JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 1, May 2004

Flight Simulation Architecture for Tunnel-in-the-Sky Guidance and Synthetic Vision

Michael L. Brychcy,* Ram Shanmugasundaram,† Donna N. Allen,‡ and Kenneth S. Harris§
The Boeing Company, Philadelphia, Pennsylvania

This paper will focus on the practical incorporation of "tunnel-in-the-sky" and synthetic vision concepts from a flight simulation software perspective. Piloted, rotorcraft flight simulation analysis of these concepts has been an ongoing activity in the Boeing Flight Simulation Laboratory in Philadelphia. The theoretical basis and advantages in augmenting pilot situational awareness was well presented by Robert R. Wilkins.[1] In this paper, the technical issues involving design architecture, visual perception and verification of these concepts will be addressed. Some of these issues include architecture optimization for future enhancements, tunnel correlation and conformality with out-the-window terrain, validation of pilot visual cues and integration of tunnel and synthetic vision concepts. Future work and directions will also be discussed, for both tunnel symbology-based guidance and synthetic vision situational awareness, in the context of practical realization in a rotorcraft flight simulation environment.

Nomenclature

E_G	=	east cig coordinate – ft
g	=	magnitude of acceleration of gravity
H_B	=	barometric altitude –ft
H_P	=	pressure altitude – ft
F_F	=	function factor – normalized
$F_{pv}\Delta\theta$	=	flight path vector delta pitch – deg
$F_{pv}\Delta\psi$	=	flight path vector delta yaw – deg
K_{PHI}	=	gain on bank angle
K_{TAE}	=	gain on track angle
K_{XTD}	=	gain on cross track deviation
N_G	=	north cig coordinate – ft
N_{Xwp}	=	next waypoint index – unitless
$\dot{P}_O$	=	push-over rate – deg
S_L	=	tunnel segment length – feet
S_S	=	tunnel segment spacing – feet
TAE	=	aircraft track angle error – difference between tunnel course and aircraft ground track angle – deg
T_A	=	tunnel altitude – ft
Y_L	=	aircraft latitude – (deg/min/s)
Z_B	=	tunnel flight profile height – ft
T_F	=	lead-frame time – s
T_H	=	tunnel height – ft
T_{HDG}	=	tunnel heading - deg
T_T	=	tunnel turn radius – deg
T_W	=	tunnel width - ft
T_Y	=	tunnel yaw - deg
V_A	=	true airspeed - kn
V_C	=	calibrated airspeed - kn
V_G	=	groundspeed - kn
V_R	=	reference speed - kn
V_{RC}	=	vertical velocity – ft/s
X_L	=	aircraft longitude – (deg/min/s)
XTD	=	aircraft cross track deviation - perpindicular distance from aircraft center of gravity to tunnel ground track - ft
$\dot{\phi}$	=	aircraft roll rate – deg/s
θ	=	pitch angle - deg

Received 10 October 2003; revision received 8 December 2003; accepted for publication 31 December 2003.

*Engineer, Scientist, Flight Simulation Laboratory.
†Engineer, Scientist, Flight Simulation Laboratory.
‡Lead Computer Image Generation Database Modeler, Flight Simulation Laboratory
§Engineer Scientist, Flying Qualities Laboratory.

ΔP_Y	=	power cue vertical offset – in. (display)	$\vartheta(5)$	=	fifth order polynomial quantity - unitless
ΔT	=	simulation execution time – s	ψ	=	aircraft heading - deg
ΔT_C	=	command-frame time – s	$\dot{\psi}$	=	aircraft yaw rate deg/s
ΔT_F	=	lead-frame time – s	ψ_M	=	aircraft magnetic heading - deg
ΔT_L	=	look-ahead time – s	ψ_T	=	aircraft true heading - deg
$\Delta\theta_Y$	=	pitch cue vertical offset – in. (display)			
ϕ	=	aircraft bank angle – deg			

I. Introduction

THE added utility, and some would argue necessity, of providing increased and improved situational awareness to pilots in critical flight maneuvers, has long been studied. For this purpose, the demonstrated benefit to be gained by pilots using augmented compensatory cues in cockpit displays was well presented by Wilkins.[1] For the most part, these studies have mainly been in relation to fixed-wing aircraft. Increasingly, however, rotorcraft as well are involved in certain pilot operational maneuvers where usage of situational awareness augmenting displays are being investigated. Some examples of these include precision approaches to airports in high traffic areas, negotiating mountainous terrain to avoid controlled flight into terrain (known to pilots as CFIT), hover related maneuvers tasking, etc.[1, 2]

Two basic areas addressed by a compensatory cockpit display involve pilot errors and workload. In order to reduce tracking errors, as dictated by current and future required navigation performance (RNP) standards and pilot workload, one must provide the pilot with increased situational awareness of the aircraft's relation to the desired flight path. This involves display of the current flight path or track and the actual aircraft performance, as well as the desired/commanded and/or predicted aircraft performance. The use of a perspective display with a predictive flight path symbology set and a synthetic vision underlay, representing a simulation of the pilot's outside world, provides that increased situational awareness.

The representation of a four-dimensional, or three space axes and one time axis, 'tunnel/pathway-in-the-sky' is one example of perspective display intended to provide more augmented situational awareness, in particular to aid the pilot in maintaining an intended flight path. Combining this with synthetic vision has largely been given impetus by the increased power, storage capacity and capabilities of computing and graphics processors, in both laboratory simulation and onboard aircraft avionics environments.[3]

The combination of tunnel-based and synthetic vision display cueing is the most recent situational awareness concept being explored to assist pilots. Here at Boeing Rotorcraft in Philadelphia, a National Rotorcraft Technology Center/Rotorcraft Industry Technology Association (NRTC/RITA) program has been under way since 1998 to develop these new display symbology concepts, building in part upon developments at the Technical University of Delft in the Netherlands, the Technical University of Munich in Germany, and Knighttime Systems, Inc. (of North Carolina, which assisted in development under contract).[1] This effort, in terms of a flight simulation environment will be fully discussed shortly, but first a brief background summary on prior work in the field is presented.

II. Previous Work in Situational Awareness Displays and Simulation

A. Tunnel-Based Studies

As cited by Barrows, et al.[4] some of the earliest work in 'tunnel' or 'pathway' three-dimensional perspective displays in a simulation environment was done by Grunwald, et al.[5] Tunnel concepts investigated by Barrows, et al. involved using a tunnel as a series of rectangular boxes extending in the form of a tunnel 'tube'.[3, 4] This was one type of tunnel-like symbology intended to guide the pilot along the intended flight path. Some additional situational awareness cueing was provided via an artificial horizon, with artificial ground, sky, and runway representation.

In comparing the variations of tunnel-based symbologies (e.g., a series of boxes, hoops, football-like 'goalposts', etc., as well as other symbol cues), a prime consideration has been a compromise between providing the pilot with sufficient information suited to the flight task vs excessive clutter.[3] This is particularly so in the limited space of conventional glass-cockpit head-down multifunction head-down displays (MFD). Another factor previous investigators have had to deal with, and particularly so for rotorcraft with attitude aligned perspective flight path displays, is optimal field-of-view (FOV), both in the vertical and horizontal axes. Benefits of higher FOV, where more of the flight path shows in the center of the display vs reduction in angular resolution are just some of the design criteria that have been weighed.[6, 7]

B. Synthetic Vision-Based Studies

Simulation studies have included 'tunnel only', 'synthetic vision only' or combination thereof, and recent analysis of the human factors issues of pathway cues in combination with synthetic vision variants was presented by Snow and French.[8] Of particular interest for the rotorcraft industry might include the work of Bachelder et al., where one goal was finding a solution for pilot tasking for approach-to/stay-within/depart-from hover maneuvers using night vision devices (NVD).[2] In part, the context of this study shows the benefit of 'tunnel-like' hover guidance symbologies overlaid on a nighttime synthetic vision database. This illustrates that augmenting situational awareness can be applied equally to rotorcraft flight operations as for fixed-wing aircraft

Synthetic vision concepts were explored in a simulation context by Hughes and Takallu, where suitable computing and graphics hardware to render such displays was discussed.[9] Similarly, Krohn and Jorgensen weigh the pros and cons of overlaying tunnel symbology on a synthetic vision terrain database (the approach adopted in the present paper) vs augmenting existing primary flight displays with SynVis images.[10]

Investigators have found that the display clutter and field-of-view issues already mentioned for tunnel-only displays are even more of an issue when combined with synthetic vision, and additional human factors issues such as pilot over-fixation on the symbology are introduced.[11]

There have been many more studies in this area of tunnel and synthetic vision situational awareness augmentation, but this brief survey illustrates some of the issues that were also of concern during our work here at Boeing.

III. Summary of the Program History

A. NRTC/RITA: 1998 – 1999

Boeing Philadelphia's Perspective Flight Guidance (PFG) development began in 1998 with an NRTC/RITA funded 'base program'.[1] The purpose was to develop and evaluate a perspective symbology set and compare that against current flight director symbology with respect to flight technical error (objective off-course guidance measurements) and pilot workload (subjective human factors measurements) while displaying to the pilot the aircraft positional relationship to the FAA Terminal Instrument Procedures (TERPS) criteria governing protected airspace limits for a selected approach.

B. NRTC/RITA and DUST : 2000 – Present

Following initial development and test through 1999, the program continued under joint funding of NRTC/RITA and the Dual-Use S&T Program (DUST), involving continued development and refinement of the symbology and pathway guidance software, as well as piloted simulation evaluations.

Over the duration of the program, work progressed through three phases, each of which consisted of a controls and display development effort. The first two included piloted simulation evaluations from which results were obtained and conclusions drawn concerning the viability of implementing this technology on an operational, vertical-flight-capable platform.

Phase I included developing the symbology set, determining guidance and display requirements, creating the software to meet those requirements, integrating the software into the Boeing Philadelphia V-22 simulation, and modifying the simulation to develop the control laws to drive the symbology. The theoretical basis for the tunnel symbology used was based on the work of Theunissen. [1, 12]

Phase II included software upgrades based on the conclusions of Phase I, development of a synthetic vision underlay capability, integration of the software into the Bell Helicopters Textron, Inc. (BHTI) XV-15 simulation and aircraft software, and additional simulation and flight test evaluations at Bell.

Phase III included software upgrades based on the conclusions of Phases I and II; the development of a Terrain Following/Terrain Avoidance (TF/TA) profile and demonstration; and software architectural changes, upgrades, and integration for real-time operation of flight director modes other than approach (i.e. heading hold/select, altitude hold/select).

IV. The Current System

The rest of this paper discusses the various components that comprise the Perspective Flight Guidance (PFG) system as implemented in a V22 simulation. Reviewed at a conceptual level are the tunnel system, flight director control laws system and graphics system, in terms of development issues. This is followed by a brief summary of possible future development directions.

A. Tunnel View System

The Tunnel View system is a set of software libraries that is responsible for interacting with the simulation to provide the three-dimensional perspective tunnel. Knighttime Systems, Inc developed it under contract. The Tunnel View System is described in the software design document.[13] It is a three-part system that is comprised of a flight plan manager component, display driver component and render manager component.

The Flight Plan Manager utilizes the current ownship state information, the flight plan, and display parameters to calculate a mathematical path segment. The flight plan may be pre-defined or generated dynamically as a function of the heading or altitude flight director core mode. The path segment represents the shape of the desired flight path by providing a mathematical description of a collection of points in space and how to connect to previous and next points. At certain times, the mathematical description may be broken up into a horizontal description and a vertical description. The Flight Plan Manager also provides tunnel-specific aircraft state information that is used to provide feedback into the flight controls system. The specific state information that is used to feed back into the flight controls system will be discussed later.

The Display Driver component utilizes the mathematical path segment and generates a description that can be rendered graphically. The display driver component generates a three-dimensional tunnel boundary whose volume encompasses the path segment. The tunnel is visually represented by a square tube that pitches and rolls along the desired flight path in order to achieve the desired flight conditions. If the display driver receives the mathematical description as a separate horizontal and vertical description, the information is merged into a single three-dimensional tunnel.

The Render Manager utilizes the graphical description from the Display Driver component, current aircraft state and renders the tunnel from the pilot's perspective. Integration issues involving the rendering process will be described later.

B. Single-Thread Tunnel Software Architecture

The principal design constraint to the PFG architecture is that it must be compatible with the current V22 simulation architecture and infrastructure. Figure 1 shows the original single-thread tunnel software and it's position within the framework of the simulation. Originally, the tunnel software was developed as a sequential pipeline and integrated as part of the avionics rendering emulation. A specialized set of routines provided the tunnel flight plan calculations to the flight director. One of the advantages of this architecture is that the changes required to implement are minimally invasive relative to the general simulation.

As designed, a flight plan along with some parameters to define the tunnel geometry is provided. As the simulation is initialized and the aircraft trim state is calculated, a tunnel is computed for a user-defined amount of time ahead of the aircraft. As the aircraft flies and advances through the tunnel, the tunnel geometry is recalculated in front of the aircraft so that the tunnel is visible for the same user-defined amount of time (based on current aircraft speed). The tunnel geometry is defined from the start and finish of the flight plan.

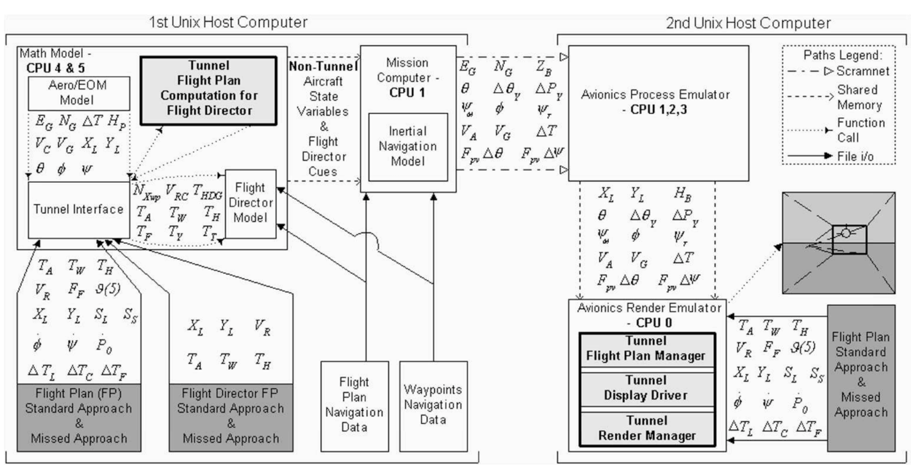

Fig. 1 Single-thread tunnel software architectural diagram.

C. Multi-Thread Tunnel Software Architecture

Figure 2 shows the various components of the tunnel software distributed across two computers and connected through reflective memory. Additionally, the Display Driver component is implemented as part of the Avionics

Process Emulation and the Render Manager component is implemented as part of the Avionics Render Emulation, which are distributed across different processors.

The main impetus leading to the creation of the multi-thread software architecture is core modes flight director operation. The core modes flight director operations provide the aircraft with heading hold and altitude hold capability. When utilizing the core modes operation, a tunnel must be generated dynamically from the current conditions to the desired aircraft heading and/or altitude. Consequently, the software provides the tunnel libraries all the necessary parameters to define a tunnel at each time frame. This functionality would be difficult to achieve if using a single processor as in the single-thread architecture without violating the simulation time frame.

The price of this capability is that approximately eight times more data is being sent from the math model/simulated mission computer to the tunnel libraries. Also, additional resident computer memory is required in the multi-thread architecture to hold the information necessary to compute the tunnel for the core modes. Currently both architectures have been successfully demonstrated at 80-hertz time frame. One advantage of the multi-thread architecture is that the calculation and rendering of the tunnel is on a separate computer. This lends itself amenable to the possibility of optimizing the tunnel calculation to run on multiple processors without modifying the rest of the architecture. This may become particularly important as the core mode algorithm develops further.

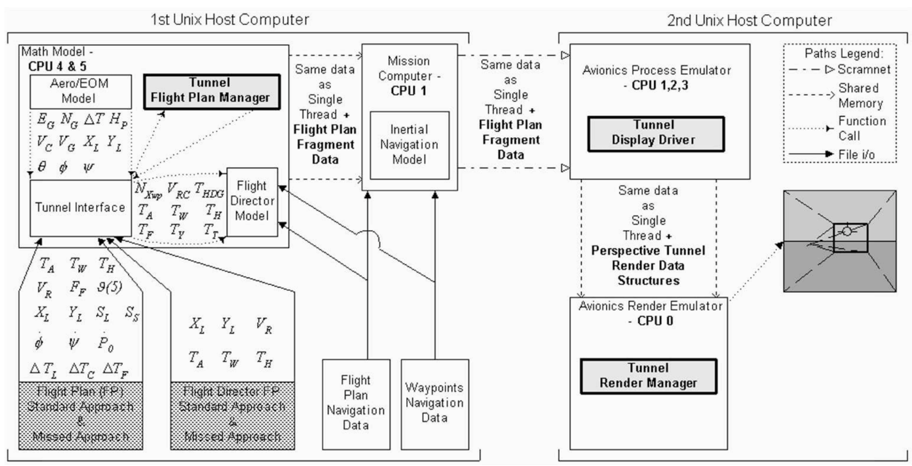

Fig. 2 Multi-thread tunnel software architectural diagram.

V. Flight Director Control Laws Component

As shown in Fig. 3, the V-22 Flight Director system operates as a closed-loop system to guide the aircraft along flight guidance commands. The components in the Guidance and Flight Director box are modeled in the simulation using MATRIXx. The guidance algorithms and flight director control laws are modeled graphically and then autocoded from the MATRIXx block diagrams before being linked into the V-22 simulation. These processes run at 20 Hz.

The Perspective Flight Guidance (PFG) tunnel was added to the simulator by modifying the flight plan format to provide the input, based on user-specified guidance and display requirements, into subcontractor-supplied software to generate the tunnel guidance and display graphics. This software was integrated into the existing math and graphics display models. Tunnel code data outputs are fed back to the flight director model and used to drive the control laws for flight director pitch and power cues and flight path vector quickening.

The pitch and power cues are displayed as command triangles added to the ends of the flight path vector (FPV) symbol, depicted in Fig. 4. The left, yellow triangle is the power command and moves within a range of ±0.85 inches above/below the FPV symbol. The cue is a “null, fly-from” command. Motion above the FPV is indication to reduce thrust control lever (TCL); motion below the FPV is indication to increase TCL. The right, orange triangle is the pitch command and moves within a range of ±1.25 inches above/below the FPV symbol. This cue is a “fly-to” command. Motion above the FPV is indication to pull back on pitch cyclic (trim up/aft); motion below the FPV is indication to push forward on pitch cyclic (trim down/forward).

The pitch and power triangles are driven by error signals generated from the difference in aircraft state feedback and tunnel code guidance feedback. A modification was made to the flight director model to provide for the option to add lead compensation to the tunnel code guidance feedback. This lead compensation is in the form of the values of tunnel code output signals at some specified time ahead of the current aircraft position. The particular signals used are the tunnel desired altitude and desired vertical velocity. These signals feed into the error signal computations. Previously, the values of these signals were sent for the current aircraft position relative to the tunnel.

With the speeds and tight tolerances required for terrain following tasks, it became evident that current state information could conflict with the desired flight path as designated by the tunnel command frame (i.e. error signals generated near the end of a descent leg would indicate control commands to descend while the command frame is indicating a need to begin climbing). In the best case, the pilot would ignore the pitch and power cues and follow the command frame; thus introducing confusion and reducing confidence in the system. In the worst case, he would attempt to null the cues, ignore the command frame, and potentially impact the ground; thus introducing a CFIT risk.

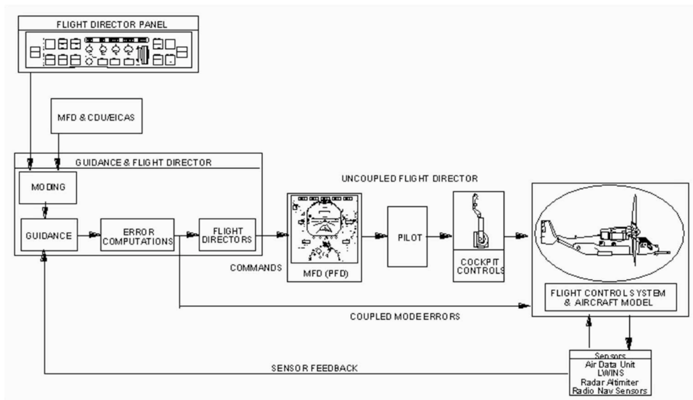

Fig. 3 System level diagram of V-22 flight director.

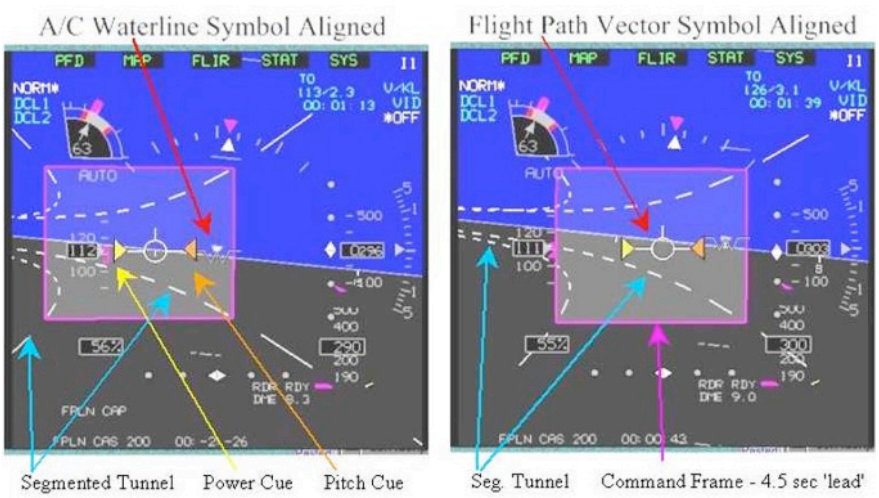

Fig. 4 Aircraft waterline symbol vs flight path vector symbol alignment.

A preliminary tuning exercise was performed in an attempt to improve coordination/synchronization of the pitch and power cues with the tunnel “command-frame” aircraft state (to be discussed below). Initially a look-ahead time equal to the command frame look-ahead time of 3.5 s was introduced for the lead-compensated signals. This, however, resulted in over-controlling with increased control input magnitudes and frequency. After some iteration, a preliminary lead-compensation look-ahead time of 2.0 s seems to coordinate the control inputs to the tunnel command frame and provide reduced vertical deviations and improved handling qualities.

The “command frame,” as seen in Fig. 4, is magenta and is shown at some designated time in front of the aircraft based on the current airspeed. The faster the aircraft is moving, the further ahead in the tunnel (and thus smaller) this frame appears. The flight path vector symbol is “quickened” to make it co-temporal with the command frame, thus making it a predictor of the future aircraft position. Hence, if the aircraft is flown with the quickened FPV (QFPV) inside of the command frame then the aircraft flight path will stay within the confines of the tunnel. The quickening algorithms are discussed next.

Vertical and Lateral FPV quickening algorithms are made available in the flight director control laws. These are implemented as follows:

$$\text{FPV horizontal displacement} = \text{FPV horizontal displacement} + \text{FPV lateral quickening} \quad (1)$$

$$\text{FPV vertical displacement} = \text{FPV vertical displacement} + \text{FPV vertical quickening} \quad (2)$$

Tuning of both vertical and lateral quickening components was performed in an effort to synchronize the FPV motion with the tunnel command frame. The effects in the vertical axis were seen to be negligible and thus quickening in that axis is nulled for piloted simulations. The lateral axis quickening was seen to have a significant effect on the piloted simulation handling qualities and thus a discussion of its implementation follows.

The FPV lateral quickening term was obtained from an approximation of the future cross track error/deviation as derived by Ashkenas[14] and appearing in Ref. 15. This is comprised of a cross track deviation component, a track angle error component, and a bank angle component.

$$\text{FPV lateral quickening} = (\boldsymbol{\delta}_{XTD} + \boldsymbol{\delta}_{TAE} + \boldsymbol{\delta}_{PHI}) * \text{first order lag} \quad (3)$$

where:

$$\boldsymbol{\delta}_{XTD} = K_{XTD} * XTD \quad (4)$$

$$\boldsymbol{\delta}_{TAE} = K_{TAE} * TAE * V_G * \Delta T_L \quad (5)$$

$$\boldsymbol{\delta}_{PHI} = \_ * K_{PH} * \phi g * \Delta T_L^2 \quad (6)$$

Tuning exercises were performed at Boeing Philadelphia for the V-22 simulator. This entailed trimming the aircraft model at conditions off the tunnel centerline (i.e. with cross track deviations, with track angle errors, and with combinations thereof) and flying to reacquire the tunnel. Numerous runs were performed to adjust the gains and lag time constant to determine trends, optimize performance, and reduce workload. The same procedure was also done while flying the steady, turning legs. The resulting gains and lag time constant for the V-22 simulation were thus derived. These were set constant across all nacelle incidence angles (I_N) (i.e. aircraft flight modes – airplane, conversion, helicopter) although the flexibility to schedule the gains for I_N was programmed into the model.

It should be noted that the resulting lateral FPV symbol motion doesn't behave like a true predictor (i.e. predicting the aircraft lateral position so many seconds into the future based on current conditions – speed and bank angle) in any of the aircraft flight modes (airplane, conversion, helicopter) due to the introduction of cross track deviation and track angle error compensation.

VI. Graphics Rendering Component

A. Coordinate System and Assumptions

A perfectly spherical earth assumption was made in terms of conversion between the geodetic and rectangular coordinate systems of the tunnel view system, as suggested in Fig. 5. However, the tunnel interface software easily allows a more sophisticated Earth model (e.g. ellipsoid based) to be used if desired. In fact, due to the fidelity of the out-of- window database, it became critical that the tunnel view libraries utilize a spherical geodetic coordinate transformation to be compatible with the other pieces of the simulation.

During the initial integration of the multi-thread software architecture, a lateral bias was observed. The magnitude of the bias appeared to be a function of direction and distance from the database origin. This bias was eventually attributed to a difference in a spherical approximation used to convert latitude and longitudes to position in the tunnel libraries, as compared to the Lambert Conic Conformal transformations used within the math model and simulated mission computer.

B. Graphics Layering Approach

For the rendering of tunnel symbologies in conjunction with standard auxiliary symbologies and synthetic vision, a multi-layered graphics approach was used, as suggested in Fig. 6. The out-the-window scene, layer (1) of Fig. 6, is based on the OpenFlight database format, and is rendered via a third party out-the-window scene generation GUI-based software toolkit.

This approach then allows for user defined overlay graphics on top of this scene, using a function callback cast as a shared object to the application. The tunnel, linear electronic attitude direction indicator (EADI), and related symbologies are then rendered, in layer (2), using OpenGL in a two-dimensional, orthographic projection mode. This allowed the tunnel 'render' library to be used as a plug-in to a third-party, two-dimensional, OpenGL/Windows-based graphical API toolkit environment, which in layer (3) of Fig. 6 rendered the topmost sundry, non-tunnel related symbologies, such as airspeed boxes, rate-of-climb indicators, and the like.

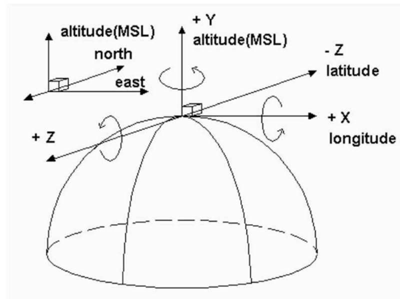

Fig. 5 Tri-part coordinate system.

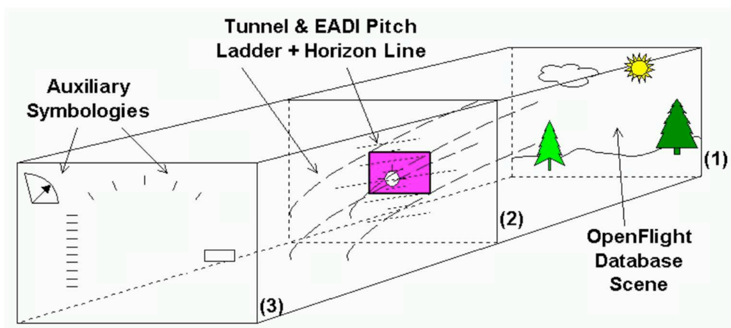

Fig. 6 Multi-layered graphics design.

To avoid complications arising in the normal OpenGL matrix stack operations and viewport aspects internal to the two-dimensional-based toolset, the tunnel three-dimensional math and three-dimensional-to-two-dimensional projection transformations were handled explicitly in the render object library. This also ensured proper clipping of the tunnel symbologies. OpenGL depth test and buffering were used for accurate and fast hidden surface removal, and upon entry to the tunnel view object, the previous depth test state is restored for continuity.

The early stages for graphics interfacing (using third-party graphical toolsets) of the tunnel and synthetic vision symbologies into the V22 simulation, made use of a Boeing in-house simulation software architecture.[16] This allowed for initial prototype and test in essentially a desktop simulation mode, followed by straightforward progression to piloted testing in our full-fidelity flight simulator.

While the third-party, two-dimensional graphical toolset supports a two-dimensional EADI (e.g. all pitch ladder 'rungs' are equidistant from each other), it was decided to render the EADI in the tunnel object as part of the three-dimensional world with fixed perspective field-of-view (FOV). This allowed a seamless and orthogonally correct co-render with the three-dimensional perspective tunnel. A consequence of this was determination of the correct vertical FOV control of the synthetic vision viewport so that the EADI and tunnel are conformal (i.e. real-world attitude changes in pitch or roll for example reflected in the SynVis image coincide with tunnel and related symbology attitude changes.

Unlike early concepts of straight lines through the four corners of sequential rectangles, here the usage of dashed lines of fixed length for both the dashes and space between them 'flying' past the pilot through the four tunnel corners, provides a temporal motion sense through the tunnel ('flow field' effect) and sense of aircraft speed relative to the ground that is a very effective cue for the pilot.

C. Synthetic Vision Database Buildup and Issues

One consequence of combining overlay tunnel symbology with underlay synthetic vision is conformality in relation to aircraft altitude, in addition to synchronization of overlay and synthetic vision for attitude changes as mentioned previously. In particular the question arises of the degree of conformality of overlaid artificial horizon with respect to the synthetic vision generated ‘true horizon’, as illustrated in Fig. 7. If the pilot were sighting along a straight-line axis through his overlay artificial horizon (assuming a heads-up (HUD) display, but this is also a valid issue for heads-down), then the ‘real’ world horizon would dip by error angle ‘A’ due to the earth’s curvature.

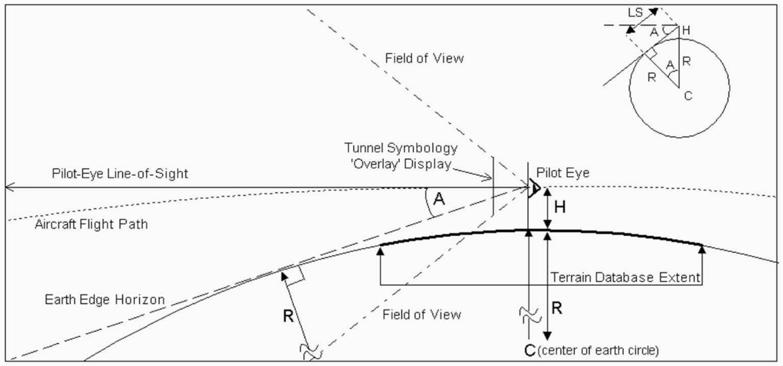

Fig. 7 Conformity error angle ‘A’ due to Earth curvature.

Assumptions and calculation quantities:

1) For very small 'H' and 'LS' relative to 'R', the triangle formed by segments 'R+H', 'LS' and 'R' is a right triangle.

2) Similarly, quantity $\frac{H}{R}$ is very small.

3) Angle 'A' is very small, so that: $\tan(A) \cong A(radians)$

4) Mean earth radius = 20.9 x 10 6 feet [17]

5) Atmospheric refraction reduces value of angle 'A', by the empirical constant 0.9216 [17]

From Figure 7 :

$$\tan(A) = \frac{LS}{R}$$

Pythagorean Theorem (and assumption 1) :

$$(R+H)^2 = LS^2 + R^2$$

$$LS = \sqrt{(R+H)^2 - R^2}$$

$$LS = \sqrt{R^2 + 2RH + H^2 - R^2}$$

$$LS = \sqrt{2RH + H^2}$$

Substituting :

$$\tan(A) = \frac{\sqrt{2RH + H^2}}{\sqrt{R^2}}$$

From assumption 2 :

$$\tan(A) = \sqrt{\frac{2H}{R}}$$

From assumptions 3,4,5 and converting to degrees:

$$A(\text{deg}) = \sqrt{\frac{2}{20.9x10^6}} * 0.9216 * \left(\frac{180}{\pi}\right) * \sqrt{H}$$

$$\therefore A(\text{deg}) = 0.016334 * \sqrt{H} \quad , H \equiv feet \qquad (7)$$

Fig. 8 Approximation of error angle ‘A’.

Based on an analysis by Fahsi* in a different context to our purposes, with some reasonable assumptions and relatively simple Euclidean mathematics, a good estimate of the functional relation of horizon dip angle error ‘A’ (deg) to aircraft altitude (ft) can be computed (see Fig. 8).

Using Eq. (7), some representative values of ‘A’ vs ‘H’ are presented in Table 1.

Table 1 Example Values from Eq. (7)

Altitude ‘H’ (ft)	Error Angle ‘A’ (deg)
30,000.0	2.83
15,000.0	2.00
7000.0	1.37
2000.0	0.73
500.0	0.36

*Data available online at http://www.research.umbc.edu/~tbenja1/umbc7/santabar/vol1/lec6/6lecture.html (cited May 2004).

This table suggests that for fixed-wing aircraft operating at high altitudes, the error in conformality between the tunnel-based overlay horizon and true earth horizon can be significant. However, for rotorcraft normally flying nap-of-the-earth maneuvers in the area of 500 to 2000 ft this error is negligible. For the purposes of our simulation, the artificial horizon was assumed, at zero deg pitch, to be collinear with the synthetic vision database horizon at all altitudes.

In addition to conformality, there are several issues that come to the forefront when trying to develop a synthetic vision visual database for flight simulation. The first is to define the unique rotorcraft requirements and then to find the source data to create the database features. Features include ground cues such as roads, railroads, rivers, lakes, fields, buildings, navigational obstacles, navigational aids, and other significant features

Terrain elevation data and feature source data are available on the Internet from the National Imagery and Mapping Agency* and the United States Geological Survey† in many formats. The federal government has data with higher fidelity, but you must have program requirements to get it. For helicopter simulation of synthetic vision the highest fidelity available should be obtained. Much of the data available from public/internet resources is so old that it is missing important features necessary for visual cueing. For example, when the initial database was built, it was missing a large 'lake' important for visual cueing in the area modeled. It was discovered that after the collection of the data a river was dammed up to form a very large reservoir. To remedy this, imagery of the area was downloaded from the Internet‡ in digital format, imported into the database tool (Terrain Experts, TerraVista Pro) and the lake was digitized into vector data and included in the database. The age of the imagery is also a factor, if it is too old, or too coarse, it will not supply the necessary source data. If you are developing a database with imagery, the projection of the imagery is also important.

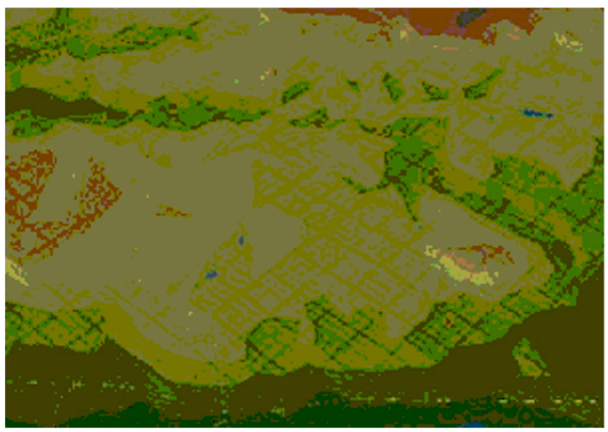

Fig. 9 SynVis terrain with contour lines.

Difficulty was encountered in placement of three-dimensional features. Desired features, not included in source data, were defined and highlighted on maps for integration. Again, age of the data was an issue, communication towers are constantly being added to the airspace, accurate data is not available for all areas. The data was several years old and the location of extremely tall communications towers was an estimate at best. With only maps to work from and interpolating measured latitude and longitude, accurate placement could not be done with confidence. Having someone stand under the desired tower with a GPS unit and recording exact measures, (latitude/longitude/altitude at the base) would give accurate placement, but would be costly (if the individual were allowed to access the location).

Different visual representations of navigational aids and obstructions were tested. Initially for the terrain, a texture grid was placed on the terrain to give the pilot visual cueing. This became a problem when the pilots mistook some of the grid lines for roads. Altitude/contour lines were placed onto the terrain with altitudes in different colors. The contour lines in the distance did not give the pilot the visual cueing necessary to determine their altitude with respect to the terrain (see Fig. 9).

*Data available online at http:// www.nima.mil (cited May 2004).

†Data available online at http://www.usgs.gov (cited May 2004).

‡Data available online at http://www.terraserver.microsoft.com (cited May 2004).

Representations of highways and railroads were tested. Data from traditional highway maps were used to indicate types of highways, for example as in Fig. 10.

Instead of roadway textures on the roads, this map legend data was used. Types of highways were denoted by color and marking designated in Fig. 10 and applied as illustrated in Fig. 11.

Fig. 10 Traditional highway markers.

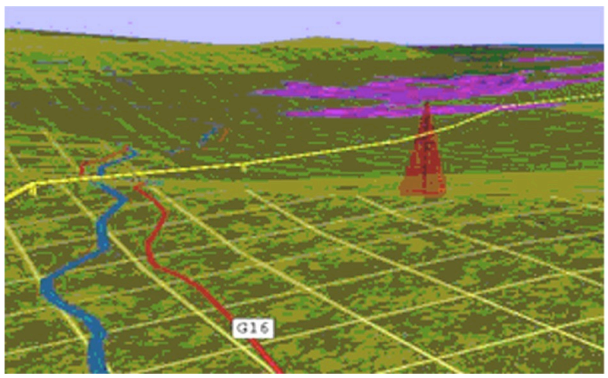

Fig. 11 Highway G16 marked appropriately.

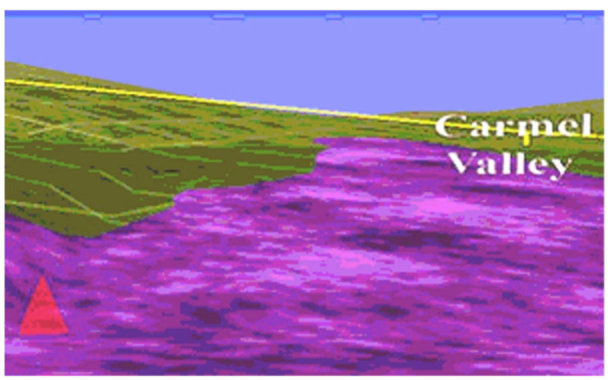

Fig. 12 City of Carmel Valley marked appropriately.

Cities were represented by colored areas on the terrain, with the city name in three-dimensional floating above it. (see Fig. 12) Air space definition was attempted with the modeling of an 'upside down wedding cake' of red transparent polygons. When the aircraft flew under the cake, it made the sky red and the transparent texture on the model did not provide sufficient cueing to determine where you were in relation to the air space.

Different types of pathway in the sky indicators or _/'s were attempted. Colors of the indicators were evaluated. White did not work against all backgrounds, neither did black. A combination of colors was determined to be the best option. The 'goalpost' was the desired indicator.

Towers were modeled to look like their representation on a map (from a distance). As the aircraft approached, they transitioned to towers with transparent 'drapes' around them indicating where the support wires were located and therefore where the pilot could not fly (see Figs. 13 and 14).

Power lines were indicated with solid towers and large yellow polygons representing the wires (see Fig. 15). The end product was used in synthetic vision simulation in a heads down display within the rotorcraft cab. A correlated out the window version of the database was displayed on a dome.

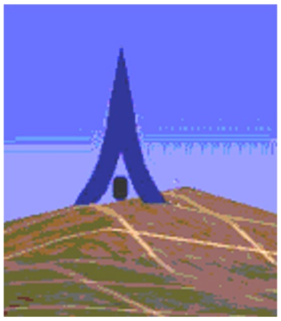

Fig. 13 Low level of detail tower.

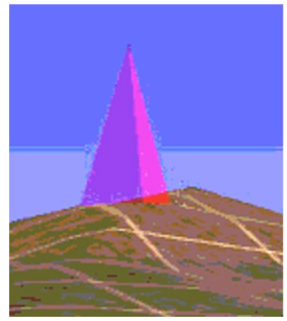

Fig. 14 High level of detail tower.

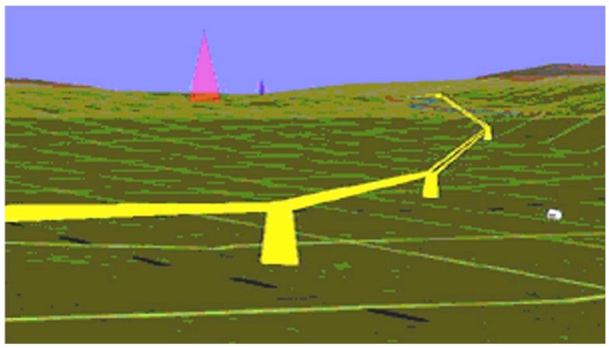

Fig. 15 Yellow power towers and power lines.

VII. Recent Development Efforts in the Tunnel Symbology

Additional features were developed in an effort to add flexibility for display optimization for future applications. One feature developed was a selectable field of view (FOV) option. For example, based on the constraints on tunnel size, aircraft speed, and command-frame look-ahead time to perform a TF/TA task, in order to achieve an acceptable (larger) tunnel size on the display the FOV needed to be reduced from the nominal 75 deg. used in approach tasks to 20 deg. This results in a magnification of the information within that region on the display. Implementing a modification to the FOV currently requires a re-compiling of the graphics task in the Boeing Flight Simulation Laboratory (FSL). This feature provides the engineer with an additional design parameter to optimize the tunnel view on the display when expanding the system's capability to other flight-directed modes.

Another feature implemented came from an idea from Theunissen at T. U. Delft to incorporate an FPV-centered display option, an option formerly available for Forward-Looking Infrared (FLIR) modes in the FSD MV-22. Conventional displays are centered about the aircraft waterline symbol. During flight at extreme attitudes (i.e. steep approaches, slow speeds) this could cause all the primary information to be confined to the bottom of the EADI with potential clipping, an undesired condition. To alleviate the anxiety associated with this condition, centering of the display about the FPV also brings the primary information back to the center of the display. This feature is currently selectable before and during run-time in the Boeing FSL. See Fig. 4 for an illustration of this distinction.

In Fig. 16 is an illustration of the Core Modes function for altitude select/hold. It can be seen here how the tunnel is curving upwards to guide the pilot to the required altitude in a steady climb.

Figure 17 illustrates the latest variant of tunnel symbology and synthetic vision combined. Shown is a 6" x 6" display field, common for many cockpit multi-function displays (MFD), with a 75 deg vertical and horizontal field-of-view (FOV). In piloted simulations, both 75 and 20 deg FOV has been used to investigate pilot perceptual and control differences, both with respect to synthetic vision cues at the periphery of the 6" x 6" display and to resolution effect on tunnel symbology at different FOV. For example, in general the command frame (magenta box of Fig 16 and 17) is larger for 20 vs 75 deg FOV, thereby providing an easier 'target' within which to maintain the flight path vector.

For a Microsoft® PowerPoint of Tunnel Evaluation Data, click here.

For a demo movie of Tunnel-SynVis TFTA, click here.

Fig. 16 Core modes — Altitude select/hold example.

Fig. 17 Tunnel and synthetic vision example.

VIII. Future Development Efforts and Direction

Based on results to date of the prior development and evaluations, future development of the PFG and Synthetic Vision systems is desired before operational systems can be fielded. It is desired to improve the flight director core modes operation and develop additional flight director modes (hover hold, depart from hover/go around (DHOV/GA)) to enhance the PFG flight director functionality. ENAV and INAV intercepts from guidance and flight director core modes need to be developed. A full simulation and flight evaluation of the flight director core modes, approaches, DHOV/GA, and hover hold modes as well as in-flight TF/TA demonstration of PFG is then required.

Integration of the Synthetic Vision system into a flight-operational platform and flight evaluations is in order as well. The application of fused sensor imaging with Synthetic Vision could also be investigated. Compatibility of the dual PFG/Synthetic Vision system with head-up displays (HUD) or head-mounted displays (HMD) could be investigated as well.

IX. Conclusions

The PFG system developed by Boeing Philadelphia has proven to be a viable guidance and display option for terminal flight operations. The system also shows promise for real-time operation of flight director core modes and for TF/TA operations as well. Given the successful modeling and simulation of this system, the favorable results to date in piloted simulation evaluation, and the immense potential for improved flight-directed operations in the future, Boeing, in conjunction with potential customers, should pursue additional development of this system. A logical next step would be to implement a "palletized" system, optimized for flight test of additional terminal operations, flight director core mode operations, and further TF/TA studies, on a research or potential customer aircraft.

Acknowledgments

Technical tasks described in this document include tasks supported with shared funding by the U.S. rotorcraft industry and government under the RITA/NASA Cooperative Agreement No. NCC2-9019, Advanced Rotorcraft Technology, December 21, 2000. Additionally, the authors gratefully acknowledge the work of Steven Williams, Software Developer at Knighttime Systems, Inc. Steve, formerly at NASA Langley Research Center, was the original architect of the tunnel generation and rendering software used in our simulation. Much thanks are also due to Robert Wilkins,[1] whose dual role as Crew Systems Engineer and expert pilot of fixed and rotary wing aircraft was crucial to the ongoing success of this project. Kudos are due also to Curtis Walz of Boeing Flying Qualities Engineering, who provided significant engineering contributions to this program, and to Duncan Hughes, of Boeing Flight Simulation Laboratory Engineering for many of his insightful inputs and discussion of synthetic vision conformality and other issues. Lastly, much appreciation is due to Steve Margetich of Boeing Flight Simulation Engineering, for his contributions in the math modeling and simulation staging aspects of this work.

References

[1]Wilkins, Jr., R. R., "Use of Predictive Guidance Displays for Increased Situational Awareness," paper presented at the American Helicopter Society 57th Annual Forum, May 2000.

[2]Bachelder, E., McRuer, D., and Hansman, R. J., "Experimental Study of three-dimensional Synthetic Cues on Rotorcraft Hover Performance," AIAA Atmospheric Flight Mechanics Conference, Aug. 2002.

[3]Barrows, A. K., and Powell, D. J., "Flying a Tunnel-in-the-Sky Display Within the Current Airspace System," AIAA Paper 2000-1059, 2000.

[4]Barrows, A. K., Alter, K. W., Enge, P., Parkinson, B. W., and Powell, J. D., "Operational Experience with and Improvements to a Tunnel-in-the-Sky Display for Light Aircraft," 10th International Technical Meeting of the Satellite Division of the Institute of Navigation, Sept. 1997.

[5]Grunwald, A. J., "Tunnel Display for Four-Dimensional Fixed-Wing Approaches," *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 7, No. 3, May-June, 1984, pp. 369-377.

[6]Theunissen, E., Sachs, G., "Spatially Integrated Data Presentation for Tiltrotors: How to Benefit from the Concepts Developed for Fixed-Wing Aircraft ?," No. 1999-01-5608, Society of Automotive Engineers, Inc., 1999.

[7]Grunwald, A. J., Robertson, J. B., and Hatfield, J. J., "Experimental Evaluation of a Perspective Tunnel Display for Three-Dimensional Helicopter Approaches," *Journal of Guidance and Control*, Vol. 4, No. 6, 1981, pp. 623-631.

[8]Snow, M. P., and French, G. A., "Human Factors in Head-Up Synthetic Vision Displays," SAE Advances in Aviation Safety Conference, 2001 Aerospace Congress, 2001.

[9]Hughes, M. F., and Takallu, Ph.D., M. A., "Terrain Portrayal for Head-Down Displays Experiment," International Advanced Aviation Technologies Conference, 2002.

[10]Krohn, P., and Jorgensen, J., "Vision 1 Synthetic Vision System," SAE Advances in Aviation Safety Conference, 2001 Aerospace Congress, Seattle, Sept. 2001.

[11]Foyle, D. C., Ahumada, A. J., Larimer, J., and Sweet, B. T., "Enhanced/Synthetic Vision Systems: Human Factors Research and Implications for Future Systems," *SAE Transactions: Journal of Aerospace*, Vol. 101, 1992, pp. 1734-1741,

[12]Theunissen, E., "Integrated Design of a Man-Machine Interface for 4-D navigation,' Delft Univ. Press, Delft, Netherlands, 1997

[13]Williams, S., "The Tunnel View System Libraries: Software Description Document," Knighttime Systems Inc., 2002. Boeing Document Number pending.

[14]Ashkenas, I., "Pilot Modeling Applications," AGARD LS-157, 1988, pp. 3-1 –3-38.

[15]Theunissen, E., Sennes, U., and Sachs, G., "Predictive Flightpath Displays for Improved Manual Control Performance," *Proceedings of the 22nd International Congress of Aeronautical Sciences*, Aug. - Sept. 2000.

[16]Brychcy, M. L., and Redkoles, P. L., "Multi-Fidelity Architecture for Rotorcraft Cockpit Simulation Training Devices," presented at the American Helicopter Society 58th Annual Forum, June 2001.