S69-13

N93-22295

146779

PILOTING DECISION AID FOR SPACECRAFT PROXIMITY OPERATIONS

Cole J. Pierce
McDonnell Douglas Space Systems Company
Houston, Texas 77062
MC: TB2EK / (713) 283-4087
FAX: (713) 283-4020

P. 3

Abstract

BACKGROUND

The concept of a decision aid to assist the piloting of a powered vehicle during a near-field (< 2000 feet) rendezvous to another spacecraft is discussed.

Using Space Shuttle rendezvous with an orbiting satellite as an example, extensive practice is normally required to successfully effect such a rendezvous with a minimum of propellant. As a rule, variations on a "point and shoot" technique are optimized and used as much as possible.

A piloting decision aid (PDA) to assist in the pointing process has been conceived and is in the preliminary stages of development. This concept may be applied to Space Shuttle proximity operations for berthing with Space Station Freedom (SSF), for Shuttle rendezvous with other spacecraft, or for autonomous rendezvous of any unmanned vehicle with SSF.

The concept originated with a task order from NASA JSC for an automated piloting procedure and was influenced by an early air-to-air missile envelope display.

DESCRIPTION

[All references to the pilot may be applied to an autopilot, as well.]

Referring to Figure 1, the pilot is presented with a computerized view of the target, e.g., SSF. This view generally corresponds to what would be seen through the overhead window of the Space Shuttle during a Shuttle rendezvous.

Superimposed over this model is a *steering circle* which defines the cone into which the vehicle must fly in order to effect the rendezvous. The offset angle limits decrease with decreasing range, allowing more tolerance at greater distance and increasing precision at close ranges. The circle is centered on the berthing facility and collapses as the vehicle nears the target.

The steering circle is modeled by a tangent function which closely approximates published acceptable angle errors for a spacecraft approaching SSF. In the PDA program, however, a maximum radius of 200 feet is assigned. At approximately 711 feet range, the circle begins to collapse, reaching a minimum of 0.5 feet when the range is zero. See Figure 2.

State vector data may be displayed on the screen. The left side is used for this purpose in the example. Crosshairs may be used to display the v-bar of the chaser vehicle.

The key to the advantage of this concept is the utilization of a *steering dot.* As depicted in the figure, the steering dot indicates where, if there were no more propulsive burns, the vehicle would intersect the yz-plane at the target. The steering dot is augmented by a cruciform pattern of small circles which depict where the steering dot would be located if a burn were initiated in each direction. This is to enable the proper timing of y- and z- burns in the manual mode and to monitor correct responses in the automatic mode.

A speed bug is displayed at top of the display to assist the pilot in keeping the closure rate of the spacecraft within allowable limits. The closure rates are range-dependent, varying from (range/2000) feet per second down to 0.275 fps (maximums) and from (range/2000 - 0.35) fps down to 0.02 fps (minimums).

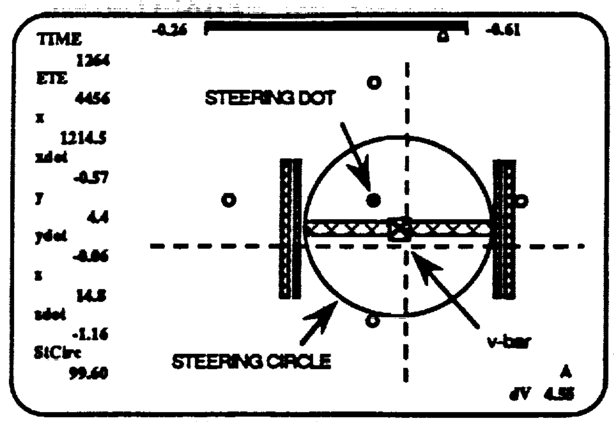
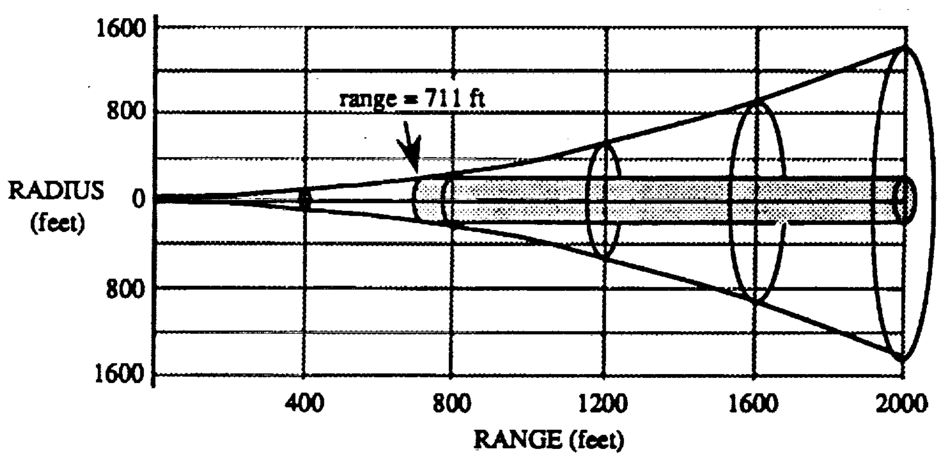


Figure 1.

The process then becomes one of simply flying the dot into the circle. Rather than experimentation and educated guessing, the pilot needs only to command burns to fly the dot into the circle and, as the circle shrinks, keep it there.



Radius (ft) = Range(ft) • TAN (0.0152 • Range + 4.92)° + 0.5

Figure 2.

The program is modularized into the following segments: (1) introduction; constants and variables defined; (2) orbital parameters defined or computed; (3) local atmospheric density and differential drag computed; (4) initial state vector defined; (5) output display generated; (6) trajectory propagated; (7) corrections computed; (8) state vector updated; and (9) output display updated.

Clohessy-Wiltshire equations of relative motion are used. The C-W equations are linearized formulations of orbital mechanics and generally accurate to better than 1 part in 100,000 at ranges of several miles. Their accuracy and speed lend themselves reliably and efficiently to application in this simulation. A relative drag factor is included.

Propulsive impulses are currently 0.10 fps in the y and z directions and 0.25 in the x. These generally conform to current simulation assumptions, although the final installation of the program will utilize the mass properties and thruster positioning and vectoring of real and simulated vehicles for a true 6-DOF simulation.

In order to propagate the position of the yz-intercept, some means of estimating the time to intercept is necessary. From the allowable maximum and minimum range rates, an average is determined which is integrated over the existing range. The resulting equation of time as a function of range closely approximates the actual times required as recorded on a series of simulations.

RESULTS

Preliminary results are promising:

Initial range	2000 ft
Initial y-offset (out-of-plane)	10 ft
Initial z-offset	20 ft
Initial x-velocity	-1.5 fps
Initial y-, z- velocities	0.0 fps
Position (y, z) tolerance at termination	± 0.5 ft

Total ΔV required for rendezvous:
10.55 fps for +vbar approach
13.75 fps for -vbar approach

For a Shuttle-type vehicle, this equates to a propellant burn of less than 100 pounds. Due to the visual cues provided, manual operation should be able to attain this efficiency with minimal practice.

CONTINUING DEVELOPMENT

In order to implement the PDA in actual systems, a precise range/range rate sensor capable of 0.1 fps resolution and a relative position sensor accurate to 0.1 foot or 0.05% are required. More accurate sensors obviously would improve the performance of the system. In addition, integration of the algorithms with existing sensor and graphics software and hardware would be necessary. Initial positioning errors and sensor inaccuracies are easily handled, as y and z errors of 200 feet at a range of 700 feet are within allowable tolerances. Errors in excess of these figures will normally only require a single initial burn to effect a nominal near-field rendezvous. An abort algorithm has been considered and would be relatively simple to implement.

The program is currently PC-based, with capability to support both CGA and VGA graphics, but is being rewritten in C as it becomes fully developed and verified. Although currently limited to point masses and unit vector propulsion, the program is being expanded and will be included in the NASA/JSC Orbital Operations Simulator (OOS) man-in-the-loop, real-time, 6-DOF simulation. Meanwhile, the PC-based versions will serve as stand-alone proof-of-concept programs as well as providing a demonstration capability at the desktop.

This work is being funded by the Systems Engineering Branch of the Engineering Directorate, NASA/JSC.