky display. He commented “It isn’t *if* you fly outside the tunnel, it’s *when* you fly outside the tunnel.” [JEN00] This thesis abandoned the acceptable/unacceptable tunnel approach for these reasons.

Following the abandonment of the acceptable/unacceptable tunnel approach an optimal tunnel approach was accepted. Initially this tunnel was to be built from the target upward, to ensure that it guided the user to the target. This however, was also deemed unfavorable, because it did not ensure the release point could be reached. While both the target and release points are assumed to be pre-determined, the user *must* begin at the release point, and he *ideally* reaches the target point.

### **3.2.2 Night Vision Goggle Integration**

NVGs collect light and focus it on the image intensifier, which absorbs this light energy and converts it into electrons. The collected electron image then strikes the phosphor screen and causes the screen to emit light users can see. The screen emits light in exactly the same pattern and intensity as the light collected at the beginning of the process, showing users a bright image of the dark scene just on the other side of the system. Unfortunately, current NVGs can only display images in one hue.

Developers can integrate a heads-up display with night vision goggles in two ways. The first is to combine the collected image with the heads-up display image before the phosphor screen. This method causes the two images to be displayed to the user as one image, in the goggle's one hue. The second way is to project the heads-up display on a transparent surface within the user's field of view on the user's side of the phosphor screen. The method results in two images, one in front of the other. The amplified version of the collected image will be in the goggle's one hue, but the heads-up image can be in as many colors as the projector allows.



Figure 15 below illustrates the problem with the first approach to integration of a heads-up display with night vision goggles. The integration of the images prior to the phosphor screen results in a monochrome image which may increase the difficulty of gathering information from the images. Integration of the images after the phosphor screen results in two images and allows for the possibility of using hue in the heads-up display that can not be produced by the night vision goggles. For this reason the heads-up display for this paratrooper's navigation reference is integrated with the night vision goggles after the collected image passes through the phosphor screen.

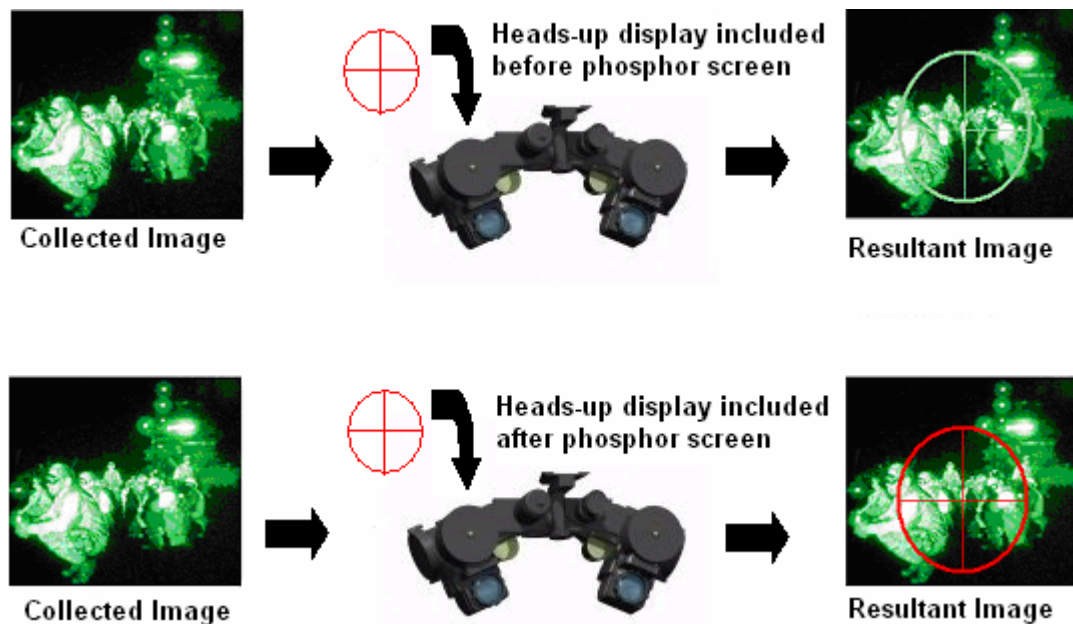


Figure 15. Inclusion of Heads-Up Display

### 3.2.2.1 Heads-Up Display Constraints

The content of a heads-up display must be minimized to provide all essential information while providing as unobstructed a view of the outside image as possible.

The heads-up display developed for this thesis is projected between the user's eye and the phosphor screen depicting the outside image. The placement of the display in front of the outside image, therefore, causes the elements within it to obscure portions of the outside image. For this reason the amount of elements and size of those elements must be minimized.

The notion of minimizing display content, or data ink, in order to use a display in a heads-up nature fundamentally shapes a display. A display considered optimal for a general full screen navigational display may clutter the screen too much to be an effective heads-up display, while the optimal heads-up display may fall short of being considered an optimal general display because of its minimalism. In order to minimize clutter and to avoid obscuring the image behind the heads-up display, it should be made from thin lines, and other thin symbology. Therein lies the challenge: to make a display with as little data ink as possible, yet enough to make the display informative and effective.

### **3.2.2.2 Hardware Constraints**

Further constraints are imposed by data processing hardware, and the mechanism displaying the paratrooper's navigation reference. The rapid increase in the capability of hardware has almost eliminated the thought of it as a constraint to conventional software; it is, however, still a constraint in this situation. This situation is different than most because all of a paratrooper's equipment must be attached to his person.

The size and weight of the computing equipment used to run a paratrooper's reference must be held to a minimum. Paratroopers, especially those with HAHO parachutes, carry a large amount of bulky or heavy equipment limiting movement. To

add a personal computer, or even a lap top computer to that equipment, is excessive. The same is true of the display device. The display device must also comply with night vision goggles which introduce constraints of their own.

The design and development portion of this thesis assumed that computing platforms capable of handling a three-dimensional display such as a primary navigation reference will become sufficiently portable within a short period of time. For example an iPAQ 3975, a lightweight, handheld computer with a 400 MHz Intel X-Scale processor and 64 MB of SDRAM [COM02], is currently on the market. Although it and other similar handheld computers do not have three-dimensional graphics capability, the fact that they are commonplace now and were unheard of a few years ago is a promising sign.

The iPAQ 3975 is mentioned here because it was the target platform of this research's navigation reference in early stages. The iPAQ was the desired platform because of its availability, performance, and size. Furthermore, there is a Java virtual machine available for the platform.

Regardless of platform, the reference has been designed to be displayed as a heads-up display on Specialized Technical Services (STS) AN/PVS-21 NVGs. The STS AN/PVS-21 NVGs are low-profile goggles that facilitate the integration of a heads-up display after the phosphor screen. The heads-up display integrated into the goggles occupies approximately one quarter of the view in one of the eyepieces. The display is limited to that size by constraints within the goggles. The component connecting the computing platform to the goggles must screw into the goggles, and the size of the opening on the goggles limits the possible size of the display.

### **3.3 Research Challenges**

The implementation of the navigation display as a heads-up display introduces expected limitations. The limitations are expected because they come along with every heads-up display. There are, however, challenges unique to the development of a GPS based navigation reference for paratroopers. These challenges include the precision and update frequency of GPS receivers, and the fact that the parachute canopy types affect paratrooper performance. These challenges are acknowledged and addressed in this section.

#### **3.3.1 GPS Precision and Update Frequency**

Although GPS is an excellent system, it is not perfect. GPS receivers cannot continuously determine position, nor are they entirely precise. This thesis recognizes these limitations of GPS and accommodates accordingly.

Although GPS satellites transmit their signals continuously [MIS01] GPS receivers update their position in a discontinuous manner. Receivers do not obtain position; rather, they compute position from data received. Data from four or more satellites are used together to determine the position of the receiver. Gathering the data and performing the computation requires time, and position data is therefore, updated at intervals. The fact that a GPS receiver updates position data at intervals, rather than continuously, combined with the lack of precision in the system, introduces a challenge into the development of a navigation display.

GPS is a fairly precise position measuring tool. The typical error in GPS position reading, however, is 1.3 meters horizontal and 2.0 meters vertical, and even greater in

three dimensions - 2.8 meters. That implies user's position reported by a GPS receiver can be anywhere within a 2.8 meter radius from his true position at every position update. This means that even if a user is completely still relative to the earth his position may be reported at different positions within the 2.8 meter sphere at every update.

Therefore, using raw data from a GPS receiver to display the user position in the display was ruled out. Mapping the user's position to the imprecise position data produced by the receiver causes an unacceptable randomness, or jumpiness in the display. It is unacceptable because it can disorient the user, or cause him to react to erroneous data. Take, for example, a situation where the display reports the user 2.5 meters above his true position, which is already higher than desired. He may descend very quickly in response, only to then be much too low. This is not only irritating to a user, but it is a problem, especially at lower altitudes where there is less time and space for a user to recover.

To solve this problem the raw position data is smoothed before incorporation into the display. The heads-up display developed for this thesis approximates the raw data with a single line and displays a position on that line rather than the raw data. The smoothing of the data also helps deal with the problem of GPS update rates. The smoothing algorithm is further described in chapter 4.

### **3.3.2 Parachute Canopy Performance Differences**

A parachute's canopy type directly affects the speed and performance of a paratrooper. This is especially significant to this thesis because the shape of the tunnel depicting the intended path is dependent on the user's canopy characteristics. The

challenge therefore was to develop a system that recognizes different parachute canopy types and is able to compute the respective tunnel.

This challenge was overcome by obtaining canopy characteristics from a file at the invocation of the application. The user is prompted to enter the name of the file that corresponds to the canopy they plan to use. This approach allows the user to easily run the application with any parachute type, as long as they have information about canopy performance.

### **3.4 Summary**

This chapter provided the requirements and methodology to tackle the research objective of the three-dimensional heads-up primary navigation reference display for paratroopers using high altitude high opening parachutes. The process of evaluation of the display was also outlined along with the metrics used to evaluate the usability and effectiveness of the display. In addition, this chapter discussed other challenges encountered during the research, including the precision and update rate of GPS receivers, and the effect parachute canopy differences have on user performance. Different approaches were described to inform the reader of alternatives considered during research.

## ***4. Design and Implementation***

### **4.1 Introduction**

This chapter covers the design and implementation details for the development of a heads-up paratrooper's navigation reference. The overall design of the system is described, with a focus on the intricacies of the Java3D™ API, the Java interface for programming three-dimensional graphics. Second, this chapter covers the implementation of that design. Attention is paid to aspects unique to this thesis, as well as the accomplishment of objectives and solutions to challenges outlined in chapter three.

### **4.2 Software Design**

The application developed for this thesis is the result of dual goals. The first goal was the development of an operational navigation reference; the second was the establishment of a firm theoretical foundation on which to develop a paratrooper's navigation reference. The software developed for this thesis was designed with these dual goals in mind. The main focus, however, was on the theoretical.

The three-dimensional heads-up paratrooper's navigation reference developed for this thesis was developed in Java, and as result, is object-oriented. The objects in the application fall into three main sections. The sections are data, display, and information gathering. These sections are not as clearly defined and de-coupled as components, but they can be thought of as separate sections. The classes of these three sections will be described here, implementation specifics, however, will be covered in the next section if warranted.

### 4.2.1 Data Section

The data section of this application consists of classes whose job is to store data. They are: `Position`, `SmoothPosition`, `MPosition`, `Parachute`, `Weather`, and `Wind`. A helper class, `Queue`, is also included in this section.

The `Position` class is designed to hold latitude, longitude, and altitude information to define a physical position. `SmoothPosition` is a specialization of `Position`. `SmoothPosition` is designed to keep a moving position from appearing jittery. Since three-dimensional GPS data is only accurate to within a 2.8 meter sphere around an actual position it is necessary to keep a history of past positions to smooth an object's path. `SmoothPosition` accomplishes this. `MPosition` is related to both of the above position classes, but it does not keep latitude, longitude and altitude data. `MPosition` is designed to store three-dimensional position data in meters away from the origin.

Instances of the `Parachute` and `Weather` classes are designed to gather and store information about the user's parachute and weather forecast, respectively. The classes do gather information, but they are primarily designed to store that information, so they are not considered as part of the information gathering section. An instance of the parachute class contains characteristics of a parachute canopy, such as glide slope and forward speed. In an operational version this class would also take into account the user's weight. `Weather` objects are used to store weather forecast data, wind forecasts in particular.

Wind forecast data is received at altitudinal increments, as described in Figure 16. The `Weather` class is designed to store this information, which it does by storing an



instance of the `Wind` class for each increment. The `Wind` class stores the wind speed and heading information for a given increment.

15 m/s 300°	Increment 3
11 m/s 280°	Increment 2
7 m/s 284°	Increment 1
Ground Level	

**Figure 16. Example Wind Forecast Data**

The final class in the in the data section is the `Queue` class. `Queue` objects are used to keep a traditional queue of objects. Once the queue is full, the objects in it are replaced in a first in first out order.

#### 4.2.2 Display Section

The display section of the application drives what the user sees. It is responsible for computing what and where items appear to the user. The section contains eight main classes in two separate branches. The objects are organized into branches in keeping with the Java3D API, a hierarchy of Java classes which serve as the interface to a three-dimensional graphics rendering system [J3D99]. Java3D programs create instances of Java3D objects and must place them into a scene graph data structure. The scene graph is a tree structure that completely specifies the content of a *virtual universe*, and how it is to be rendered [J3D99].

#### 4.2.2.1 Java3D

*Virtual universe* is the Java3D term for the class that encompasses all the three-dimensional objects and information about how they are illuminated and viewed. It is the top of the scene graph tree, which has two main branches, or branch groups - a scene branch graph, and a view branch graph. The scene branch graph defines the Java3D objects, including their geometry, color, texture, and position of shapes, and the qualities and positions of lights. The view branch graph, however, describes the characteristics of the viewer of the contents in the universe. The description of the contents takes the form of several classes. A basic view of the scene graph is presented in Figure 16; the view branch graph is highlighted.

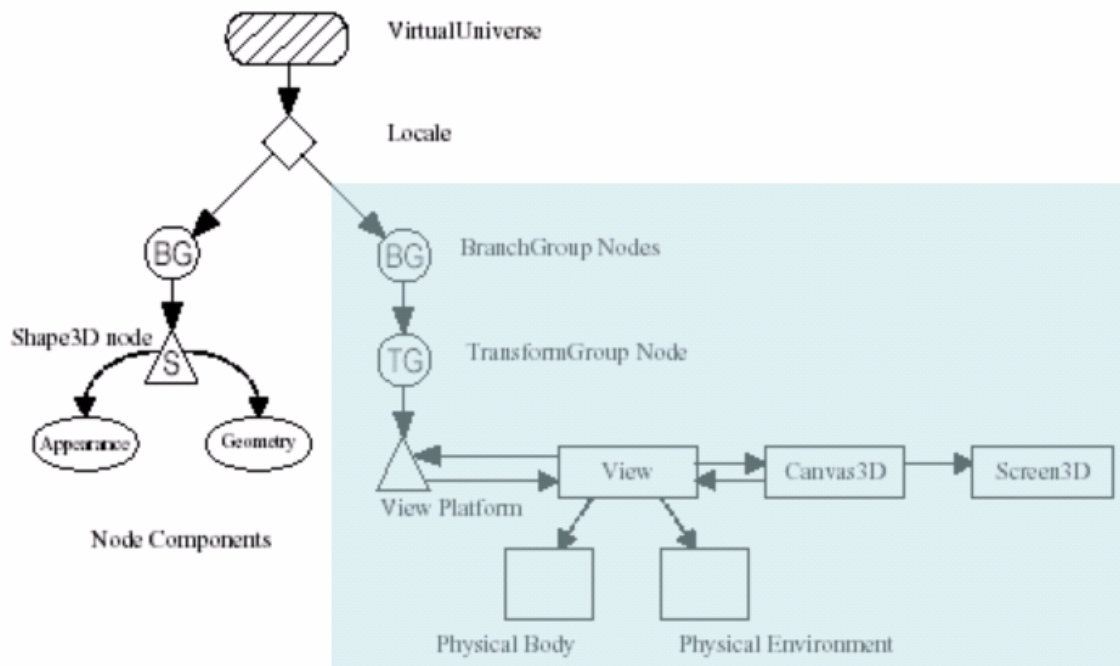


Figure 17. Basic Java3D Scene Graph [J3D99]

`TransformGroup` objects and `Transform3D` objects are also included in Java3D. A `TransformGroup` links all the Java3D objects within it. This is useful when animating, or otherwise moving a three-dimensional scene. Take for example a three-dimensional person. If a programmer wished to link a hand to an arm, she would put the hand and arm in the same transform group. `TransformGroup` objects can be placed within other `TransformGroup` objects. This can be used to further link Java3D objects. Perhaps the programmer would like to animate their three-dimensional person. If she wishes to move the person's arms without moving the torso, but wants the arms to remain attached to the torso if the person begins to walk, she may put the entire body in a `TransformGroup` with the torso and arms in separate transform groups within the body `TransformGroup`.

`Transform3D` objects are used to move entire transforms, including all their Java3D objects and sub-transform groups. `Transform3D` objects can move transform groups to defined positions, rotate them about the X,Y, or Z axis, or even animate them. `TransformGroup` objects and `Transform3D` objects are used extensively in this application.

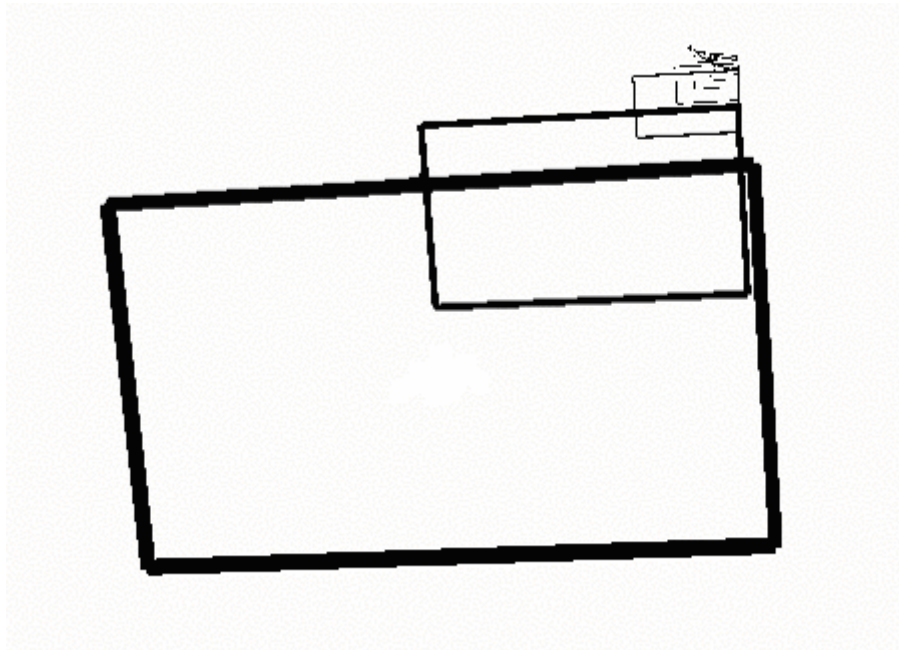
#### **4.2.2.2 Display Section Classes**

The seven main classes of the display section are: `Display`, `MyScene`, `Tunnel`, `Edge`, `PathInfo`, `MyView`, and `PositionBehavior`. They are organized into two branches, linked by the `Display` class. The branch beginning with `MyScene` includes `Tunnel`, `Edge`, and `PathInfo`. The branch beginning with `MyView` includes `PositionBehavior`.

An instance of the `Display` class is simply used to act as an interface to the display branch. It is not designed to have purpose aside from increasing modularity during engineering.

The scene branch, which begins with the `MyScene` class, creates all the Java3D objects within the three-dimensional space displayed to the paratrooper when running the application. An instance of the `Tunnel` class creates the tunnel which the paratrooper will travel through. A `Tunnel` object computes the shape of the tunnel by computing a path that will take the paratrooper from the release point to the target.

Once the path is determined, instances of the `Edge` class are placed along the path to form a tunnel. Instances of the `Edge` class are transform groups which contain Java3D objects representing the outer limits of the tunnel (see Figure 18).



**Figure 18. Tunnel Composed of Edge Objects**

The View Branch, which begins with the `MyView` class, determines what is displayed to the user. It does this by establishing how the three-dimensional space (filled by the scene branch) is viewed, and adding Java3D objects that are in constant view. The branch also includes inner classes of `MyView`, `Text` and `Indicator`, and the `PositionBehavior` class.

An instance of `MyView` creates inner classes `Text` and `Indicator` to aid the paratrooper in navigation. An instance of the `Text` class includes text information to notify the user of his current position information and desired heading, as well as target information. An instance of the `Indicator` class contains a Java3D object to represent the user's current velocity. The object, a circle, is at a constant distance of two meters from the user to reflect the direction (not magnitude) of the current velocity vector to help him navigate.

The final class, `PositionBehavior`, is used to move the user through the three-dimensional space. It is based on the position and velocity of the paratrooper in the physical world. When the application receives a notification of the paratrooper's position in the physical world the display is updated to reflect that in the three-dimensional space of the navigation reference.

#### **4.2.3 Information Gathering Section**

The information gathering section of the application consists of only two classes which are designed primarily to obtain information. The two classes in the information gathering section are, `SerialReader` and `GetInfoFrame`.

An instance of the `SerialReader` class is designed to read from the serial port. For a paratrooper's navigation reference to work it must have position data. This data is obtained via a GPS receiver, which is attached to a serial port. `SerialReader` is designed to continuously read from the serial port gathering data. It then parses the data it receives and uses the resultant information to update the user's position.

`GetInfoFrame`, on the other hand, is designed to gather information from the user only once. It gathers the paratrooper's intended release and target point's latitude, longitude, and altitude information. It also gathers the name and location of the files representing parachute and weather data. The names of these files are then used by instances of the `Parachute` and `Weather` classes, respectively.

### **4.3 Software Implementation**

The classes described above were implemented in the Java programming language. This section describes the implementation of these classes and outlines their interaction. The interaction is outlined by tracing through the system during operation. Since this program is object-oriented, and multi-threaded, there is concrete flow-of-control, but a rough outline is given to provide insight into operations.

At run time the system contains several objects. The classes these object belong to and the multiplicity of the instances of those classes are as follows:

- `Position` – many
- `SmoothPosition` – one
- `MPosition` – many
- `Parachute` – one
- `Weather` – one
- `Wind` – many
- `Display` – one

- MyView – one
- Tunnel – one
- Edge – many
- PathInfo – many
- MyView – one
- PositionBehavior – one
- SerialReader – one, and
- GetInfoFrame – one

#### **4.3.1 Implementation of Classes**

The description of the implementation of the classes in this program does not cover every class. Some classes, such as the classes in the data section, are so simple their implementation does not warrant description. They will, however, be included in the outline of the system interaction. The classes that are described in this section are: MyScene, Tunnel, Edge, MyView, PositionBehavior, SerialReader, GetInfoFrame, SmoothPosition, Queue, and MPosition. The relationship between these classes is illustrated in the simplified class diagram of the system in Figure 19.

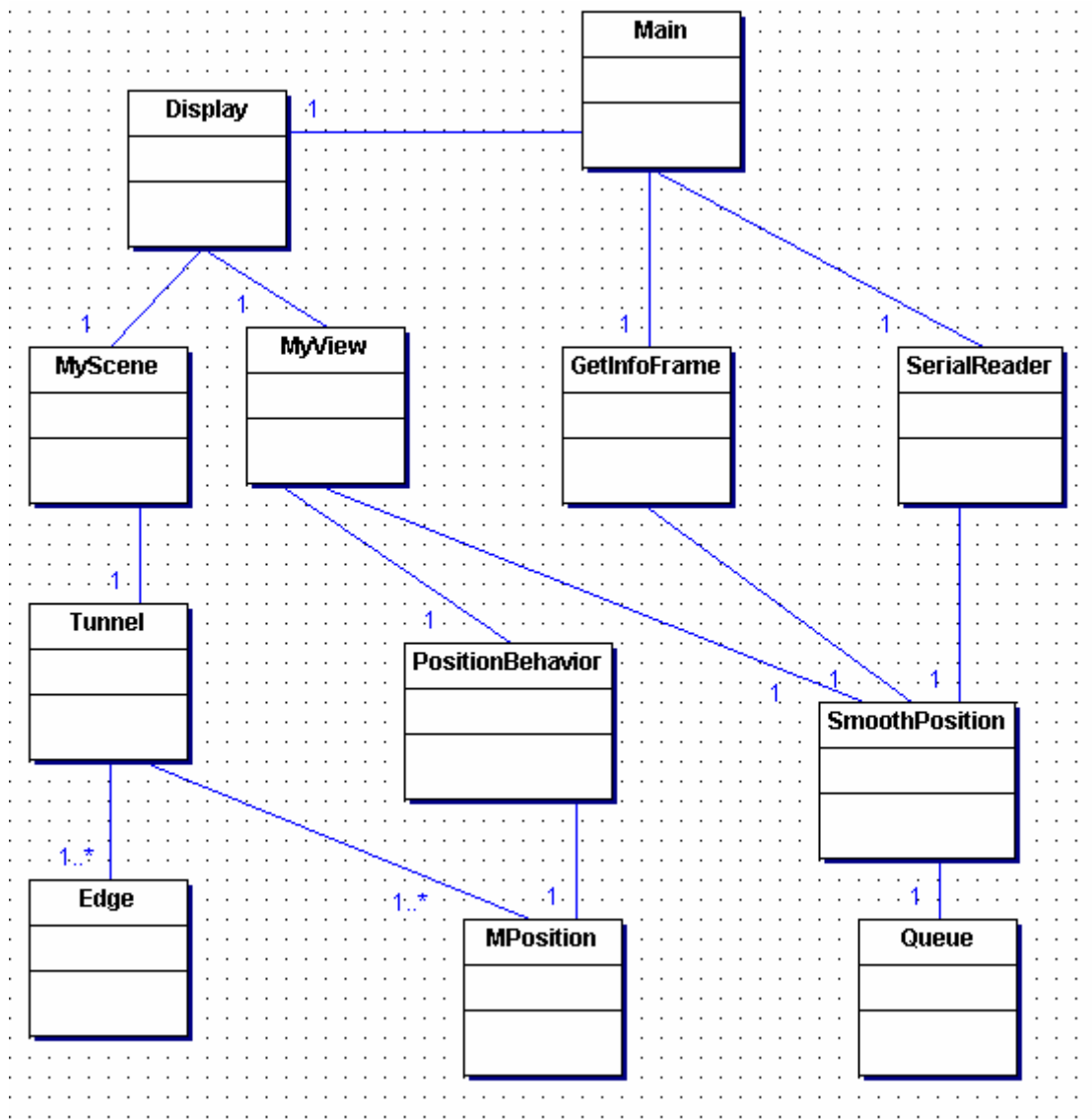


Figure 19. Simplified Class Diagram of Reference System

#### 4.3.1.1 MyScene

The `MyScene` class is an extension of Java3D's `BranchGroup` class. The main job of instances of the class is to create an instance of the `Tunnel` class. `MyScene` also

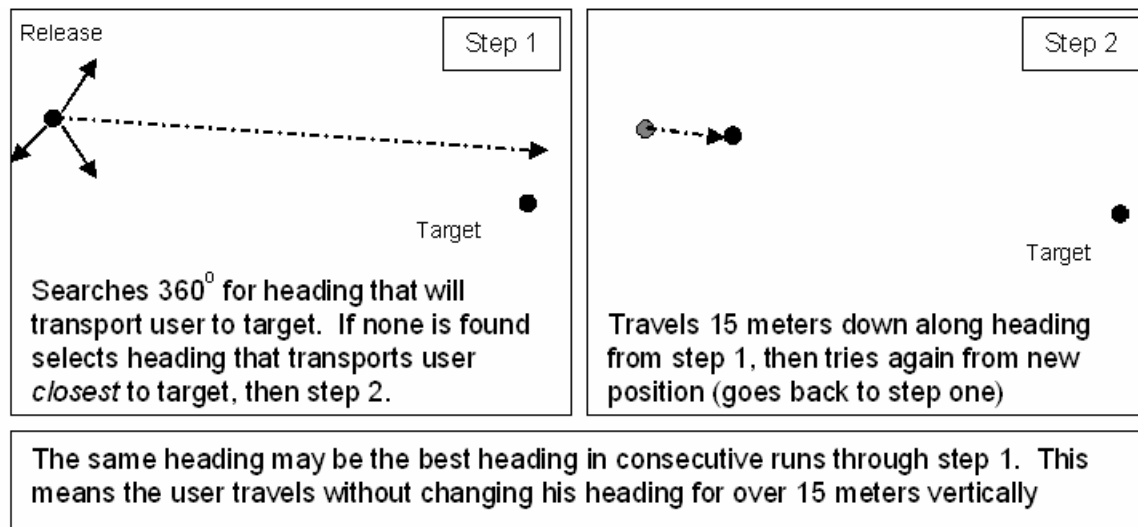


sets the application window's background color to black, and creates a white three-dimensional box at the origin. This box represents the target.

#### **4.3.1.2 Tunnel**

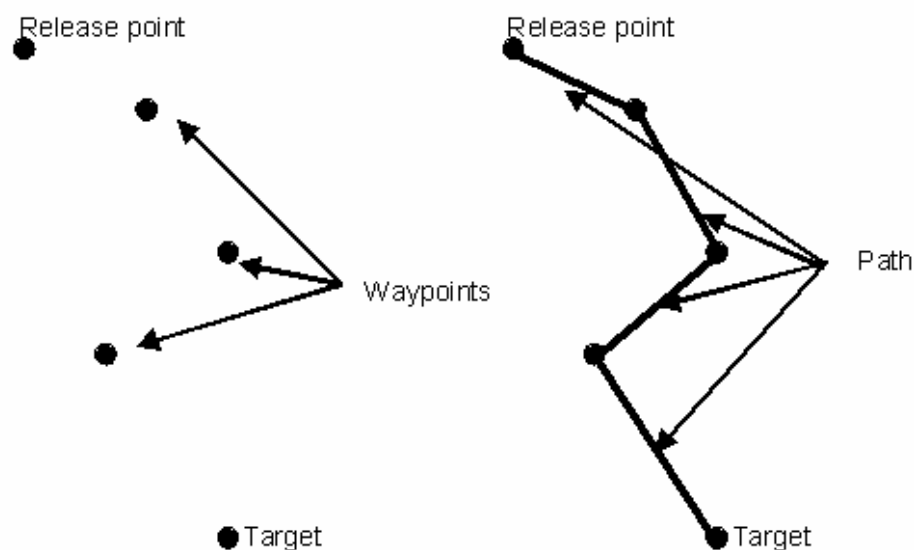
The `Tunnel` class is the heart of the three-dimensional space that makes up the navigation reference. The `Tunnel` object uses the parachute and weather information to derive the best path for the paratrooper to take from the release point to the target. This is accomplished through the implementation of a simple greedy search algorithm.

The algorithm to compute the path contains a loop. Every iteration of the loop iterates through all possible headings in attempt to find the constant heading that will take the user from a given position  $x$  to within a certain distance from the target. If no heading brings the user to within the given distance the algorithm moves  $x$  down a given distance at the heading that brought the user closest to the target. This continues until either the target altitude is reached, or  $x$  has been moved to within a predetermined vertical distance from the target (see Figure 20).



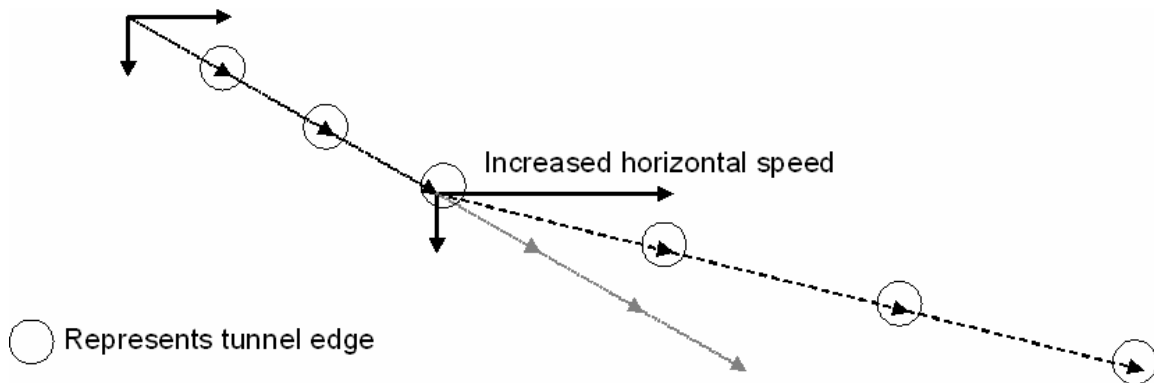
**Figure 20. Greedy Algorithm Used to Determine the Best Path**

Every heading change, along with the location of the change is saved. These changes in heading are considered waypoints, and define the path (see Figure 21). The information is saved as `PathInfo` objects and stored in a list.



**Figure 21. Waypoints Define Path Shape**

Once the path of the tunnel is made the actual tunnel is built around it. This is done by placing `Edge` objects centered on the path. The `Edge` objects are centered on the path and perpendicular to it. `Edge` objects are placed at a uniform vertical distance of four meters from each other on straight stretches of the tunnel. An `Edge` is placed along the path one meter down from every turn, however, making them closer where the user must turn. Although the `Edge` objects are placed at a uniform vertical distance from each other (with the exception of turns), they are not necessarily placed at a uniform horizontal distance. This is because changes in wind speed or changes in user heading relative to wind heading change the ratio of horizontal speed to vertical speed; increases in horizontal speed result in increases of distance between `Edge` objects. This is described in Figure 22. Arrow objects are also placed on the path at turning points. Arrows are used to convey to the user the next heading he must take.



**Figure 22. Effect of Speed on Separation of Tunnel Edges**

#### 4.3.1.3 Edge

The `Edge` class extends Java3D's `TransformGroup` class, and is itself a conglomeration of other `TransformGroup` objects. Within each of the inner transform groups, four in all, is a line. The lines are moved together to form a rectangle (see Figure 23) which is used to determine the outer bound of the path. The `Edge` object is moved to a specified point on the path and rotated by specified angles in order to appear perpendicular to the path.

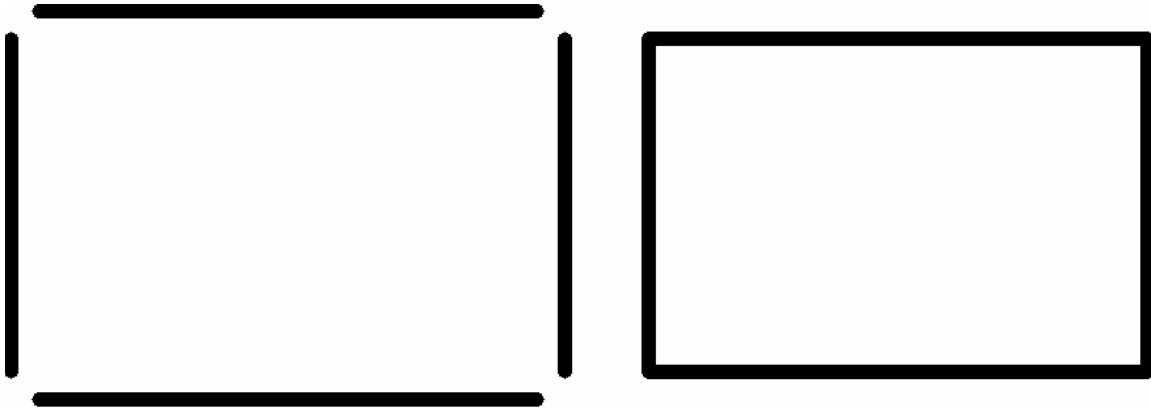


Figure 23. Four Objects (left) Used to Create One Rectangle (right)

#### 4.3.1.4 MyView

The `MyView` class, which extends `BranchGroup`, is a class necessary for the implementation of the reference in Java3D. By default the `BranchGroup` class that defines the view branch defines the position of the viewer, as well as how the viewer views the screen (e.g. Java3D facilitates room and head-mounted displays). This implementation of a view `BranchGroup` adds a predictor symbol and text to the viewer.

Since the indicator symbol and text are added to the viewer and not to the three-dimensional scene, they are constantly visible to the user. The `MyView` implementation of a view `BranchGroup` also adds an implementation of a `Behavior` class. `Behavior` is an abstract Java3D class, implementations of which move the viewer through the three-dimensional space defined by the scene branch.

The information added to the view serves to aid the user in navigating. The indicator symbol is a circle that indicates to the user his current velocity to help him determine the direction of his travel more precisely. The text added to the screen, however, is to give context to the three-dimensional tunnel's qualitative data (see Figure 24). A concrete instance of the `Behavior` class, `PositionBehavior`, has been added to the viewer to move the user through the three-dimensional space.

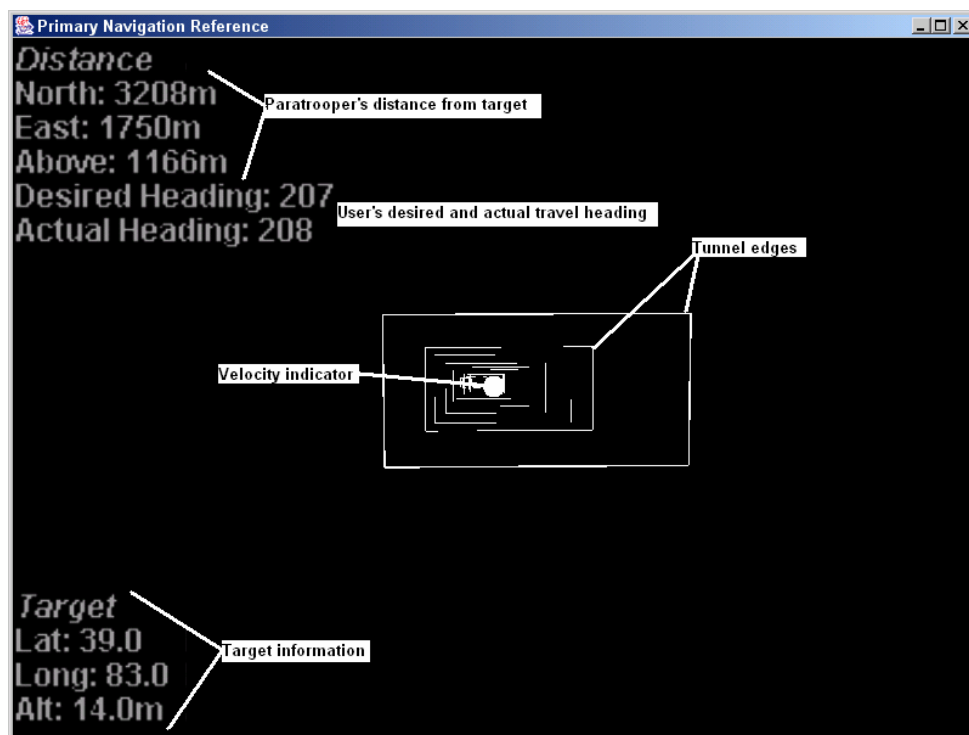


Figure 24. Navigation Reference with Textual and Graphical Information

The `PositionBehavior` class has been implemented to deal with the unique situation of parachute travel. An implementation of `PositionBehavior` catches events fired by the `SmoothPosition` object representing the paratrooper when its position is updated (discussed later). It reads values of the `SmoothPosition` object and updates the viewer's position accordingly. By computing the difference between the former and current positions the `PositionBehavior` object is able to determine the angles (heading and slope) of the user, and adjusts the viewer to not only take the new position, but to face the direction of the user's travel.

`PositionBehavior` computes the distance between the former and current positions by converting their geodetic coordinates (latitude, longitude, and altitude) into distance information, then working with the distances. In this implementation the process of converting the geodetic coordinates to distances uses the user and target information. That way it is known not only how far the user has traveled, but his position relative to the target. Every time the user's position is changed his distance from the target (in meters) is computed, and then compared to his previous distance. The algorithm used to convert the geodetic information to distance information is shown in Table 1.

**Table 1. Conversion of Geodetic Coordinates to Distance**

User position's geodetic coordinates	$p_u = [\lambda_u \phi_u \gamma_u]$
Target position's geodetic coordinates	$p_t = [\lambda_t \phi_t \gamma_t]$
Earth's semi-major axis (in meters)	$a = 6378135.0$
Earth's eccentricity	$e = 0.0818191908426$
East/West difference	$\Delta\lambda = \lambda_u - \lambda_t$
North/South difference	$\Delta\phi = \phi_u - \phi_t$
Altitude difference	$\Delta\gamma = \gamma_u - \gamma_t$
$R_p$ (temporary variable)	$\frac{a}{\sqrt{(1 - e^2 \sin^2 \lambda_u)}}$
$R_m$ (temporary variable)	$\frac{a(1 - e^2)}{(1 - e \sin^2 \lambda_u)^{3/2}}$
User's distance east of target	$\frac{(R_p + \gamma_u)}{\cos \phi_u} \Delta\lambda$
User's distance north of target	$(R_m + \gamma_u) \Delta\phi$
User's distance up from target	$\Delta\gamma$

#### 4.3.1.5 SerialReader

The `SerialReader` class implements Java interfaces `Runnable`, and `SerialPortEventListener`. By implementing the interface `Runnable`, instances of `SerialReader` are allowed to spawn new threads; by implementing `SerialPortEventListener`, instances are allowed to retrieve data sent to the machine's serial port.

This class is customized to read data in GPS format. It reads from the serial port every time something is sent to the serial port. GPS receivers send messages in packets instead of sending entire messages at once. Instances of `SerialReader` listen to the serial port and concatenate the packets together until a whole message has been received.

Once a whole message is identified it is parsed and the latitude, longitude, and altitude data extracted. This information is used to update the user's position.

#### **4.3.1.6 GetInfoFrame**

An instance of the `GetInfoFrame` class produces a frame intended to gather information from the user. This is done at the invocation of the application, before the display has been created. The frame (pictured in Figure 25) requests the user enter the latitude, longitude and altitude values of his desired target and his projected release point. It also requests that the user browse to find files holding the information about his parachute canopy and current wind forecast.

In order to limit error the user is required to lock the values of all of the fields before submitting the values. Although not flawless, forcing the user to lock the fields requires him to think, albeit briefly, about each field. The fields, which cannot be modified when locked, can be unlocked and modified if the user decides he needs to change the value of a given field. To further avoid potential mistakes, the fields cannot be locked unless the field has been given a value.

Once the user presses the *submit* button (if all fields have been locked) the information is saved and the frame is closed.



Enter information, then set (lock) each group when finished. All fields must be set in order to submit

Target:

Latitude  ☒ North ☐ South

Longitude  ☐ East ☒ West

Altitude

Release Point:

Latitude  ☒ North ☐ South

Longitude  ☐ East ☒ West

Altitude

Weather:

Parachute:

Figure 25. Screen Shot of GetInfoFrame Window

#### 4.3.1.7 SmoothPosition

The `SmoothPosition` class is an extension of the `Position` class. `Position` objects simply store information (latitude, longitude, and altitude), while `SmoothPosition` objects store several positions and use them to compute a smooth path. This is done by maintaining a `Queue` object (briefly described below) of past positions, and using them to produce a smooth approximate path. The smoothing process, designed to reduce the jumpiness that can result from the inaccuracy inherent in

GPS, is based on average velocity. As such, it assumes that within a relatively small sample velocity remains fairly constant.

The smoothing algorithm works by computing the average velocity of the user given three-dimensional position samples in a moving window. Once the average velocity in each dimension is calculated, the position second deep in the window (which will be the leading after the next update) is updated with an approximation – the average velocity in all dimensions. The current position is also approximated. Its approximation uses the approximate value of the first position in the window, and adds to that the average velocity in all dimensions multiplied by the window size. Meanwhile, the original sample value of the current position remains in the window. The algorithm is explained in mathematical notation and pseudo-Java code in Table 2, while an example result is pictured in Figure 26.

**Table 2. Smoothing Algorithm**

Basic algorithm	$\bar{v} = \left( \frac{1}{k-1} \right) \sum_{i=0}^k \Delta v_i = \frac{x_{i+1} - x_i}{\Delta t}$ <p> <math>\bar{v}</math> - average velocity  <math>k</math> - window size  <math>x_i</math> - position <math>i</math> in the window  <math>t</math> - time </p>
Pseudo-Java	<pre>//sum the distances in three dimensions for (int i = QUEUE_SIZE-1; i &lt; 0; i++) {     pos1 = queue.get(i);     pos2 = (Position)queue.get(i-1);     sumLat = sumLat + (pos1.getLat()-pos2.getLat());     sumLon = sumLon + (pos1.getLon()-pos2.getLon());     sumAlt = sumAlt + (pos1.getAlt()-pos2.getAlt()); }</pre>

```

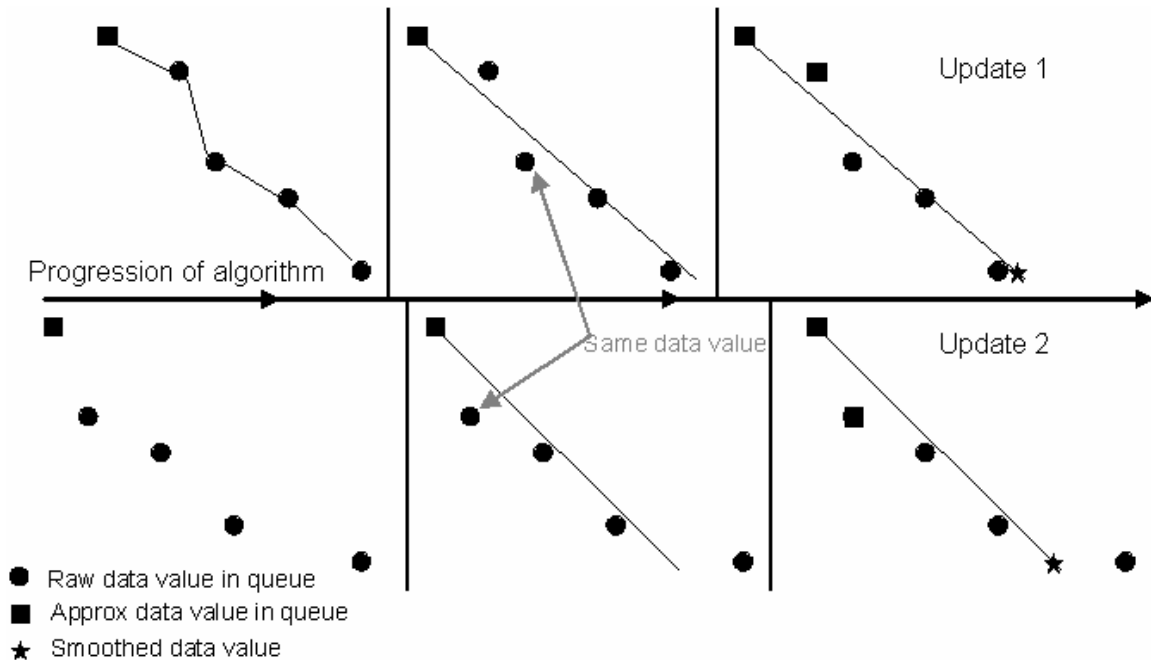
//use the distances and window size to get a velocity
avgLatVelo = sumLat/(QUEUE_SIZE-1);
avgLonVelo = sumLon/(QUEUE_SIZE-1);
avgAltVelo = sumAlt/(QUEUE_SIZE-1);

//get the (approximate) value at the front of the window
firstInQueue = queue.get(0);

//set smoothed (approx) position values for this sample
//firstInQueue + velocity in each direction
setLat(firstInQueue.getLat()+((QUEUE_SIZE-1)*avgLatVelo));
setLon(firstInQueue.getLon()+((QUEUE_SIZE-1)*avgLonVelo));
setAlt(firstInQueue.getAlt()+((QUEUE_SIZE-1)*avgAltVelo));

//set the second position in window to an approximation
// for next time around
secondInQueue = queue.get(1);
secondInQueue.update((firstInQueue.getLat()+avgLatVelo),
                    (firstInQueue.getLon()+avgLonVelo),
                    (firstInQueue.getAlt()+avgAltVelo));

```



**Figure 26. Visual Demonstration of Algorithm Results**

#### **4.3.1.8 Queue**

The `Queue` class is a simple extension of Java's `Vector` class. This class however, only allows itself a predetermined size. Once that size is reached every time an object is added to the end the first the object that has been in the queue the longest is eliminated from the data structure. This technique allows for a moving window that suits this application well, because it only keeps track of the last few reported position samples.

#### **4.3.1.9 MPosition**

The final class implementation covered is that of the `MPosition` class. `MPosition` stores three-dimensional position data. The `MPosition` class can be instantiated by passing it either three-dimensional position data in meters, or by passing two `Position` objects. If `Position` objects are passed, `MPosition` converts the geodetic information in `Position` objects to coordinates in meters through the same process as the one described in the `PositionBehavior` class.

The `MPosition` class also has the ability to update position values by shifting them over in a certain direction and down. This ability is exploited during determination of the user's path. The `Tunnel` object passes horizontal and vertical distance and direction of travel then the `MPosition` class computes and sets the new position.

#### **4.3.2 Interaction of Classes**

A rough sketch of the system is provided here by following the basic flow of control of the system. Although the system is multi-threaded, and does not have a set

flow of control, the system does perform several steps in a serial manner. That basic manner begins with program invocation and ends with the user reaching the ground, at which point it has performed its full function.

At the onset of the program a `Position` object called *target*, and a `SmoothPosition` object called *user* are created. These objects represent the user's target and the user, respectively. Secondly, the application checks the system for a serial port. If there is a serial port, a `SerialReader` object is created to read data from the port, parse the data and update the *user* position.

If the serial port can be found, a `GetInfoFrame` object is created. This object creates a window on the screen that gathers information about user's target, release point, and parachute and current wind conditions. Once the user submits the information it is saved and the window is closed.

Next, a display object is created. The display object creates an instance of the `MyScene` classes described above. `MyScene` creates an instance of the `Tunnel` class which uses parachute and weather data (`Parachute` and `Weather` objects) to create a path, and then a tunnel from the user's release point to his target.

During the development of a tunnel `MPosition` objects are used heavily. When computing the heading(s) the user must take to the target, `MPosition` objects are used. For example, the user's release position is set as an `MPosition` object and moved along the proposed path. `MPosition` objects are positions in terms of meters from the target, and are more efficiently manipulated since a parachute's forward speed and wind speed are measured in terms of meters. `MPosition` objects are also used to determine the

position of the `Edges` that define the tunnel the paratrooper travels through and the `Arrows` that indicate heading changes to him. Once the tunnel is built the scene does not change, because the tunnel is static.

After the `MyScene` object is created, a `MyView` object, *myView*, is created to determine how the user sees the scene that was created just prior. *MyView* creates the text and indicator objects that are constantly in the user's view. It also creates a `PositionBehavior` object, *positionBehavior*, which handles events fired by `SmoothPosition` objects.

Every time a `SmoothPosition` object is updated it fires a JavaAWT event. These events are handled by *positionBehavior*. *PositionBehavior* re-computes the user's position and angle of the viewer every time it receives an event of this type. This process continues until the user turns the application off.

#### **4.4 Summary**

This chapter discussed in detail the design and implementation of the paratrooper's navigation reference. The structure of the system and essential purpose of each class was covered. Furthermore, the implementation of the more complex, or significant, classes was covered, and a short walk-through of the program was given. Algorithms unique or essential to this application were also described and given in mathematical notation.

## ***5. Evaluation and Results***

### **5.1 Introduction**

This chapter describes the techniques used to evaluate the reference application developed in this research. The reference was evaluated on its accomplishment of this research's three main objectives, development of a three-dimensional navigation display, convincing representation of the user as traveling through that three-dimensional space, and production of a display with the ability to serve as a heads-up display with NVGs. It also describes a modified reference application used to interactively simulate a paratrooper's HAHO jump, and record pertinent information.

A primary navigation reference that is not sufficiently effective can not lead the user to the correct target. A reference's effectiveness is reliant on its ability to calculate the correct path for a paratrooper, as well as its readability, and usability. It is possible for the reference to provide the most accurate information possible, but if it is not conveyed to the user it is ineffective. The development of an ineffective system is not satisfactory, only a system that is effective (correct, usable, and readable) is acceptable. This section describes the process through which the display was tested and the metrics against which it was tested.

### **5.2 Evaluation Process**

The reference developed for this thesis was developed for paratroopers with the cooperation of paratroopers. A group of paratroopers, the potential users, gave their requirements for an effective display. This thesis solicited requirements from users in

order to make the resultant application operationally effective. The system requirements were informally gathered from paratroopers. A point of contact within the paratrooper community was asked what were considered the elements essential to navigation that must be included in a primary navigation reference. He gathered several other paratroopers, and together they developed the following list of requirements: altitude (of user), distance to target, bearing, and waypoints [SMI02].

After the development of a functional application it was presented to testers to determine its effectiveness. The users, tested in simulated conditions, were for the most part inexperienced with parachuting. As a group of users with little, if any, experience with parachute travel, the usefulness of their comments on technical aspects of the reference was limited. Their comments on the readability of the display and their ability to navigate using the reference were, however, extremely valuable. Modifications were made to the system as a consequence of the test results.

### **5.2.1 Modified Evaluation System**

After a working navigation reference display was developed for this thesis, it was tested with a simulation in order to verify display concepts and performance without endangering users. This required modifying the application to receive simulated input instead of live GPS data. In order to do this the part, the system that gathered GPS data was removed, as was the portion of the system used to convert the user's physical position to a position in the three-dimensional space of the graphical display. These portions of the system were replaced by another portion that used weather and parachute



information, as well as interactive input from the user, to determine the velocity and position of the user in the three-dimensional space of the graphical display.

The modified system worked as a simulation and differed, on a high level, from the GPS-based system in that it actively determined the position of the user. The GPS-based implementation of the system passively gathered position information from the physical world, and then used that information in the display. The simulation version of the system actively determined the position of the user and displayed that position.

In order to determine the user's position, the simulation used the same parachute and weather data used to determine the shape of the reference's navigation tunnel. This information determined the speed and direction that the wind pushed the user, and the forward airspeed of the user. The simulation allows the user to interactively determine his heading, and therefore, his ground speed.

### **5.3 Evaluation**

The reference developed for this thesis was evaluated with a test that involved multiple subjects using the reference to navigate using the simulation system described above. The testing technique and its results are discussed below.

#### **5.3.1 Simulation Testing**

Evaluation using the simulation system described above was conducted over the course of one week. During that week thirty-three subjects were asked to navigate through two simulated HAHO jumps using the navigation reference. The directions given to the subjects were minimal. Each individual was briefed on their objective and

how to steer in the simulation. Explanation of the individual elements of the reference was purposely vague, however.

Subjects were allowed to steer left and right only using a keypad. Their speed and glide slope was based on real world parachute performance. However, unlike a paratrooper in the physical world who can adjust his speed by applying or releasing his brakes, subjects were only allowed to steer left and right. This approach tested the subject's ability to stay on track while traveling at a constant forward air speed, which is more limited than parachute flight.

During the test the subject's position in the display's three-dimensional space was recorded twenty times each second. Every position was compared to their desired position at that height and the subject's variance recorded. The desired or optimal path was previously known. It was the path around which the tunnel edges were placed and defines the desired positions. Of the three dimensions, height is the only one that the subject's position and the desired path must have in common – for every recorded subject's position height, the desired path also has a position at that height. So, the subject's variance in the other two dimensions was recorded. The data gathered in this test determined how well the subjects could stay on course, and whether or not the variance compounded over time.

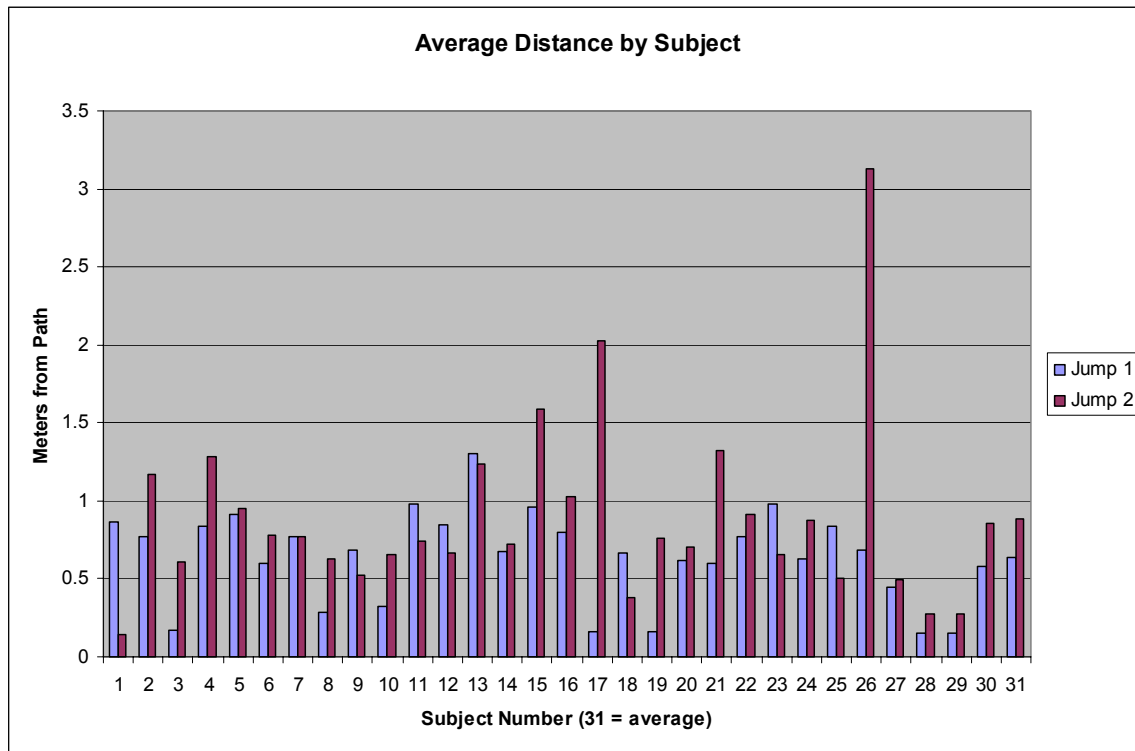
Subjects performed two simulated jumps each. Both of the jumps originated from the same release point 1386 meters above, 3751 meters north, and 2066 meters east of the target. The parachute characteristics were the same in each simulated jump, only the weather forecast differed. The difference in the wind directions and speed greatly altered

the shape of the tunnel. In the first jump the tunnel contained one turn, or change in user heading, while the tunnel in the second jump contained twenty-seven turns. Both jumps resulted in the recording of 2800 data points.

#### **5.3.1.1 Results**

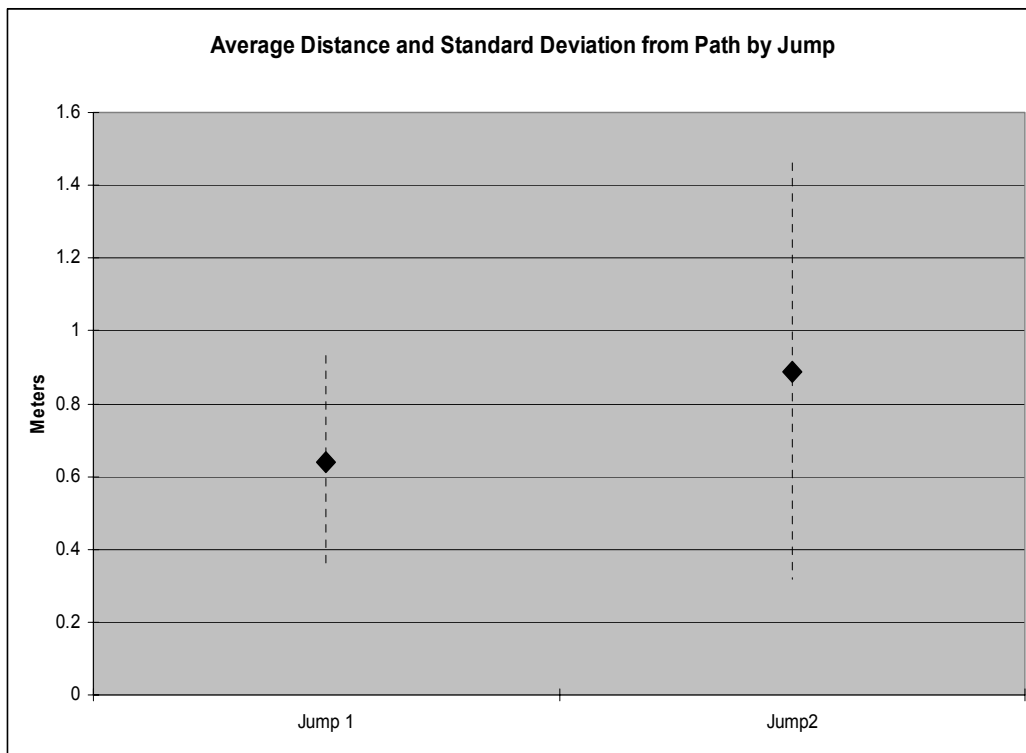
The results of this test prove promising for an operational version of the paratrooper's primary navigation reference. Subjects' distances from the desired path during sixty simulated jumps (each subject performed two simulated jumps; the results from three subjects were removed due to input errors) averaged less than one meter, showing that on average the subjects stayed well within the four meter (horizontal) by two meter (vertical) tunnel. Furthermore, results show that user error does not necessarily compound over time.

On average subjects stayed within the designated tunnel. During both tests subjects averaged 0.764578 meters from the path, with a standard deviation of 0.479813 meters. The average distance from the path was lower for the first jump, 0.640766m, with a standard deviation of 0.290495m, than the second, 0.888389m, standard deviation of 0.571317m. Figure 27 below shows each subject's average distance from the desired path. In this Figure the subjects' average is shown as subject 31.



**Figure 27. Average Distance by User. Only Two Users Exceeded Two Meters**

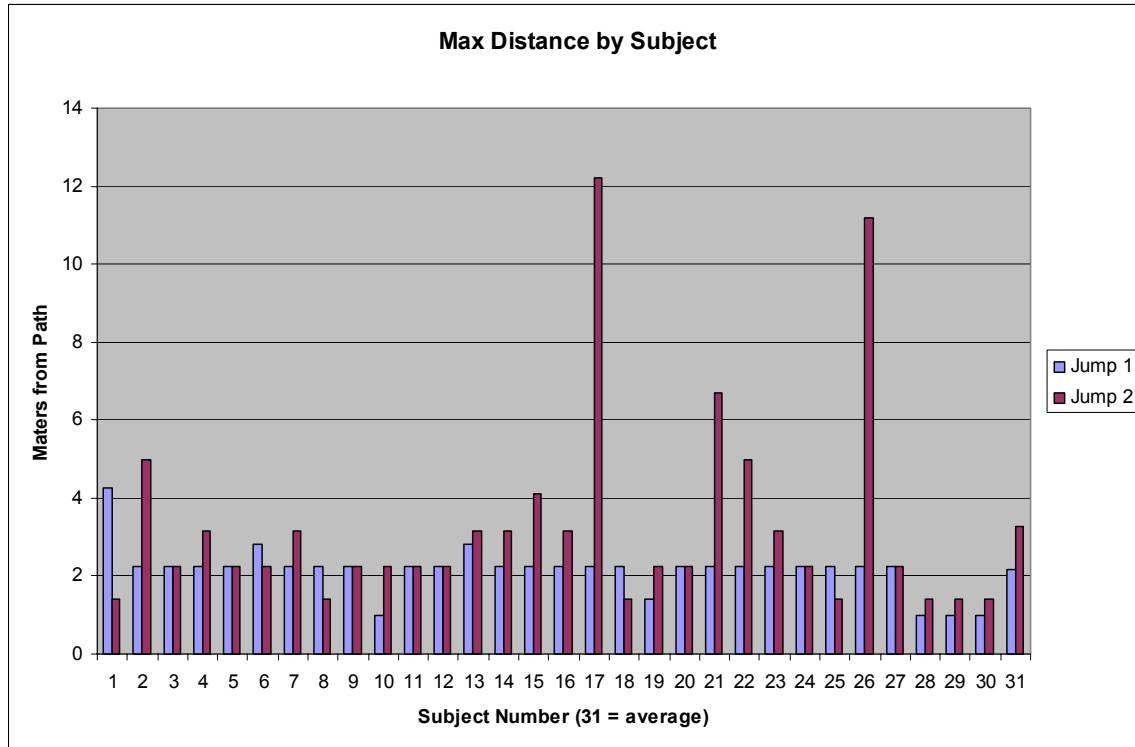
While it was expected that subjects would stay on path much better in the first jump than the second because of the relatively small number of turns (one turn for every twenty-seven in the second), there is no statistical difference between the two. This is because the standard deviations of the two overlap (see Figure 28). The fact that the averages are not statistically different, despite the disparate number of turns, speaks to the success of the tunnel-in-the-sky navigation technique.



**Figure 28. Average Distance and Standard Deviation of Subjects by Jump**

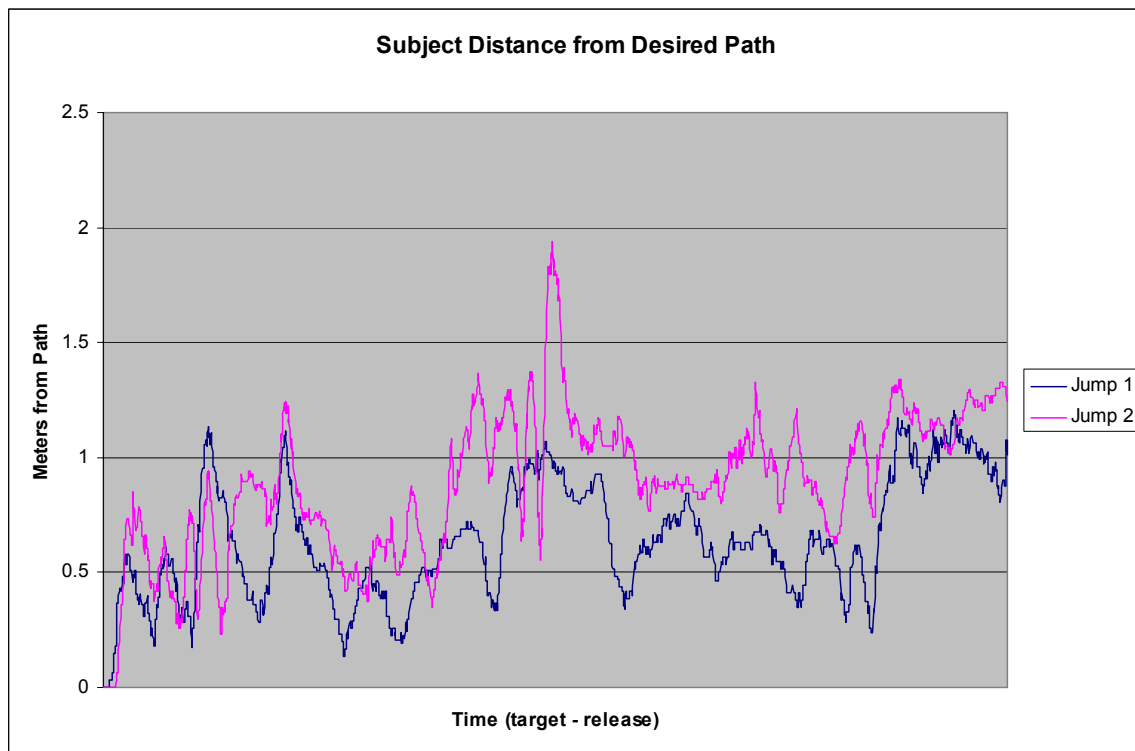
Results also indicate that subjects stayed relatively close to the desired path throughout the simulated jumps. Figure 29 presents subjects' maximum distances from the desired path. The figure shows that, for the most part, most of the values do not greatly exceed two meters. The subjects' average (jump 1: 2.15024m, standard deviation 0.613569, jump 2: 3.25627m, standard deviation 2.561907), depicted as subject 31, shows that on average their maximum distance from the path did not significantly exceed two meters, the horizontal distance of the tunnel edge from the path. Again, this demonstrates the value of the visual boundaries given by the tunnel. Subjects noticed

they were near the edge, or outside of the tunnel and steered themselves back toward the center, or the desired path.



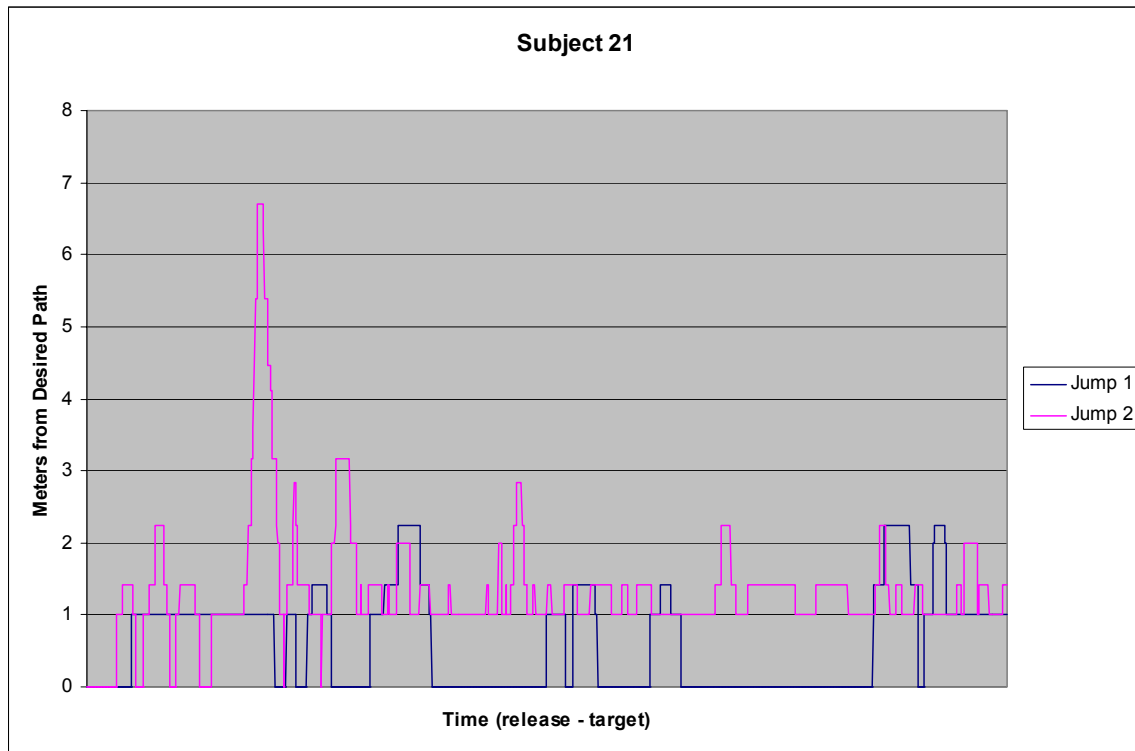
**Figure 29. Maximum Distances from Desired Path, by Subject**

The test also showed that user error does not necessarily compound over time. Subjects were able to regain the desired path after straying from it, despite only having the ability to control horizontal heading. Figure 30 shows the average path of the subjects in both jumps. The subjects were started on the desired path, which is why the distance begins at zero, and although on average they never return to the path (they were within the tunnel the entire way), the figure shows that in places distance from the path is reduced. This is more clearly shown in the results of the individuals.



**Figure 30. Average Distance of Subjects from Desired Path**

Figure 31 shows subject twenty-one's difference from the desired path. It should be obvious from Figure 29 that he strayed fairly far from the desired path in jump two, in fact only two subjects strayed farther in all of the testing. What Figure 29 does not show that Figure 31 does, is that he (subject twenty-one) left the path early on, yet returned to travel almost exclusively within the tunnel for the remainder of the simulated jump.

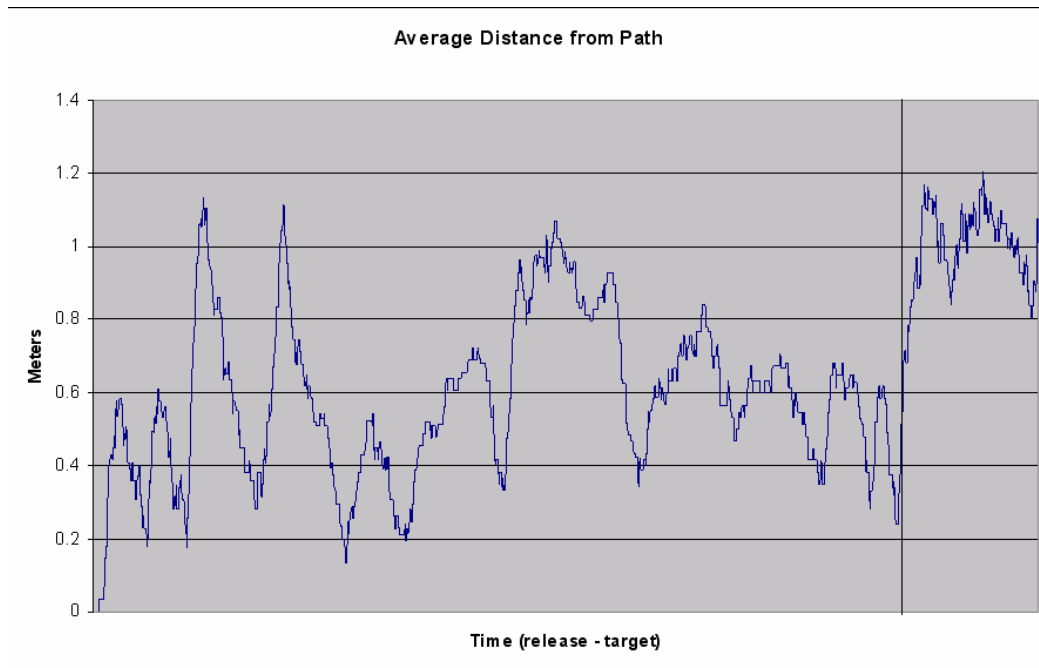


**Figure 31. Subject 21's Difference from the Desired Path**

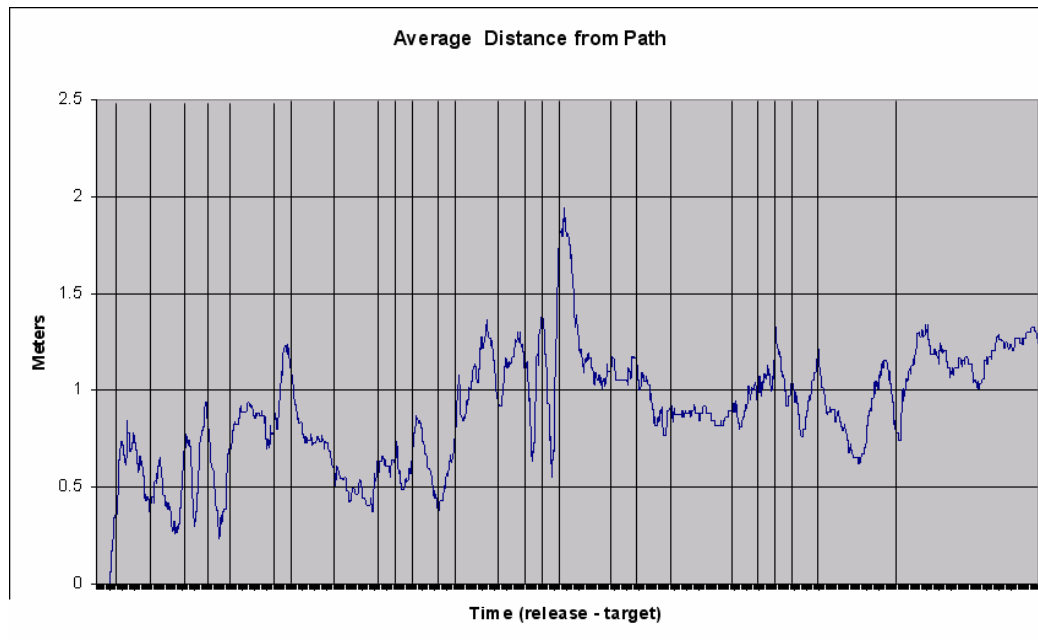
Although users can reduce their distance from the desired path after straying from it, this reduction has a limit. The *point of no return* is the distance from the desired path beyond which the user cannot regain the desired path. This *point of no return* is situation dependent; it is defined by the parachute canopy's glide slope, forward speed, and wind conditions, which vary for every jump. In fact, the *point of no return* varies within a jump, because wind conditions change at different altitudes. Therefore, statistical analysis attempting to determine a general *point of no return* is meaningless. What is meaningful, however, is the fact that users can move closer to the desired path, but that ability is limited by the *point of no return*.



Test results from these parachute jump simulations yield further information when the positions of turns on the path are taken into account. Subjects strayed from the desired path immediately after turns. In the space between turns they were able to attempt to regain the path. During the second simulated jump, the space and time between turns was extremely close in parts. In these parts the average subject distance from the desired path increases steadily. Figures 32 and 33 show subject averages and uses vertical lines to indicate where the user is required to turn, or change heading. There is only one turn in Figure 32, and twenty-seven in Figure 33.



**Figure 32. Distance from Path for Jump 1. Turn Indicated with Vertical Line**



**Figure 33. Distance from Path for Jump 2. Turns Indicated with Vertical Lines**

The results from this test show that users can easily follow a tunnel-in-the-sky to navigate a HAHO jump path. Overall, subject performance was very good. Subjects' were, on average, within the tunnel presented to them through the entirety of both simulated jumps. They were able to regain the desired path despite occasionally straying from the path. Furthermore, it should be noted that straying from the path, and even straying from the tunnel, did not preclude users from re-obtaining the desired path.

#### **5.3.1.2 Test Environment**

This test was performed on a Dell® Dimension 8200 desktop PC with a Pentium® 4 two gigahertz processor and one gigabyte of RAM. Subjects were solicited via email at the Air Force Institute of Technology, and came almost exclusively from the student

body. The subjects were briefed individually prior to testing. No personal information was recorded from subjects.

#### **5.3.1.3 Test Limitations**

The nature of this test was limited to a simple simulation, and as such was unable to gather any feedback on the reference's effectiveness in operation in the physical world. Furthermore, subjects, for the most part, had no experience with parachuting and, therefore, could not comment on the reference's effectiveness in operational conditions. This test was designed for preliminary evaluation of the reference before any operational testing would occur.

#### **5.3.2 Other Test Notes**

Feedback from subjects during this test was a catalyst for changes in display techniques. The major change suggested by subjects was the removal of an arrow indicating user heading changes. The arrows were placed within the three-dimensional scene at the center of the tunnel, oriented to indicate the user's new desired heading. Subjects frequently expressed that the arrows passed too fast, and they were unable to gather any information from the arrows.

Subjects expressed that they were unable to discern the heading indicated by the arrows when they were not at the center of the tunnel. Unfortunately, that is when a user would most need the information presented by an arrow indicating heading. This problem was a result of perspective. Although perspective can aid in navigation, as

evidenced by the helpfulness of the tunnel, that is not always the case. As a result of these problems, heading indication arrows were removed from the navigation reference.

#### **5.4 Summary**

After soliciting requirements for a primary navigation reference from paratroopers a functional reference application was developed. Non-operational evaluation of that reference was performed in two manners. The first manner was the testing of subject performance in two simulations of HAHO parachute jumps.

The reference application developed for this thesis was modified for the simulation testing. Subjects' were asked to navigate through the simulated three-dimensional space using a keyboard. User positions were recorded and compared to the optimal path determined by the system. This evaluation was designed to demonstrate a user's ability to travel within the tunnel and whether straying from the tunnel compounded error over time. Results from this evaluation prove that users are able to follow the three-dimensional tunnel and that navigation errors do not necessarily compound over time. These results prove favorable for the development of an operational version of the navigation reference.

## ***6. Conclusions and Recommendations***

### **6.1 Introduction**

The system developed for this thesis represents a novel application of three-dimensional graphics and GPS. Evaluations have shown the integration of these two systems has the potential to produce an effective navigation tool for paratroopers performing HAHO jumps.

### **6.2 Research Impact**

This research introduced a novel application combining three-dimensional graphics and GPS for a primary navigation reference for paratroopers. It uses three-dimensional graphics to realistically portray a paratrooper's movement in the physical world, measured by GPS, as movement in a computer generated scene. Evaluation of a prototype system validates the effectiveness of such a three-dimensional navigation reference for paratroopers.

This application, which was designed as a heads-up display in night vision goggles, can be used by the DoD to enhance paratroopers' period of operations. A GPS-based three-dimensional navigation reference as described and prototyped in this research would increase a paratrooper's ability by allowing him to travel through inclement weather. Fog and low cloud cover would become covert insertion aids rather than hindrances.

### **6.3 Outline of Future Work**

This research explored the development of a primary navigation reference for paratroopers. It laid the preliminary framework for such a system. Evaluation of that framework provided positive results, and it appears the development of an operational version of the system is feasible. Refinements and enhancements of the system are also possible.

Refinement of algorithms used in the development and display of the reference's tunnel can be made. This research did not focus on the optimization of algorithms; it focused instead on finding algorithms to meet objectives. One example of an area of the system that could be optimized is the placement of tunnel edges. The current application develops a static path from release to target and places tunnel edges along the entire path. Development of an algorithm that only places edges visible to the user would reduce memory use and may make possible the enhancement of the system – a dynamic tunnel.

Enhancing the system by building a path to follow a user specified ground track is also desirable. A paratrooper's mission may take him near hostile territory or mountainous terrain. He may want to travel around such things. In such cases he could specify a ground track with two dimensional waypoints.

The addition of terrain to the reference may also be a desirable enhancement. The ability to see terrain may give users a sense of security and situational awareness.

#### **6.4 Summary**

This research investigated concepts central to the design and development of a GPS-based three-dimensional heads-up primary navigation reference for paratroopers performing HAHO jumps. It included research into the areas of GPS, three-dimensional

graphics, navigation displays, and parachute travel, among others. Knowledge gained in these four main areas was combined to design and develop a navigation reference possible to serve as a heads-up display on paratroopers' night vision goggles.

In this process, algorithms to develop a visual tunnel from a paratrooper's release point to his target were implemented. These algorithms include an algorithm to determine the path a user should take to arrive at his target. This path is defined by waypoints, and the heading the user should take at each waypoint. Another algorithm was developed to place a visual tunnel around that path to aid the paratrooper in maintaining that path.

The system developed was designed with flexibility in mind. It allows the user to specify his parachute characteristics and weather forecast information. This allows the system to operate effectively with any appropriately formatted user parachute canopy and weather forecast information.

As a result, a flexible framework for a paratroopers' primary navigation reference was developed. Future research may focus on optimization of the framework or enhance of the abilities of the reference.

## BIBLIOGRAPHY

- [BAR97] Barrows, A., K. Alter, P. Enge, B. Parkinson, and J. D. Powell. *Operational Experience with and Improvements to a Tunnel-in-the-Sky Display for Light Aircraft*. Presented at the Institute of Navigation GPS conference, Kansas City, MO, 1997.
- [BAR99] Barrows, A., and J. D. Powell. *Tunnel-in-the-Sky Cockpit Display for Complex Remote Sensing Flight Trajectories*. Presented at the Fourth International Airborne Remote Sensing Conference and Exhibition/21<sup>st</sup> Canadian Symposium on Remote Sensing, Ottawa, Canada, 1999.
- [BAR00] Barrows, A., and J. D. Powell. *Flying a Tunnel-in-the-Sky Display within the Current Airspace System*. Presented at the American Institute of Aeronautics and Astronautics 38th Aerospace Sciences Meeting, Reno, NV, 2000.
- [BER02] Berland, J. C. Email to author. 18 November, 2002.
- [BRA99] Braukus, M., and K. Barnstorff. *Synthetic Vision Could Help Pilots Steer Clear of Fatalities*. National Aeronautics and Space Administration news release. 13 May 1999.
- [COM02] Compaq Computer Corporation. "Compaq at Home." <http://athome.compaq.com/store/default.asp?page=option&ComponentID=16499> 2002
- [FOR03] Forty-Fifth Space Wing. "Navstar Global Positioning System." <https://www.patrick.af.mil/facts/gps.htm> 2003.
- [J3D99] Bouvier, D. *Getting Started with the Java3D API*. Sun Microsystems. 1999
- [JAC02] Jacobs, T. Class handout, CSCE 699, Computer Graphics using Java3D. School of Engineering and Management, Air Force Institute of Technology, Wright-Patterson AFB OH, 2002.
- [JEN00] Jennings, C., A. Barrows, K. Alter, and J. D. Powell. *Synthetic Vision Displays for Instrument Landings and Traffic Awareness – Development and Flight Testing*. Presented at the Digital Avionics Systems Conference, Philadelphia, 2000.
- [LOP02] Lopez, C. T. "SECAF, CSAF: People are Air Force's Most Valued Asset." Air Force Print News. 29 July 2002.
- [MIS01] Misra, P., and P. Enge. *Global Positioning System, Signals, Measurements, and Performance*. Lincoln, MA: Ganga-Jamuna Press, 2001.



- [MIS03] Mist Mobile Integrated System Technology Inc. "Description of Mission Planner, a Product of MMIST." <http://www.mmist.ca/MissionPlanner/description.htm> 2003.
- [MMI03] Mist Mobile Integrated System Technology Inc. "Sherpa Features, Product of MMIST." <http://www.mmist.ca/Sherpa/featuresS.htm> 2003.
- [NAV03] Navocap. "Navocap: Operational Paratroopers Navigation System, OPANAS." <http://perso.wanadoo.fr/navocap/html/ang/opanas-ang.htm> 2003.
- [ONS03] OnStar. "What is OnStar? Frequently Asked Questions." [http://www.onstar.com/visitors/html/ao\\_faq.htm](http://www.onstar.com/visitors/html/ao_faq.htm) 2003.
- [ONT99] On-Target Brochure. SSK Industries, 1999.
- [OPA03] OPANAS Brochure. <http://www.pia.com/SSK/opanas/op-page1.htm> 2003.
- [REI00] Reising, J., and M. Snow. *Curved Landing Approaches: Where is the Payoff for Pathway?* Presented at the Spatial Disorientation Symposium, San Antonio, 2000.
- [SMI02] Smith, Joshua. Email to author. 24 September, 2002.
- [SNO99] Snow, B., J. Reising, K. Liggot, and T. Barry. *Flying Complex Approaches Using A Head-Up Display: Effects of Visibility and Display Type*. 1999.
- [STE44] Stewart, J., and N. Pierce. *Marine and Air Navigation*. Boston, MA: Ginn and Company, 1944.
- [STS03] Specialized Technical Services. "HUD." <http://www.sts-eo.com/nf/hud.html> 2003
- [WIL00] Williams, K. *Impact of Aviation Highway-in-the-Sky Displays on Pilot Situational Awareness*. U. S. Department of Transportation. 2000.

## VITA

First Lieutenant Brian A. Balazs attended the Illinois Institute of Technology where he participated in the Air Force Reserve Officer Training Corps (ROTC). He graduated with high honors and received a Bachelor of Science degree in Computer Science in December 1999. Following graduation and completion of ROTC he was commissioned as a Second Lieutenant.

Lieutenant Balazs attended the Basic Communications Officer Training (BCOT) course at Keesler AFB, MS directly upon commission. After completion of BCOT, he served a 15 month short tour at Lajes Field, Azores, Portugal. In August 2001, Lieutenant Balazs entered the Air Force Institute of Technology (AFIT) as a graduate student in the Computer Science and Engineering Department. He graduated from AFIT with a Master of Science degree in Computer Science with a focus on Software Engineering.

<b>REPORT DOCUMENTATION PAGE</b>				Form Approved OMB No. 074-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p><b>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</b></p>					
<b>1. REPORT DATE (DD-MM-YYYY)</b> 25-03-2003		<b>2. REPORT TYPE</b> Master's Thesis		<b>3. DATES COVERED (From – To)</b> Aug 2001 – Mar 2003	
<b>4. TITLE AND SUBTITLE</b>  A THREE-DIMENSIONAL HEADS-UP PRIMARY NAVIGATION REFERENCE DISPLAY FOR PARATROOPERS PERFORMING HIGH ALTITUDE HIGH OPEN JUMPS				<b>5a. CONTRACT NUMBER</b>	
				<b>5b. GRANT NUMBER</b>	
				<b>5c. PROGRAM ELEMENT NUMBER</b>	
<b>6. AUTHOR(S)</b>  Balazs, Brian A., 1 Lt, USAF				<b>5d. PROJECT NUMBER</b>	
				<b>5e. TASK NUMBER</b>	
				<b>5f. WORK UNIT NUMBER</b>	
<b>7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)</b> Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way, Building 640 WPAFB OH 45433-7765				<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>  AFIT/GCS/ENG/03-02	
<b>9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> Specialized Technical Services Attn: Mr. Robert B. Stephens 2794 Indian Ripple Road Beavercreek, OH 45440				<b>10. SPONSOR/MONITOR'S ACRONYM(S)</b>	
				<b>11. SPONSOR/MONITOR'S REPORT NUMBER(S)</b>	
<b>12. DISTRIBUTION/AVAILABILITY STATEMENT</b>  APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.					
<b>13. SUPPLEMENTARY NOTES</b>					
<b>14. ABSTRACT</b> <p>The Department of Defense (DoD) relies on the para-dropping of resources to meet different objectives in order to accomplish missions during peace-time, war-time, or military operations other than war. The resources dropped to the ground via parachute range from supplies and equipment to the most valued asset, people. Tactics have been developed to increase the safety of troops parachuting into areas of conflict. These tactics include high-altitude high-opening (HAHO) jumping and night jumping.</p> <p>HAHO jumping allows paratroopers to travel large distances in the air away from the path of the delivering aircraft. While night jumping, done with the aid of night vision goggles (NVGs), provides paratroopers with the cover of night. Both of these tactics aid in avoiding detection. These techniques, however, have their drawback: low cloud cover and fog can often delay mission accomplishment due to a lack of visibility. However, low cloud cover and foggy conditions also provide a tremendous aid in covert insertion missions and enhance the element of surprise.</p> <p>This research introduces a novel application combining three-dimensional graphics and GPS for a primary navigation reference for paratroopers. It uses three-dimensional graphics to realistically portray a paratrooper's movement in the physical world, measured by GPS, as movement in a computer generated scene. This reference, presented as a heads-up display on the NVGs paratroopers already wear, facilitates mission accomplishment in cloudy and foggy conditions. Evaluation of a prototype system validates the effectiveness of such a three-dimensional navigation reference for paratroopers.</p>					
<b>15. SUBJECT TERMS</b> Parachute jumping, Parachutists, Paratroopers, High Altitude High Open (HAHO), Heads-up display, Night vision goggles, Three dimensional display systems, Three dimensional, Computer graphics, Navigation, Navigation reference, Global Positioning System (GPS)					
<b>16. SECURITY CLASSIFICATION OF:</b>			<b>17. LIMITATION OF ABSTRACT</b>	<b>18. NUMBER OF PAGES</b>	<b>19a. NAME OF RESPONSIBLE PERSON</b>
a. REPORT	b. ABSTRACT	c. THIS PAGE			Stephen P. Chambal, Capt, USAF (ENS)
U	U	U	UU	99	<b>19b. TELEPHONE NUMBER (Include area code)</b> (937) 255-6565, ext 4314; e-mail: Stephen.Chambal@afit.edu