Linear-Quadratic Stationkeeping for the STS Orbiter

David C. Redding* and Neil J. Adams*
Charles Stark Draper Laboratory, Cambridge, Massachusetts
and
Edward T. Kubiak†
NASA Johnson Space Center, Houston Texas

An automatic controller for Space Shuttle Orbiter stationkeeping and formationkeeping is presented. The controller uses feed-forward control for precise stationkeeping at nonequilibrium set points and for efficient position maneuvers, and a discrete-time linear-quadratic feedback regulator for disturbance rejection. Design of the regulator emphasizes good limit-cycling performance in the presence of sensor noise and a control threshold nonlinearity. Results show that 20-ft precision can be achieved using current Orbiter sensors, and that 2-ft precision would be possible with enhanced sensors.

I. Introduction

THE SPACE Shuttle Orbiter is capable of precise control of its position in orbit, relative to a particular orbital station (stationkeeping) or relative to another satellite (formationkeeping).[1] Most such operations flown to date have been under manual control. The Orbiter pilot uses data from an inertial measurement unit (IMU), a ranging radar, and ground-tracking inputs, processed in an onboard navigation Kálmán filter to provide complete Orbiter and target satellite state information. This information plus direct visual observation is used by the pilot to determine required translational velocity corrections, which he implements using a cockpit-mounted control stick to activate body-fixed reaction control (RCS) jets.

In this paper, we describe a new stationkeeping/formationkeeping controller that automates some of the manual procedures currently used. The new controller, which we call "linear-quadratic stationkeeping" (LQS), uses a discrete-time linear-quadratic regulator for feedback control. It also uses a feed-forward loop for precise control at nonequilibrium set points, such as points above or below the reference point, or points moving along a constrained path.

The controller takes as its basic input a target position and velocity relative to a specified station or satellite. If the target differs from the current position and velocity by more than specified amounts, the controller automatically computes a maneuver to attain the desired state. Maneuver trajectories follow either a straight-line path or a null-force coasting path. The feed-forward loop is used to initiate and terminate the maneuvers. The regulator provides continuous closed-loop control, both during maneuvers, to keep the Orbiter tracking the maneuver trajectory, and when not maneuvering, to keep it on station.

The advantages of an automatic controller include reduced pilot workload, especially for long-duration formationkeeping at medium distances, and improved performance (as measured by precision of relative position control, fuel consumption, and RCS jet duty cycles).

Presented as Paper 86-2222 at the AIAA/AAS Astrodynamics Conference, Williamsburg, VA, Aug. 18-20, 1986; received Sept. 11, 1987; revision received Jan. 6, 1988. Copyright © 1988 by C. S. Draper Lab. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.

*Staff Engineer. Member AIAA.
†Project Engineer.

Very high levels of performance can be attained, depending on sensor accuracy and control-jet granularity. Current sensors allow precision of about 20 ft and 0.05 fps. Possible future Orbiter enhancements include a new laser relative-position sensor,[2] which will improve navigation state estimation by a factor of 30; and new jet-firing command software, which could reduce control granularity by a factor of 10.

The LQS has been proposed as a software upgrade to the Orbiter digital autopilot (DAP).

Previous work in this area includes discussion by Bryson of on-off and continuous stationkeeping controllers.[3] Vassar and Sherwood developed a digital stationkeeping control approach[4] that we extend in this paper. A "phase-space" on-off stationkeeping controller was used as part of the OEX experimental autopilot, which has been flight-tested on the Orbiter.[5] This approach was extended to incorporate a coupled position and attitude-maneuver capability.[6] An alternate approach to long-term stationkeeping, involving large limit cycles on traveling ellipses, was developed by Gustafson and Kriegsman.[7] A thorough review of optimal orbital maneuvers is provided by Marec.[8]

The work presented here has been extended to utilize the traveling ellipse approach to stationkeeping.[9] Chen applied LQG/LTR techniques as an alternative to the LQS feedback regulator.[10]

The remainder of the paper is organized as follows. Section II presents the difference equations used to describe Orbiter motion. Key issues in the design of the feedback control loop are the selection of gains and sample time for desirable performance characteristics in the presence of sensor noise and control nonlinearities. These are discussed in Sec. III.

Section IV presents the feed-forward logic. Steady-state solutions of the linearized relative-motion orbit equations provide incremental Δv's to be applied in nonequilibrium conditions. We discuss feedforward implementation of two-burn maneuvers, which are often fuel-optimal, and straight-line maneuvers.

The LQS controller computes velocity change (Δv) commands in local vertical, local horizontal (LVLH) coordinates. They are converted to body-frame jet-firing commands in a separate Δv loop discussed in Sec. V.

Simulation results showing performance in representative tasks are presented in Sec. VI. Section VII provides a statement of conclusions to date and a discussion of remaining issues and tasks.

II. Difference Equations

Relative motion of the Orbiter with respect to a target satellite (or other reference point) in circular orbit is most conveniently described in local vertical, local horizontal (LVLH) coordinates (see Fig. 1). If we assume the positions of the Orbiter and reference point are close (X_{LVLH} can be large if measured circumferentially) and the reference orbit is circular, Orbiter motion is governed by the well-known Clohessey–Wiltshire (or Euler–Hill) equations

$$\ddot{x} = 2\omega_0 \dot{z} + f_x \tag{1}$$

$$\ddot{y} = -\omega_0^2 y + f_y \tag{2}$$

$$\ddot{z} = -2\omega_0 \dot{x} + 3\omega_0^2 z + f_z \tag{3}$$

Here, ω_0 is the orbital rate of the reference point; x, y, and z are the LVLH components of position (Fig. 1); and f_x, f_y, and f_z are LVLH normal forces.

It is convenient to measure time in units of $(1/\omega_0)$ in (1–3), in order to remove their dependency on orbital altitude. We define $\boldsymbol{r} = (x, y, z)^T$ as the LVLH position vector and $\boldsymbol{v} = (\dot{x}, \dot{y}, \dot{z})^T$ as the LVLH velocity. The translational state vector $\boldsymbol{x}$ is thus $\boldsymbol{x} = (\boldsymbol{v}^T, \boldsymbol{r}^T)^T$. If we assume a constant LVLH force $\boldsymbol{f}_i$, the solution of (1–3) over a period $\tau = \omega_0(t_{i+1} - t_i)$ (normalized sample time t_s) is

$$\boldsymbol{x}_{i+1} = \begin{bmatrix} T_{vv} & T_{vr} \\ T_{rv} & T_{rr} \end{bmatrix} \boldsymbol{x}_i + \begin{bmatrix} T_{vf} \\ T_{rf} \end{bmatrix} \boldsymbol{f}_i \tag{4}$$

where

$$T_{vv} = \begin{bmatrix} (4c-3) & 0 & 2s \\ 0 & c & 0 \\ -2s & 0 & c \end{bmatrix} \tag{5}$$

$$T_{vr} = \begin{bmatrix} 0 & 0 & 6(1-c) \\ 0 & -s & 0 \\ 0 & 0 & 3s \end{bmatrix} \tag{6}$$

$$T_{rv} = \begin{bmatrix} (4s-3\tau) & 0 & 2(1-c) \\ 0 & s & 0 \\ 2(c-1) & 0 & s \end{bmatrix} \tag{7}$$

$$T_{rr} = \begin{bmatrix} 1 & 0 & 6(\tau - s) \\ 0 & c & 0 \\ 0 & 0 & (4-3c) \end{bmatrix} \tag{8}$$

$$T_{vf} = \frac{1}{\omega_0^2} \begin{bmatrix} (4s-3\tau) & 0 & 2(1-c) \\ 0 & s & 0 \\ 2(c-1) & 0 & s \end{bmatrix} \tag{9}$$

$$T_{rf} = \frac{1}{\omega_0^2} \begin{bmatrix} (4-4c-3\tau^2) & 0 & 2(\tau - s) \\ 0 & (1-c) & 0 \\ 2(s-\tau) & 0 & (1-c) \end{bmatrix} \tag{10}$$

The forces $\boldsymbol{f}$ include environmental effects and jet forces. The RCS jets on the Orbiter are of sufficiently high thrust, however, that they can be treated as impulsive changes in velocity, or Δv's. Other forces can be ignored in the control design process. If we assume that the jet impulses are imparted at the start of each stage, the system difference equation (4) becomes

$$\boldsymbol{x}_{i+1} = \begin{bmatrix} \boldsymbol{v}_{i+1} \\ \boldsymbol{r}_{i+1} \end{bmatrix} = \begin{bmatrix} T_{vv} & T_{vr} \\ T_{rv} & T_{rr} \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_i + \Delta \boldsymbol{v}_i \\ \boldsymbol{r}_i \end{bmatrix} \tag{11}$$

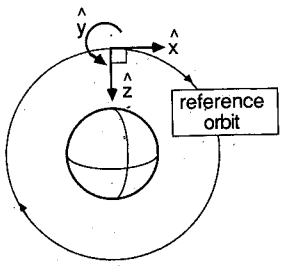

Fig. 1 LVLH coordinate system.

or

$$\boldsymbol{x}_{i+1} = \Phi \boldsymbol{x}_i + \Gamma \Delta \boldsymbol{v}_i \tag{12}$$

where Φ is the full-system transition matrix and $\Gamma^T = [T_{rr}^T, T_{vr}^T]$.

This formulation (14) was developed by Vassar and Sherwood.[4] It expresses the propagation of the full state as an explicit function of the controls (the jet Δv's) in a form convenient for controls analysis.

III. Feedback Control

The objectives of our feedback control design are to minimize fuel consumption and jet firings while providing good linear-response characteristics and small errors. The task is complicated by the presence of sensor noise and control-jet granularity.

We achieve efficient control by designing for specific limit-cycling characteristics. We model jet granularity as a threshold nonlinearity, so that control firings are done only when the commanded Δv is greater than Δv_{thresh} (in some axis). Position and velocity deadzones are then created indirectly, by choice of the elements of the quadratic cost matrices.

The minimum time between firings is explicitly set with the choice of the controller sample time t_s. The times used are quite long, up to 2 min, to provide acceptable error performance with few jet firings. The resulting designs provide good linear response for larger errors and good limit-cycling behavior when errors are small. Very large errors are handled by the feed-forward loop as maneuvers.

The main sensor for formationkeeping is the Orbiter ranging radar, which provides information on target satellite direction and range. The unfiltered radar data are corrupted by random noise of large amplitude ($\sim$30 ft). The navigation filter combines the radar inputs with IMU and other data to produce the estimated translational state used by the LQS. The filter outputs, sampled every 20–100 s, are still quite noisy, with errors of up to 15 ft. To avoid worst-case jet firings that would drive the Orbiter away from the set point, we designed deadzones of about 20 ft in position and about 0.05 fps in velocity.

The RCS control jets have fixed thrust levels and are restricted to a minimum of 80 ms of on-time. This results in significant control granularity and (necessarily) finite errors. For the large (870 lb thrust) primary RCS jets, this granularity is 0.02–0.08 fps, depending on the direction of the Δv. Currently, only these jets are used in translational control. Use of the smaller (24 lb thrust) vernier RCS jets in conjunction with the PRCS would reduce this granularity to below 0.01 fps on the average. This capability has been demonstrated using the OEX experimental autopilot.[3]

The RCS jets are also used to control attitude. Coupling of attitude firings into translational motion represents the largest unmodeled disturbance for LQS control. Other disturbances include aerodynamic drag and eccentricity of the reference orbit.

To apply the standard discrete-time, linear plant, quadratic cost optimal regulator techniques,[10,11] we assume a constant-gain controller of the form

$$\Delta \boldsymbol{v}_i = \kappa(\boldsymbol{x}_c - \boldsymbol{x}_i) \tag{13}$$

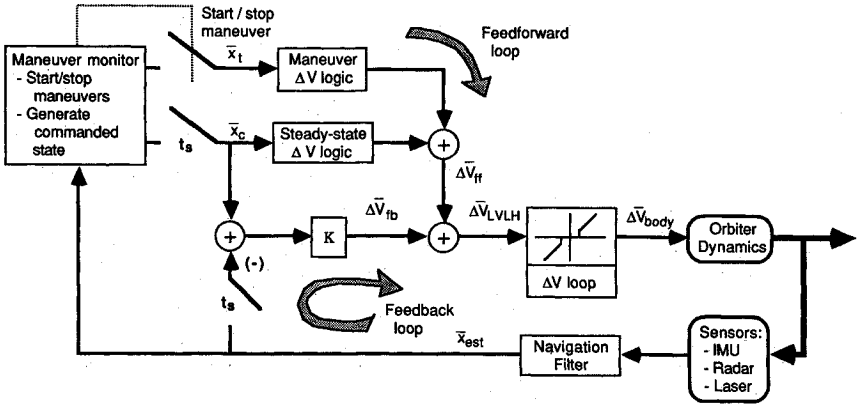

Fig. 2 Sketch of the LQS controller.

(see Fig. 2). The constant-gain matrix κ is computed to minimize the quadratic cost function

$$J = \sum_{i=1}^{\infty} [x_i^T Q x_i + \Delta v_i^T R\, \Delta v_i] \tag{14}$$

To select the elements of the cost matrices Q and R, we apply Bryson's rule[10,11] (of thumb), which states that Q and R should be diagonal, with elements q_{ii} and r_{ii} equal to the inverse square of the nominal deviation of the ith output x_{nom} and control Δv_{nom}. This selection of cost elements allows variations of each state and control to participate in the cost on a fixed percentage basis. Roughly speaking, if the error x is x_{nom}, then the commanded Δv will be Δv_{nom}. In our case, we set Δv_{nom} to the firing threshold level Δv_{thresh}. Then when the state errors are less than x_{nom}, the commanded Δv is less than Δv_{thresh}, and no firings result. Thus, we create a deadzone defined approximately by x_{nom} and Δv_{nom}.

To keep the number of control design parameters to a manageable few, we weight each velocity state, each position state, and each control equally. In normalized time,

$$Q = \text{diag}\left[\left(\frac{1}{v_{\text{nom}}^2}\right), \left(\frac{1}{v_{\text{nom}}^2}\right), \left(\frac{1}{v_{\text{nom}}^2}\right), \left(\frac{1}{r_{\text{nom}}^2}\right), \times\left(\frac{1}{r_{\text{nom}}^2}\right), \left(\frac{1}{r_{\text{nom}}^2}\right)\right] \tag{15}$$

$$R = \text{diag}\left[\left(\frac{1}{\Delta v_{\text{nom}}^2}\right), \left(\frac{1}{\Delta v_{\text{nom}}^2}\right), \left(\frac{1}{\Delta v_{\text{nom}}^2}\right)\right] \tag{16}$$

The parameters that define the control design are the sample time t_s, the state-weighting parameters v_{nom} and r_{nom}, and the control-weighting parameter Δv_{nom} (except for the laser case, which uses off-diagonal elements in the x, z positions).

The size of this deadzone is chosen to reduce sensitivity to sensor error, as discussed. Picking $r_{\text{nom}} > 15$ ft provides that most spurious signals will not excite a jet firing. Those that do will at least not send the Orbiter in the wrong direction. Filtered radar velocity error is about 0.05 fps, so selecting $v_{\text{nom}} > 0.1$ fps will similarly reduce sensitivity to spurious signals.

Sample time t_s can be selected to give a desirable limit-cycle period or rise time. Generally, longer times between firings are preferred on the basis of fewer jet cycles and better fuel efficiency. Smaller t_s are required in extreme disturbance environments, or when tight control is desired. We have found that with the radar active, a rise time of about 200 s is desirable, corresponding to a Δv of 0.1 fps in response to a 20-ft error. This leads to selection of t_s to be 60–100 s, values that are very much larger than the typical feedback firing time (< 1 s), consistent with our assumption of impulsive Δv's.

Specific values for the cost-weighting matrices and t_s for use with the radar were chosen by systematically varying the cost

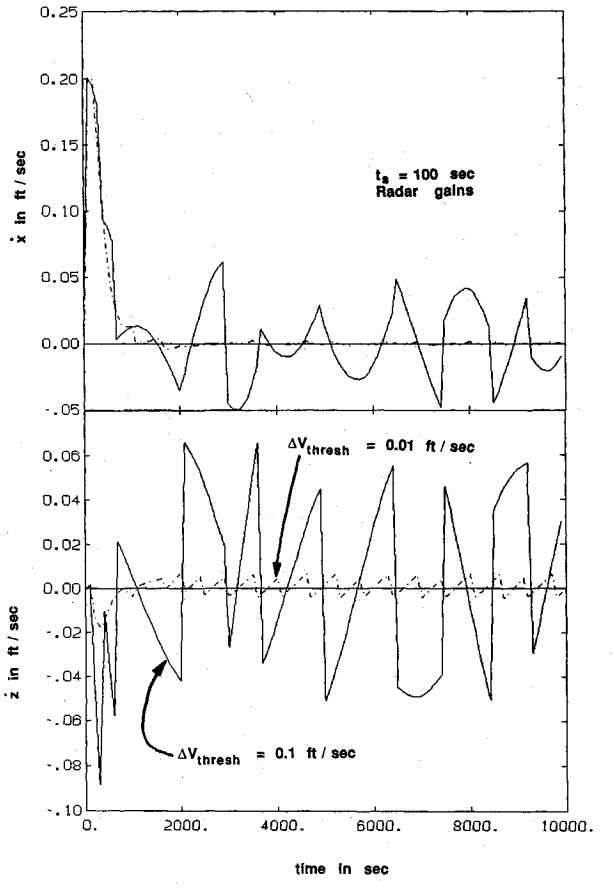

Fig. 3 Velocity response to a position command: radar gains.

Table 1 Feedback gains

Radar gains ($t_s = 100$ s):

$$K = \begin{bmatrix} 0.7488 & 0 & -0.0009 & 2.1246\omega_0 & 0 & -0.4273\omega_0 \\ 0 & 0.7473 & 0 & 0 & 2.1005\omega_0 & 0 \\ -0.0009 & 0 & 0.7567 & 0.4214\omega_0 & 0 & 2.3127\omega_0 \end{bmatrix}$$

Laser gains ($t_s = 15$ s):

$$K = \begin{bmatrix} 0.7372 & 0 & -0.1044 & 21.541\omega_0 & 0 & -5.1637\omega_0 \\ 0 & 0.7904 & 0 & 0 & 19.772\omega_0 & 0 \\ 0.1044 & 0 & 0.7512 & -3.981\omega_0 & 0 & 21.174\omega_0 \end{bmatrix}$$

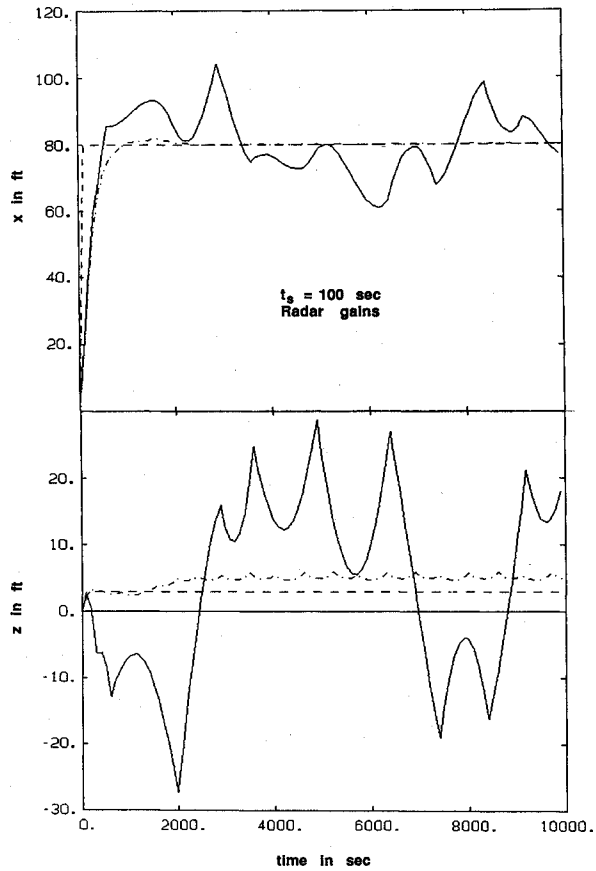

Fig. 4 Position response to a position command: radar gains.

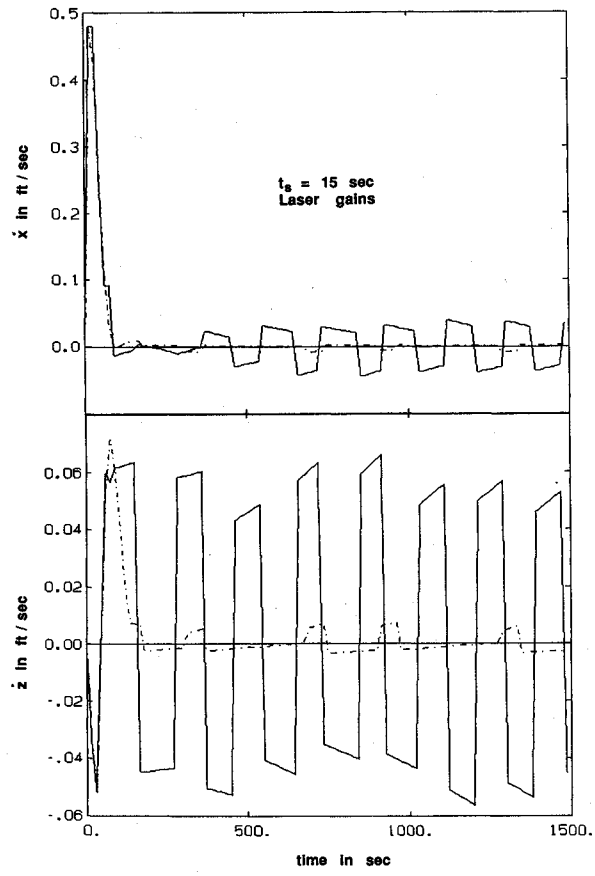

Fig. 5 Velocity response to a position command: laser gains.

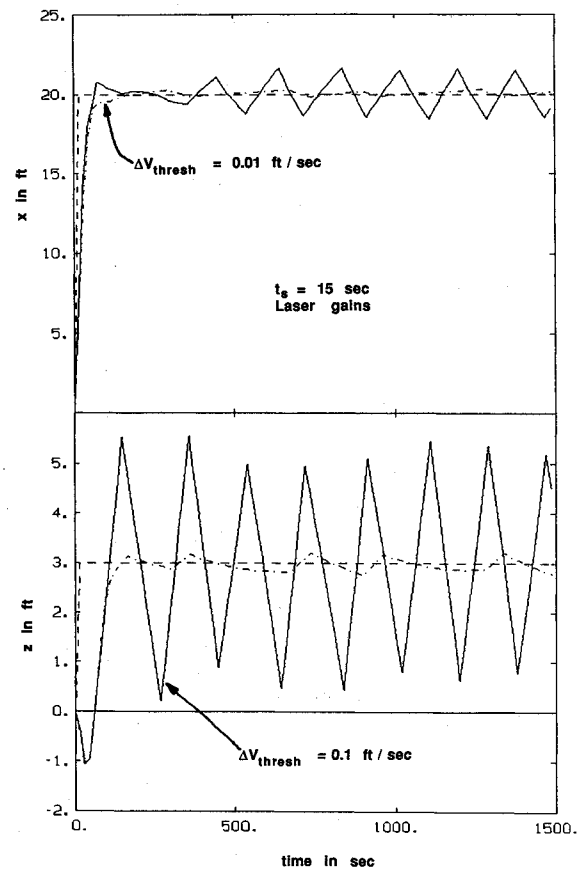

Fig. 6 Position response to a position command: laser gains.

parameters within the bounds just discussed and examining the resulting time-response characteristics. The values chosen are $\Delta v_{nom} = 0.1$ fps, $r_{nom} = 20$ ft, and $v_{nom} = 0.1$/s. The resulting gains are given in Table 1. An example showing undisturbed response to a step command in x and z is given in Figs. 3 and 4, showing results with two levels of Δv_{thresh}. The $\Delta v_{thresh} = 0.1$ fps is the more realistic case and is the design point. The $\Delta v_{thresh} = 0.1$ fps represents performance that could be realized with the new jet-selection logic,[2] combining the primary and vernier RCS jets to obtain more precise Δv control. The performance shown also assumes use of a more accurate sensor than the radar.

With the normal Δv threshold, the controller has a deadzone of about 20 ft on either side of the commanded position. With errors inside this deadzone, no firings are done. When the error is greater than about 20 ft, the controller fires to reduce it, which is the desired performance.

More detailed simulation results, including sensor noise and plant disturbances, are presented in Sec. VI.

Stationkeeping using the proposed laser ranger presents a slightly different set of problems. The filtered laser estimates of the relative state contain errors on the order of 0.5 ft and 0.01 fps. The control Δv threshold remains about 0.08 fps for the standard jet logic.

We desire to keep errors below about 2 ft, so we select $r_{nom} = 2$ ft, $v_{nom} = 0.1$ fps, and $\Delta v_{nom} = 0.1$ fps as initial values.

In this case, using diagonal cost matrices leads to distinct "beating" in the x and z outputs. Adding off-diagonal elements to Q to penalize the x-z terms eliminated the beating in favor of more uniform deadbanding. These terms are $Q_{xz} = Q_{zx} = 1/(0.08 \text{ fps})^2$. The laser gains are given in Table 1. This creates a deadzone of about 2 ft on either side of the commanded position, consistent with the expected error level. The velocity following a control firing can be expected to be about 0.08 fps, so the deadzone is traversed in about 50 s. Thus, a sampling

time t_s of 15–30 s would be appropriate. Undisturbed time response to an x-z command is shown in Figs. 5 and 6. Again, more detailed results are presented in Sec. VI.

The LQS feedback controller has the advantages of simplicity and a compact computer implementation, while delivering the desired performance. It is versatile, predictable, reliable, and compatible with standard Orbiter operations. Other linear feedback designs are possible (e.g., Ref. 10).

IV. Feed-Forward Control

The LQS feed-forward loop performs two main functions (Fig. 2). The first is to compute open-loop Δv control in nonequilibrium steady-state conditions. These occur when stationkeeping away from $z = 0$ or $y = 0$, i.e., below or above the reference point, or out of the reference orbital plane. They also occur during straight-line or other constrained maneuvers.

The second function is the maneuver monitor. If the Orbiter position differs from the commanded position or velocity by a crew-specified amount (nominally twice the feedback loop deadzone), the maneuver monitor computes an open-loop maneuver from the current to the desired position, following either a straight-line, constrained trajectory, or a two-burn coasting trajectory. The maneuver logic starts the maneuver by commanding initial feed-forward Δv firings and moving the feedback loop setpoint x_c incrementally along the open-loop nominal trajectory. When the nominal trajectory reaches the desired point, feed-forward Δv firings terminate the maneuver.

The feed-forward loop normally executes every t_s seconds, together with the feedback loop, except when a maneuver is commanded by the crew, or is terminated. These conditions signal immediate execution of the feed-forward maneuver Δv logic, which also restarts the feedback loop at that time. Feedback control Δv's are added directly to the feed-forward Δv (Fig. 2).

Steady-state feed-forward control is determined directly from (1–3). We command $\boldsymbol{x}_c = (\dot{x}_c, \dot{y}_c, \dot{z}_c, x_c, y_c, z_c)$, where $\ddot{x}_c = \ddot{y}_c = \ddot{z}_c = 0$. Thus,

$$0 = 2\omega_0 \dot{z}_c + f_x \tag{17}$$

$$0 = -\omega_0^2 y_c + f_y \tag{18}$$

$$0 = -2\omega_0 \dot{x}_c + 3\omega_0^2 z_c + f_z \tag{19}$$

Integrating (19–21) over t_s to obtain the feed-forward Δv_{ff} required to maintain this commanded state (in real units) yields

$$\Delta v_{x_{ff}} = -2\omega_0 t_s \dot{z}_c \tag{20}$$

$$\Delta v_{y_{ff}} = \omega_0^2 t_s y_c \tag{21}$$

$$\Delta v_{z_{ff}} = \omega_0 t_s (2\dot{x}_c - 3\omega_0 z_c) \tag{22}$$

The total (feed-forward + feedback) control is computed every t_s seconds as

$$\Delta \boldsymbol{v} = \Delta \boldsymbol{v}_{ff} + \kappa(\boldsymbol{x}_c - \boldsymbol{x}) \tag{23}$$

The commanded state x_c is computed by the maneuver monitor as follows. Crew commands, including target position r_t, maneuver type, maneuver speed v_m, position deadband r_{db}, and velocity deadband v_{db}, are continuously read into the flight-control computers from crew cockpit data-entry panels. If a maneuver is not being done, then the commanded state is set to the target state ($\boldsymbol{r}_c = \boldsymbol{r}_t$, $\boldsymbol{v}_c = 0$). Current state is compared to the commands. If

$$|\boldsymbol{r}_c - \boldsymbol{r}| > r_{db} \tag{24}$$

or

$$|\boldsymbol{v}_c - \boldsymbol{v}| > v_{db} \tag{25}$$

then a maneuver is computed.

For both maneuver types, end-time is computed from the requested maneuver speed v_m as

$$t_f = |\boldsymbol{r}_c - \boldsymbol{r}|/v_m \tag{26}$$

If the straight-line maneuver type is requested, the commanded velocity is

$$\boldsymbol{v}_c = (\boldsymbol{r}_c - \boldsymbol{r}/|\boldsymbol{r}_c - \boldsymbol{r}|)\, v_m \tag{27}$$

The maneuver is started immediately by firing a feed-forward Δv equal to the difference of $\boldsymbol{v}_c$ and $\boldsymbol{v}$, plus one-half of the steady-state control required to keep tracking the straight-line maneuver path

$$\Delta v_{x_0} = \dot{x}_c - \dot{x} - \omega_0 t_s \dot{z}_c \tag{28}$$

$$\Delta v_{y_0} = \dot{y}_c - \dot{y} + \tfrac{1}{2}\omega_0^2 t_s y_c \tag{29}$$

$$\Delta v_{z_0} = \dot{z}_c - \dot{z} + \tfrac{1}{2}\omega_0 t_s (2\dot{x}_c - 3\omega_0 z) \tag{30}$$

Commanded position $\boldsymbol{r}_c = (x_c, y_c, z_c)^T$ is set to current position at the start of the maneuver and incremented at each subsequent t_s as

$$\boldsymbol{r}_{c_{i+1}} = \boldsymbol{r}_{c_i} + \boldsymbol{v}_c t_s \tag{31}$$

The maneuver is terminated when $t = t_f$, by setting $v_c = 0$, $r_c = r_t$ and firing a final Δv of the form of (20–22).

The two-burn, coasting-arc maneuver type is handled similarly, except no steady-state control is required to keep tracking the maneuver path. This means that total fuel expended during the maneuver will be lower than for a straight-line maneuver. Indeed, this type of orbital rendezvous trajectory is often globally fuel-optimal.[8] Straight-line maneuvers are sometimes preferable for operations in proximity to other satellites, however, as they can be easier for the crew to visualize. Also, the two-burn maneuvers are run only when maneuver time is less than π/ω_0, to avoid ambiguity in (4).

The initial Δv is computed using (4) with $\tau = \omega_0(t_f - t_0)$ as

$$\Delta \boldsymbol{v}_0 = \boldsymbol{v}_0 - \boldsymbol{v} \tag{32}$$

where

$$\boldsymbol{v}_0 = \omega_0 T_{rv}^{-1}(\boldsymbol{r}_t - T_{rr}\boldsymbol{r}_0) \tag{33}$$

($\boldsymbol{r}_0 = \boldsymbol{r}$ at the start of the maneuver).

The commanded state is updated every t_s to provide setpoints for the feedback loop, again using (4), as

$$\boldsymbol{r}_c = T_{rr}\boldsymbol{r}_0 + (1/\omega_0)\, T_{rv}\boldsymbol{v}_0 \tag{34}$$

$$\boldsymbol{v}_c = \omega_0 T_{vr}\boldsymbol{r}_0 + T_{vv}\boldsymbol{v}_0 \tag{35}$$

with $t = w_0(t - t_0)$. The final maneuver burn is done as before, when $t = t_f$, using (28–30).

The maneuver monitor is also run during maneuvers, to check that the maneuver target has not been changed by the crew, and to compare current position with the commanded point on the maneuver trajectory. This is done using (24) and (25), with x_c set by (27) and (31) or by (34) and (35). If r_{db} or v_{db} are exceeded, a new maneuver is computed.

Other options exist for computing feed-forward maneuver types. Maneuvers Δv's tend to be quite large, so that modeling them as finite-thrust burns rather than impulsive Δv's can lead to improved efficiency. More accurate two-burn maneuver solutions are discussed in Ref. 6.

A second option is for nonstationary stationkeeping. If the required stationkeeping zone is fairly large, cyclic, looping trajectories within this zone can be done at a reduced fuel cost relative to stationkeeping at a single point.[7] Extension of the LQS to track these sorts of trajectories is discussed in Ref. 9.

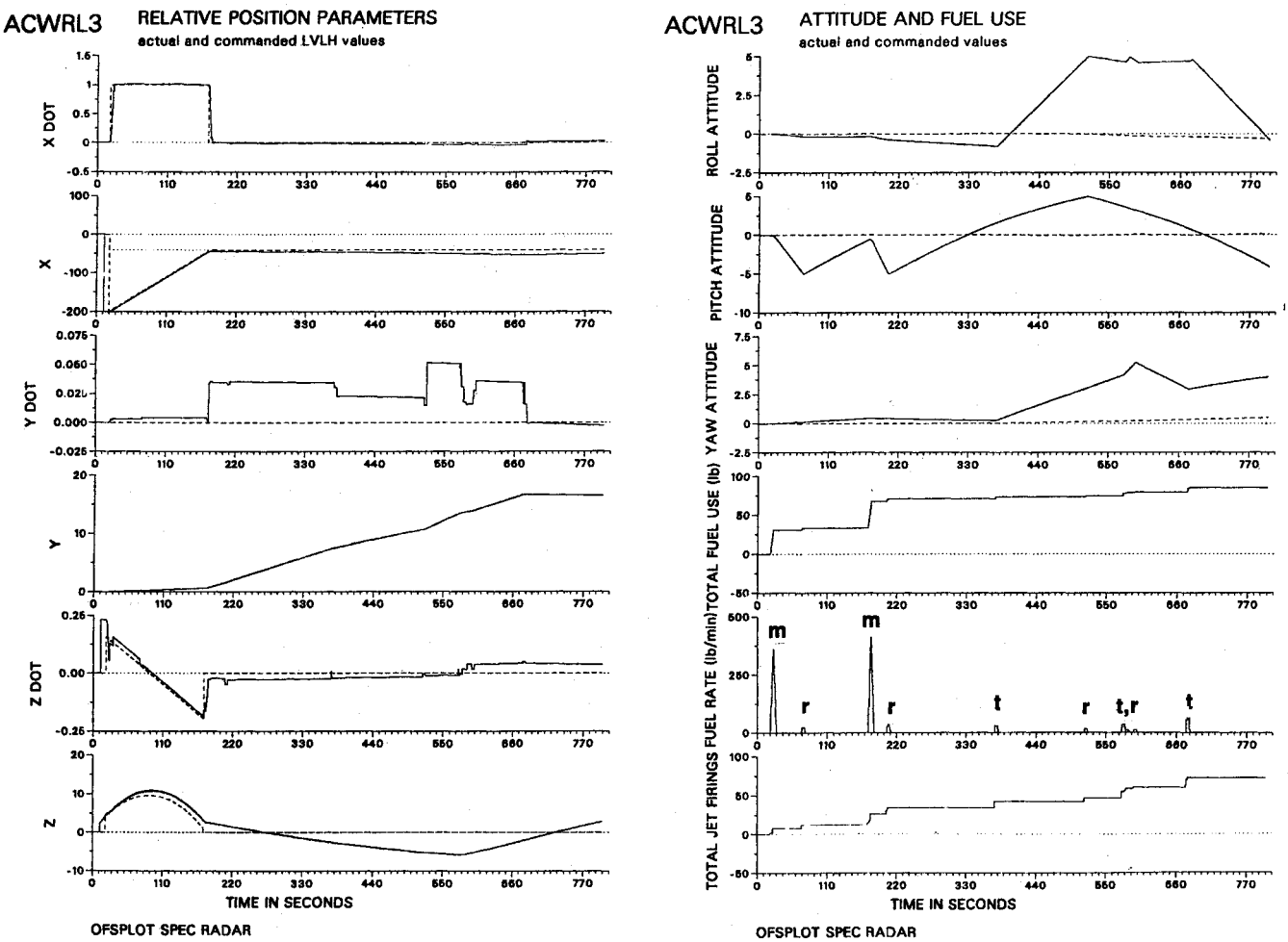

Fig. 7 V-bar maneuver example.

V. The Δv Loop

Conversion of the LQS LVLH Δv commands to body-axis jet-firing commands is handled in a separate, fast Δv loop (Fig. 2). This procedure directly automates the current manual Δv procedure. It is not the most efficient approach, but does provide reasonable performance while allowing the pilot to easily monitor its operation. It is also compatible with the current jet-selection logic.

The Δv loop is activated when a Δv command exceeds the Δv threshold, which is set to twice the RCS granularity. It is executed at the fastest DAP rate (80 ms cycle time) until the Δv command has been satisfied to the accuracy of the RCS granularity.

The first task of the Δv loop is to transform the LVLH Δv request to body frame. It then computes a body-axis acceleration-command vector a_c, which is used by the standard primary RCS jet-selection logic[13] to turn on jets that provide acceleration in the direction of a_c. The resulting single-cycle Δv is subtracted from the total requested Δv_{body} to create a new Δv_{body} command. The process is repeated each subsequent DAP cycle, until Δv_{body} is driven to zero (plus or minus the jet granularity).

The acceleration command a_c is computed on an axis-by-axis basis, following the standard manual procedure. If Δv_{body} includes a positive $\dot{z}_{body}$ component, the $\dot{x}_{body}$ command is satisfied first, by setting $a_c = (\pm 1, 0, 0)^T$ until the x component of Δv_{body} has been reduced to less than the jet granularity. Then, the $\dot{y}_{body}$ command is satisfied, followed by the $\dot{z}_{body}$ command, repeating in the same order if necessary until $\Delta v_{body} = 0$. If Δv_{body} has a negative $\dot{z}_{body}$ component, the process starts with $a_c = (0, 0, -1)^T$ and continues in the same order as just cited until $\Delta v_{body} = 0$. The different starting points reflect different degrees of Orbiter jet-axis cross coupling: The component with the worst coupling is satisfied first.

Other Δv command loop schemes exist that are more efficient than this axis-by-axis approach. These solve the linear programming problem,[2,6] minimizing total fuel use while satisfying the requested Δv_{body}. These tend to be about 20–30% more fuel-efficient for Δv's of about 1 fps magnitude, which are typical maneuver startup or shutdown values. The difference is less pronounced for the smaller Δv's typical of LQS feedback control. Thus, a more sophisticated jet selection will improve position-maneuver performance more than stationkeeping performance.

VI. Results

Detailed simulation results were computed for many cases, to verify and tune the LQS in the desired performance envelope, with accurate representation of the orbital disturbance environment. The Draper orbital flight simulator (OFS) was used. Disturbance effects modeled included differential aerodynamic drag, Earth oblateness, orbital eccentricity, Orbiter rotational/translational jet coupling, ranging radar, and IMU noise.

The LQS was shown to perform quite well using gains derived following the procedure outlined in Sec. IV. Stationkeeping and position-maneuver performance are very good in the desired envelope of about ± 1000 ft in z, ± 2000 ft in y, and several miles in x, for small relative velocities. This envelope could be expanded quite easily.

In this section, we present results from three example maneuvers. Two of these were computed using the ranging radar and the radar gains just presented. The maneuver deadbands r_{db}

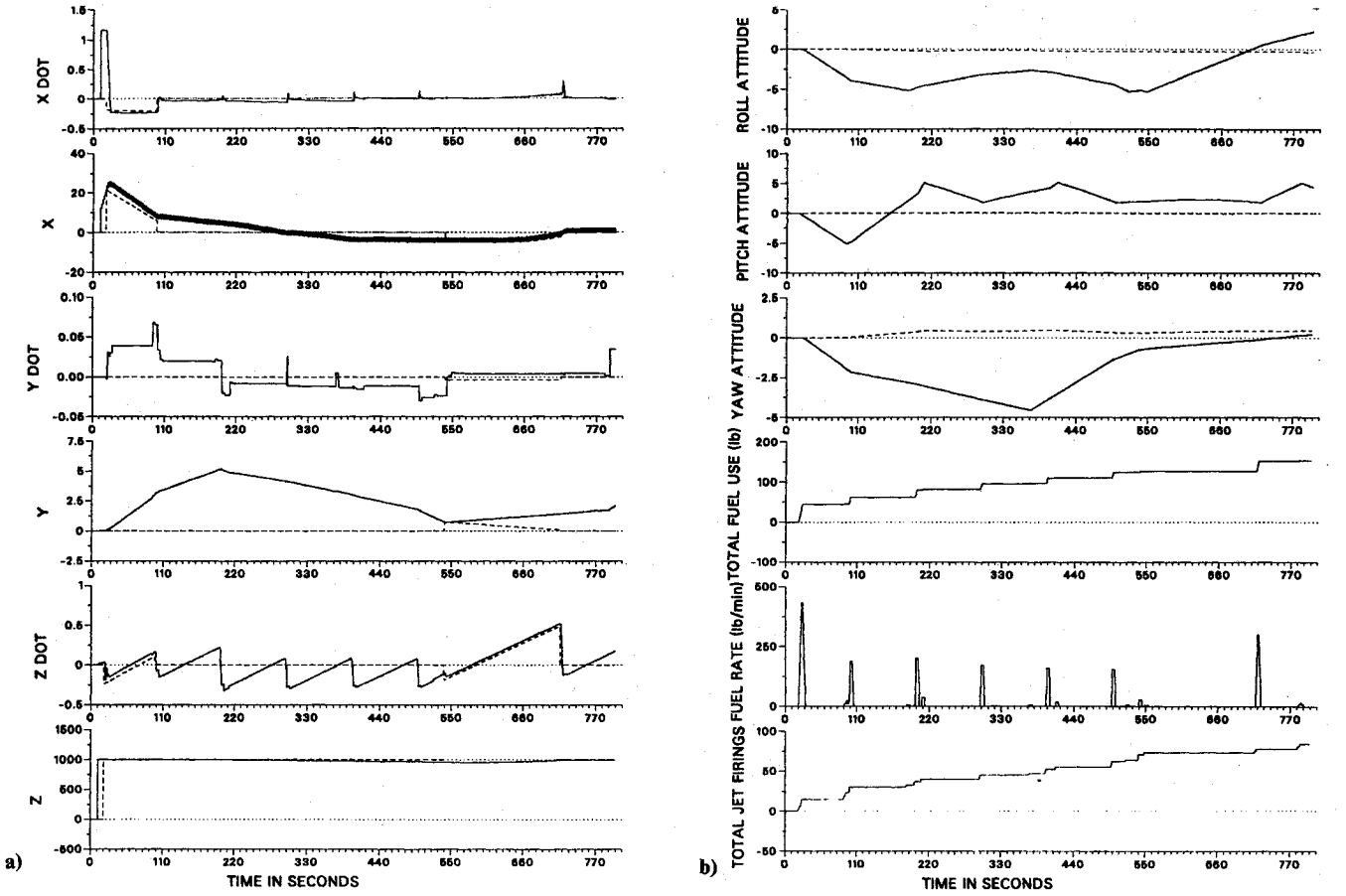

Fig. 8 45 deg approach example; OFSPLOT spec radar: a) ACWRL5 relative position parameters—actual and commanded LVLH values, and b) ACWRL5 attitude and fuel use—actual and commanded values.

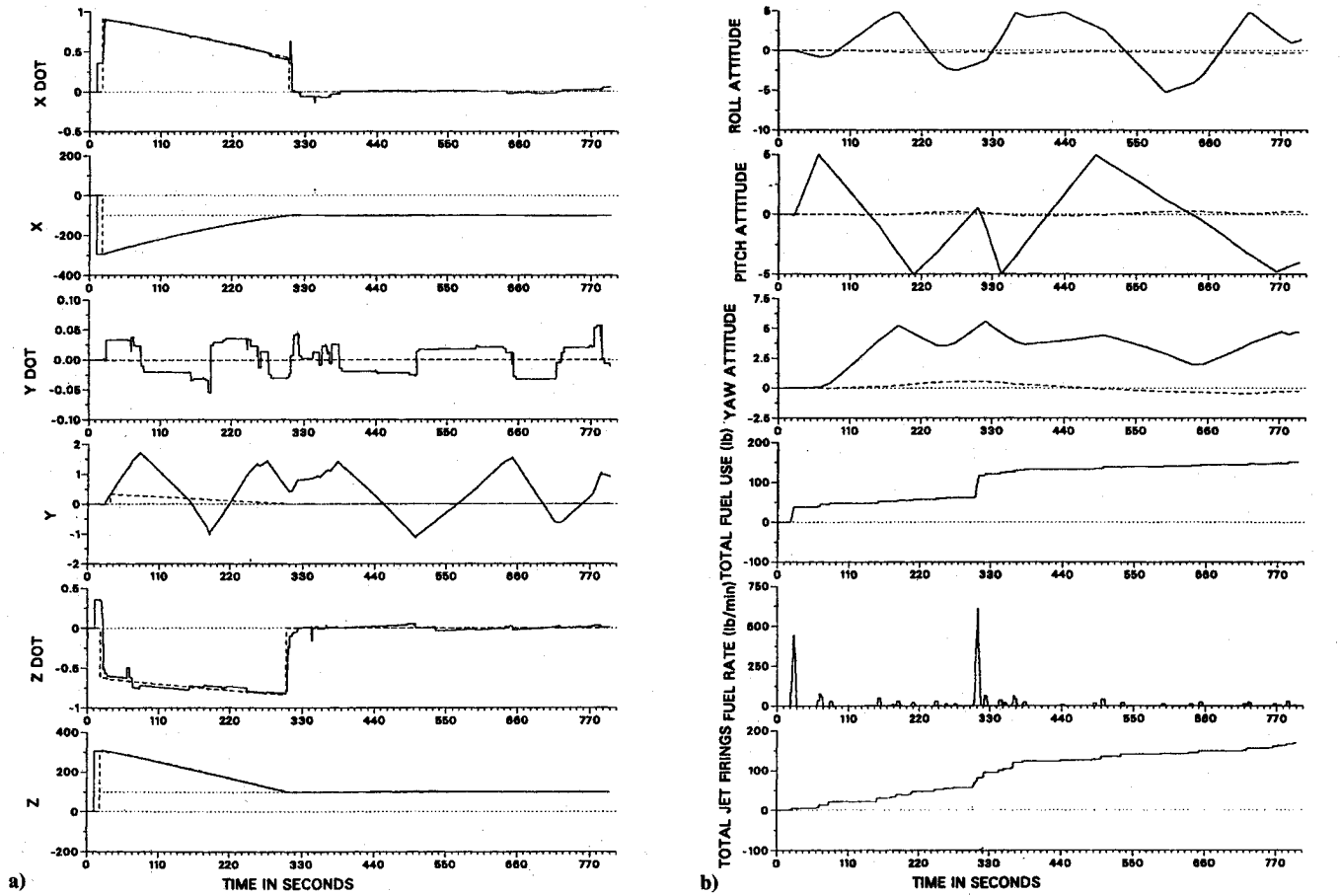

Fig. 9 R-bar stationkeeping example; OFSPLOT laser ranger: a) ACWPH4 relative position parameters—actual and commanded LVLH values, and b) ACWPH4 attitude and fuel use—actual and commanded values.

and v_{db} were 40 ft and 0.5 fps, respectively. The third example uses the laser ranger and laser gains from Sec. III; the deadbands used were $r_{db} = 4$ ft and $v_{db} = 0.4$ fps. The maneuver velocity in all three examples was $v_m = 1.0$ fps, and all used the standard primary RCS jet-selection logic.

Example 1 shows a "v-bar" (X_{LVLH}) approach to a target satellite, followed by 10 min of formationkeeping. The run started at 0 s, with the Orbiter 200 ft behind the satellite with an initial z_{LVLH} velocity error of 0.24 fps (actual position and velocity are shown on Fig. 7). The LQS was turned on at $t = 20$ s with a commanded position 40 ft behind the satellite. The LQS immediately computed a two-burn maneuver to achieve this position. The large firings shown in the fuel plots (Fig. 7) are the maneuver start and stop burns. Of the smaller firings, some are rotational deadband firings (marked with an *r*) and three are LQS feedback firings (marked with a *t*). The latter occur, on the average, once every 200 s, or about every other feedback cycle. Position errors remain within the desired 20-ft deadband.

The second example (Fig. 8) is a more demanding, nonequilibrium point stationkeeping example. Here, the Orbiter is located 1000 ft down the "r bar" (z_{LVLH}), with a large (1.2 fps) x velocity error. The commanded position is the same as the initial position; however, the velocity error exceeds v_{db} and so a small maneuver is done in the first 100 s. The substantial steady-state disturbance due to orbital dynamics is opposed by large feed-forward firings in z. A second maneuver (at 540 s) occurs when the error in z reaches r_{db} (40 ft). Other errors are all within 20 ft. The large fuel usage in this case is purely a reflection of the large steady-state orbital disturbance.

The final example shows a "45 deg" approach (from behind and below) to the target satellite using the laser ranger. The improved accuracy of this instrument allows precision stationkeeping of about 2 ft per axis, even with large control granularity effects, as shown in Fig. 9. Jet activity is of relatively high frequency due to the tight control and short (15 s) sample time.

VII. Conclusion

This paper has presented a linear-quadratic stationkeeping controller, proposed as a software update to the Space Shuttle Orbiter digital autopilot. The controller provides automatic position-hold and position-maneuver capability using existing Orbiter sensors and RCS jets. Discrete-time, linear-quadratic techniques were used to define feedback gains and sampling times that give desirable performance in the presence of sensor noise, control granularity, and plant disturbances. Large errors are corrected using feed-forward control to command either minimum-fuel or straight-line maneuvers. Feed-forward control also is used to cancel steady-state disturbances at nonequilibrium setpoints.

Detailed simulation results show that the proposed controller can provide excellent performance. This performance would improve further if the current radar was augmented with a laser ranger, or if the current jet-selection logic was enhanced to allow use of both the primary and vernier jets for translational control.

The cost of implementing the proposed controller is a function partly of the amount of added computer code. We estimate the total increase to be less than 2000 words in the Orbiter onboard computers. Computer CPU time impact is quite low. Cockpit displays will require modification to support the new capability.

Acknowledgments

The authors wish to thank P. Kachmar, E. Bergmann, D. Sargent, P. Hattis, and B. Persson of the Draper Laboratory, and R. Vassar of Lockheed Palo Alto Research Laboratory for their contributions.

This paper was prepared at the Charles Stark Draper Laboratory under NASA Contract NAS9-16023. Publication of this report does not constitute approval by NASA of findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

References

[1]*Rendezvous/Proximity Operations Workbook*, Flight Operations Directorate, NASA Johnson Space Center, Houston, TX, RNDZ 2102, Feb. 1985.

[2]Bergmann, E. V., "Enhanced Proximity Operations for Space Station," presentation given at NASA Johnson Space Center, Houston, TX, June 1985.

[3]Bryson, A. E., *Stability and Control of Flight Vehicles*, (book draft), Stanford Univ., Stanford, CA, Sept. 1985.

[4]Vassar, R. H. and Sherwood, R. B., "Formationkeeping for a Pair of Satellites in a Circular Orbit," *Journal of Guidance, Control and Dynamics*, Vol. 8, March–April, 1985, pp. 235–242.

[5]Persson, B. A., "Status of OEX Autopilot Postflight Analysis Mission 61B," C. S. Draper Lab., Cambridge, MA, Memo OEX-86-1, March 1986.

[6]Redding, D. C., Persson, B. A., and Bergmann, E. V., "Combined Solution of Spacecraft Rotational and Translational Maneuvers," *Proceedings of the 1986 AIAA Guidance, Navigation and Control Conference*, AIAA, New York, 1986, pp. 441–451.

[7]Gustafson, D. E. and Kriegsman, B. A., "Station-Keeping Guidance," C. S. Draper Lab., Cambridge, MA, Rept. NAS9-10268, March 1972.

[8]Marec, J. P., *Optimal Space Trajectories*, Elsevier, New York, 1979.

[9]Adams, N. J., Redding, D. C., and Cox, K., "Linear-Quadratic Stationkeeping on Travelling Ellipses," American Astronautical Society Paper AAS-87-003, Feb. 1987.

[10]Chen, G., "A Linear Quadratic Gaussian with Loop Transfer Recovery Proximity Operations Autopilot for Spacecraft," M. Sc. Thesis, Dept. of Aeronautics and Astronautics, Massachusetts Inst. of Technology, Cambridge, MA, Sept. 1987.

[11]Franklin, G. F. and Powell, J. D., *Digital Control of Dynamic Systems*, Addison-Wesley, Reading, MA, 1980.

[12]Bryson, A. E. and Ho, Y. C., *Applied Optimal Control*, Hemisphere, New York, 1975.

[13]"Space Shuttle Operational Functional Subsystems Software Requirements Level C, Part C," Rockwell International, Downey, CA, Rept. STS83-0009, Change PCN3, June 1984.