

(Preprint) AAS 11-XXX

MULTI-MANEUVER CLOHESSY-WILTSHIRE TARGETING

David P. Dannemiller*

Orbital rendezvous involves execution of a sequence of maneuvers by a chaser vehicle to bring the chaser to a desired state relative to a target vehicle while meeting intermediate and final relative constraints. Intermediate and final relative constraints are necessary to meet a multitude of requirements such as to control approach direction, ensure relative position is adequate for operation of space-to-space communication systems and relative sensors, provide fail-safe trajectory features, and provide contingency hold points. The effect of maneuvers on constraints is often coupled, so the maneuvers must be solved for as a set. For example, maneuvers that affect orbital energy change both the chaser's height and downrange position relative to the target vehicle. Rendezvous designers use experience and rules-of-thumb to design a sequence of maneuvers and constraints.

A non-iterative method is presented for targeting a rendezvous scenario that includes a sequence of maneuvers and relative constraints. This method is referred to as Multi-Maneuver Clohessy-Wiltshire Targeting (MM_CW_TGT). When a single maneuver is targeted to a single relative position, the classic CW targeting solution is obtained.

The MM_CW_TGT method involves manipulation of the CW state transition matrix to form a linear system. As a starting point for forming the algorithm, the effects of a series of impulsive maneuvers on the state are derived. Simple and moderately complex examples are used to demonstrate the pattern of the resulting linear system. The general form of the pattern results in an algorithm for formation of the linear system. The resulting linear system relates the effect of maneuver components and initial conditions on relative constraints specified by the rendezvous designer. Solution of the linear system includes the straight-forward inverse of a square matrix. Inversion of the square matrix is assured if the designer poses a controllable scenario - a scenario where the constraints can be met by the sequence of maneuvers. Matrices in the linear system are dependent on selection of maneuvers and constraints by the designer, but the matrices are independent of the chaser's initial conditions. For scenarios where the sequence of maneuvers and constraints are fixed, the linear system can be formed and the square matrix inverted prior to real-time operations.

Example solutions are presented for several rendezvous scenarios to illustrate the utility of the method. The MM_CW_TGT method has been used during the preliminary design of rendezvous scenarios and is expected to be useful for iterative methods in the generation of an initial guess and corrections.

*Aerospace Technologist, Flight Dynamics Division, DM34, NASA-Johnson Space Center, 2101 NASA Parkway, Houston, Texas, 77058, David.P.Dannemiller@nasa.gov.

MULTI-MANEUVER CLOHESSY-WILTSHIRE TARGETING

David P. Dannemiller*

A non-iterative method is presented for targeting a rendezvous scenario that includes a sequence of maneuvers and relative constraints. This method is referred to as Multi-Maneuver Clohessy-Wiltshire Targeting (MM_CW_TGT). When a single maneuver is targeted to a single relative position, the classic CW targeting solution is obtained. The MM_CW_TGT method involves manipulation of the CW state transition matrix to form a linear system. Solution of the linear system includes the straight-forward inverse of a square matrix. Example solutions are presented for several rendezvous scenarios to illustrate the utility of the method.

INTRODUCTION

Orbital rendezvous involves execution of a sequence of maneuvers by a chaser vehicle to bring the chaser to a desired state relative to a target vehicle while meeting intermediate and final relative constraints. Intermediate and final relative constraints are necessary to meet a multitude of requirements such as to control approach direction, ensure relative position is adequate for operation of space-to-space communication systems and relative sensors, provide fail-safe trajectory features, and provide contingency hold points. The effect of maneuvers on constraints is often coupled, so the maneuvers must be solved for as a set. For example, maneuvers that affect orbital energy change both the chaser's height and downrange position relative to the target vehicle. Rendezvous designers use experience and rules-of-thumb to design a sequence of maneuvers and constraints.

A non-iterative method is presented for targeting a rendezvous scenario that includes a sequence of maneuvers and relative constraints. This method is referred to as Multi-Maneuver Clohessy-Wiltshire Targeting (MM_CW_TGT). When a single maneuver is targeted to a single relative position, the classic CW targeting solution is obtained.¹

As a starting point for forming the MM_CW_TGT algorithm, the effects of a series of impulsive maneuvers on the state are derived. The MM_CW_TGT method involves manipulation of the CW state transition matrix to form a linear system. Simple and moderately complex examples are used to demonstrate the pattern of the resulting linear system. The general form of the pattern leads to an algorithm for formation of the linear system. The resulting linear system relates the effect of maneuver components and initial conditions on relative constraints specified by the rendezvous designer. Solution of the linear system includes the straight-forward inverse of a square matrix. Inversion of the square matrix is assured if the designer poses a controllable scenario – a scenario where the constraints can be met by the sequence of maneuvers.

*Aerospace Technologist, Flight Dynamics Division, DM34, NASA-Johnson Space Center, 2101 NASA Parkway, Houston, Texas, 77058, David.P.Dannemiller@nasa.gov.

A numerical example is presented to illustrate the utility of the method and to provide comparison data.

NOTATION

Vectors are denoted by an over arrow (e.g. \vec{r}). Subscripts denote time or event information. For example, \vec{r}_0 is the value of \vec{r} at time t_0 .

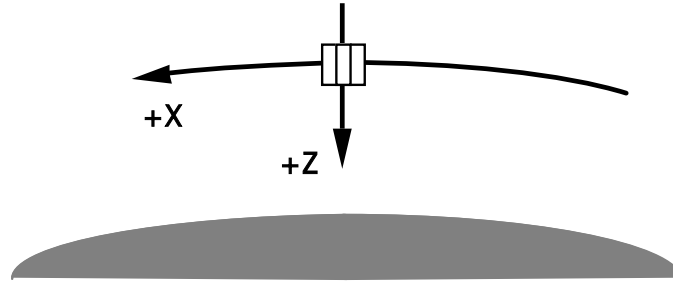


Figure 1. Local Vertical Curvilinear Coordinate System

State vectors are in Local Vertical Curvilinear (LVC) coordinates (axes orientation is shown in Figure 1) and are represented as:

$$\vec{s} = \begin{bmatrix} \vec{r} \\ \vec{v} \end{bmatrix}$$

where \vec{r} is the position vector and \vec{v} is the velocity vector.

$$\vec{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \vec{v} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}$$

The 6x6 Clohessy-Wiltshire (CW) transition matrix is denoted by C_m^n where the subscript denotes the initial time and superscript denotes the final time, t_m and t_n respectively in this case.

Impulsive maneuvers are instantaneous changes in velocity at a specified time. To remove ambiguity, it is necessary to specify pre or post-maneuver when referring to velocity and state vectors. Subscripts - and + are used to refer to pre and post-maneuver quantities respectively.

$$\Delta \vec{v}_n = \vec{v}_{n+} - \vec{v}_{n-} = \begin{bmatrix} \Delta \dot{x}_n \\ \Delta \dot{y}_n \\ \Delta \dot{z}_n \end{bmatrix}$$

While not strictly required, subscripts - and + are also applied to times to differentiate an interest in pre or post-maneuver quantities (e.g. t_{n-} , t_{n+}).

EFFECT OF A SERIES OF IMPULSIVE MANEUVERS

As a starting point for forming the MM.CW.TGT algorithm, the effects of a series of impulsive maneuvers on the state are derived.

Given an initial state \vec{s}_{0-} at time t_{0-} and impulsive maneuvers $\Delta\vec{v}_0$, $\Delta\vec{v}_1$, and $\Delta\vec{v}_2$ at times t_0 , t_1 , and t_2 , determine the state at time t_{2+} .

$$\begin{aligned}
\vec{s}_{0+} &= \vec{s}_{0-} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} \\
\vec{s}_{1-} &= C_0^1 \cdot \vec{s}_{0+} \\
&= C_0^1 \cdot \vec{s}_{0-} + C_0^1 \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} \\
\vec{s}_{1+} &= \vec{s}_{1-} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_1 \end{bmatrix} \\
&= C_0^1 \cdot \vec{s}_{0-} + C_0^1 \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_1 \end{bmatrix} \\
\vec{s}_{2+} &= C_1^2 \cdot \vec{s}_{1+} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_2 \end{bmatrix} \\
&= C_1^2 \cdot \left(C_0^1 \cdot \vec{s}_{0-} + C_0^1 \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_1 \end{bmatrix} \right) + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_2 \end{bmatrix} \\
&= C_0^2 \cdot \vec{s}_{0-} + C_0^2 \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} + C_1^2 \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_1 \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_2 \end{bmatrix}
\end{aligned}$$

In general there will be impulsive maneuvers at times $t_0, t_1, t_2, \dots, t_n$. The general form for the effect of this series of n impulsive maneuvers is obtained by extrapolating the pattern above.

$$\vec{s}_{n+} = C_0^n \cdot \vec{s}_{0-} + C_0^n \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} + C_1^n \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_1 \end{bmatrix} + C_2^n \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_2 \end{bmatrix} + \dots + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_n \end{bmatrix} \quad (1)$$

The MM_CW_TGT method involves manipulation of the CW state transition matrix to form a linear system. Two examples are used to demonstrate the pattern of the resulting linear system, a simple targeting example and a more complex targeting example.

SIMPLE TARGETING EXAMPLE

Assume a pair of horizontal maneuvers transfer a chaser vehicle from a stable location on the x axis at \vec{s}_{0-} to another stable location \vec{s}_{f+} in an integral number of orbits. Figure 2 shows the desired trajectory.

For this example, Eq. (1) takes the following form.

$$\vec{s}_{f+} = C_0^f \cdot \vec{s}_{0-} + C_0^f \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_f \end{bmatrix} \quad (2)$$

Restructure this equation in 2 steps. The first step takes advantage of knowledge of the selected maneuver components. Since $\Delta\vec{v}_0$ and $\Delta\vec{v}_f$ are specified as horizontal, they take the form

$$\Delta\vec{v}_0 = \begin{bmatrix} \Delta\dot{x}_0 \\ 0 \\ 0 \end{bmatrix}, \Delta\vec{v}_f = \begin{bmatrix} \Delta\dot{x}_f \\ 0 \\ 0 \end{bmatrix} \quad (3)$$

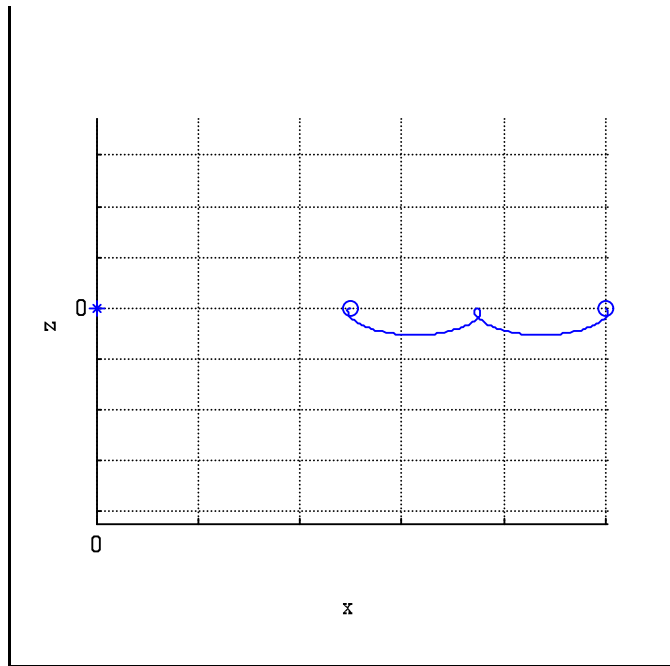


Figure 2. Simple Targeting Example

Substitute Eq. (3) into Eq. (2).

$$\vec{s}_{f+} = C_0^f \cdot \vec{s}_{0-} + \begin{bmatrix} * & * & * & C_0^f(1,4) & * & * \\ * & * & * & C_0^f(2,4) & * & * \\ * & * & * & C_0^f(3,4) & * & * \\ * & * & * & C_0^f(5,4) & * & * \\ * & * & * & C_0^f(5,4) & * & * \\ * & * & * & C_0^f(6,4) & * & * \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta \dot{x}_0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta \dot{x}_f \\ 0 \\ 0 \end{bmatrix}$$

One of the CW transition matrices is expanded. Most of the terms are “*”, denoting that their specific values do not matter. Because of the form of $\Delta \vec{v}_0$, only one column is relevant. The identity

matrix is added in front of the $\Delta\vec{v}_f$ term for symmetry. Simplifying,

$$\begin{aligned}\vec{s}_{f+} &= C_0^f \cdot \vec{s}_{0-} + \begin{bmatrix} C_0^f(1,4) \\ C_0^f(2,4) \\ C_0^f(3,4) \\ C_0^f(4,4) \\ C_0^f(5,4) \\ C_0^f(6,4) \end{bmatrix} \cdot \Delta\dot{x}_0 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \cdot \Delta\dot{x}_f \\ \vec{s}_{f+} &= C_0^f \cdot \vec{s}_{0-} + \begin{bmatrix} C_0^f(1,4) & 0 \\ C_0^f(2,4) & 0 \\ C_0^f(3,4) & 0 \\ C_0^f(4,4) & 1 \\ C_0^f(5,4) & 0 \\ C_0^f(6,4) & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta\dot{x}_0 \\ \Delta\dot{x}_f \end{bmatrix}\end{aligned}\quad (4)$$

The second restructuring step takes advantage of the selected constraints. For this example the final state is specified as a stable location on the x axis (i.e. $\dot{x}_f = 0$). This implies that \vec{s}_{f+} takes the form

$$\vec{s}_{f+} = \begin{bmatrix} x_f \\ * \\ * \\ 0 \\ * \\ * \end{bmatrix}\quad (5)$$

Note that y_f , z_f , \dot{y}_f , and \dot{z}_f are not explicitly constrained. Given the example setup (i.e. initial state is stable on the x axis, and transfer time is an integral number of orbits), by definition z_f and \dot{z}_f are both zero. Since no maneuver components are specified in the y direction, neither y_f nor \dot{y}_f can be used as constraints.

Substitute Eq. (5) into Eq. (4), and then remove all but the x_f and \dot{x}_f rows.

$$\begin{aligned}\begin{bmatrix} x_f \\ 0 \end{bmatrix} &= \begin{bmatrix} C_0^f(1,1) & C_0^f(1,2) & C_0^f(1,3) & C_0^f(1,4) & C_0^f(1,5) & C_0^f(1,6) \\ C_0^f(4,1) & C_0^f(4,2) & C_0^f(4,3) & C_0^f(4,4) & C_0^f(4,5) & C_0^f(4,6) \end{bmatrix} \cdot \vec{s}_{0-} \\ &+ \begin{bmatrix} C_0^f(1,4) & 0 \\ C_0^f(4,4) & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta\dot{x}_0 \\ \Delta\dot{x}_f \end{bmatrix}\end{aligned}\quad (6)$$

Eq. (6) is a system of linear equations with 2 equations and 2 unknowns – the desired maneuver components $\Delta\dot{x}_0$ and $\Delta\dot{x}_f$. Use shorthand notation to compact Eq. (6).

$$\vec{s}_{constraint} = A \cdot \vec{s}_{0-} + B \cdot \Delta\vec{v}\quad (7)$$

The n-vector $\vec{s}_{constraint}$ contains the selected constraints, where n is the number of constraints. The nx6 matrix A is the effect of \vec{s}_{0-} on the constraints. The n-vector $\Delta\vec{v}$ contains the selected

maneuver components. The $n \times n$ matrix B is the effect of $\Delta \vec{v}$ on the constraints. Rearrange Eq. (6) to solve for $\Delta \vec{v}$.

$$\Delta \vec{v} = B^{-1} \cdot (\vec{s}_{constraint} - A \cdot \vec{s}_{0-}) \quad (8)$$

Applying the maneuver components $\Delta \dot{x}_0$ and $\Delta \dot{x}_f$, at the specified times, will satisfy the constraints.

To summarize, a sequence of maneuvers (times and components) and constraints (times and components) were designed. A system of linear equations was structured in 2 steps. The first step eliminated terms associated with zero maneuver components. The second step formed the $\vec{s}_{constraint}$ vector, and the A and B matrices by keeping only the terms associated with constraints. Since the number of maneuver components is equal to the number of constraints, the resulting B matrix is square. Inversion of the B matrix is assured if the designer poses a controllable scenario – a scenario where the constraints can be met by the sequence of maneuvers.

MODERATELY COMPLEX TARGETING EXAMPLE

Using the multi-maneuver CW technique to solve the simple example in the last section is overkill, but provides initial insight into the pattern of the resulting linear system. More insight is provided in this section by a moderately complex example.

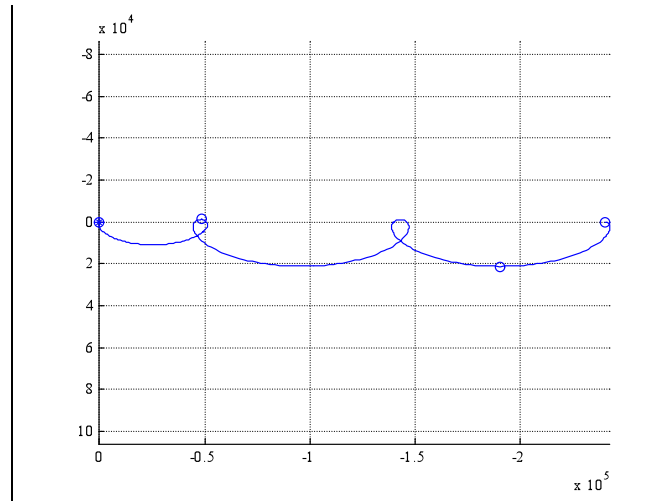


Figure 3. Moderately Complex Targeting Example

For this scenario, a sequence of maneuvers NC, NH, TI (Transition Initiate), TF (Transition Final) transfer a chaser vehicle from an initial state on the x axis at \vec{s}_{0-} , to a desired TI position at \vec{s}_{TI+} , to the final state at \vec{s}_{TF+} . Figure 3 shows the desired trajectory.

The burn names used here have Gemini, Apollo, and Shuttle heritage. “N” in the burn names was originally (circa 1964) a Mission Control Docking Initiation (DKI) rendezvous burn targeting program counter variable. “N” stood for the Nth crossing of the chaser line of apsides where the burn was performed (such as 1 for first apogee, 1.5 for first perigee, 5 for fifth apogee, etc.). The burn type was added using subscripts as 1_C (Catch-up or phasing), 2.5_H (Height), 3_{PC} (Plane Change),

4_{SR} (Slow Rate or coelliptic), and 4.5_{CC} (Corrective Combination). Eventually the apse count and subscripts notation were dropped. The resulting burn names became NC, NH, NPC, NSR, and NCC.

We specify horizontal maneuvers at NC and NH, and 2-axis maneuvers at TI and TF. Typically, TI and TF include maneuver components in the y direction. They are not included in this example in order to keep the problem tractable. The maneuvers take the following form.

$$\Delta\vec{v}_{NC} = \begin{bmatrix} \Delta\dot{x}_{NC} \\ 0 \\ 0 \end{bmatrix}, \Delta\vec{v}_{NH} = \begin{bmatrix} \Delta\dot{x}_{NH} \\ 0 \\ 0 \end{bmatrix}, \Delta\vec{v}_{TI} = \begin{bmatrix} \Delta\dot{x}_{TI} \\ 0 \\ \Delta\dot{z}_{TI} \end{bmatrix}, \Delta\vec{v}_{TF} = \begin{bmatrix} \Delta\dot{x}_{TF} \\ 0 \\ \Delta\dot{z}_{TF} \end{bmatrix}$$

We have constraints at TI and TF.

$$\vec{s}_{TI+} = \begin{bmatrix} x_{TI} \\ * \\ z_{TI} \\ * \\ * \\ * \\ * \end{bmatrix}, \vec{s}_{TF+} = \begin{bmatrix} x_{TF} \\ * \\ z_{TF} \\ \dot{x}_{TF+} \\ * \\ \dot{z}_{TF+} \end{bmatrix}$$

Notice that the number of maneuver components and the number of constraints are both 6, which will result in a square B matrix. Also notice that the scenario is controllable; NC/NH are capable of maneuvering the chaser to the specified TI position x_{TI} and z_{TI} ; TI/TF are capable of achieving the desired final conditions.

Since constraints are specified at 2 times, the effects of maneuvers at both TI and TF must be formulated. First TI.

$$\vec{s}_{TI+} = C_0^{TI} \cdot \vec{s}_{0-} + C_{NC}^{TI} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{NC} \end{bmatrix} + C_{NH}^{TI} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{NH} \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{TI} \end{bmatrix}$$

Next TF.

$$\vec{s}_{TF+} = C_0^{TF} \cdot \vec{s}_{0-} + C_{NC}^{TF} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{NC} \end{bmatrix} + C_{NH}^{TF} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{NH} \end{bmatrix} + C_{TI}^{TF} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{TI} \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{TF} \end{bmatrix}$$

The equations are restructured in 3 steps. The first step takes advantage of knowledge of the

selected maneuver components.

$$\begin{aligned}
\vec{s}_{TI+} &= C_0^{TI} \cdot \vec{s}_{0-} + \begin{bmatrix} C_{NC}^{TI}(1,4) \\ C_{NC}^{TI}(2,4) \\ C_{NC}^{TI}(3,4) \\ C_{NC}^{TI}(4,4) \\ C_{NC}^{TI}(5,4) \\ C_{NC}^{TI}(6,4) \end{bmatrix} \cdot \Delta \dot{x}_{NC} + \begin{bmatrix} C_{NH}^{TI}(1,4) \\ C_{NH}^{TI}(2,4) \\ C_{NH}^{TI}(3,4) \\ C_{NH}^{TI}(4,4) \\ C_{NH}^{TI}(5,4) \\ C_{NH}^{TI}(6,4) \end{bmatrix} \cdot \Delta \dot{x}_{NH} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_{TI} \\ \Delta \dot{z}_{TI} \end{bmatrix} \\
\vec{s}_{TF+} &= C_0^{TF} \cdot \vec{s}_{0-} + \begin{bmatrix} C_{NC}^{TF}(1,4) \\ C_{NC}^{TF}(2,4) \\ C_{NC}^{TF}(3,4) \\ C_{NC}^{TF}(4,4) \\ C_{NC}^{TF}(5,4) \\ C_{NC}^{TF}(6,4) \end{bmatrix} \cdot \Delta \dot{x}_{NC} + \begin{bmatrix} C_{NH}^{TF}(1,4) \\ C_{NH}^{TF}(2,4) \\ C_{NH}^{TF}(3,4) \\ C_{NH}^{TF}(4,4) \\ C_{NH}^{TF}(5,4) \\ C_{NH}^{TF}(6,4) \end{bmatrix} \cdot \Delta \dot{x}_{NH} \\
&+ \begin{bmatrix} C_{TI}^{TF}(1,4) & C_{TI}^{TF}(1,6) \\ C_{TI}^{TF}(2,4) & C_{TI}^{TF}(2,6) \\ C_{TI}^{TF}(3,4) & C_{TI}^{TF}(3,6) \\ C_{TI}^{TF}(4,4) & C_{TI}^{TF}(4,6) \\ C_{TI}^{TF}(5,4) & C_{TI}^{TF}(5,6) \\ C_{TI}^{TF}(6,4) & C_{TI}^{TF}(6,6) \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_{TI} \\ \Delta \dot{z}_{TI} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_{TF} \\ \Delta \dot{z}_{TF} \end{bmatrix}
\end{aligned}$$

The second step in restructuring takes advantage of the selected constraints. Eliminate all rows that are not constraints.

$$\begin{aligned}
\begin{bmatrix} x_{TI} \\ z_{TI} \end{bmatrix} &= \begin{bmatrix} C_0^{TI}(1,1) & C_0^{TI}(1,2) & C_0^{TI}(1,3) & C_0^{TI}(1,4) & C_0^{TI}(1,5) & C_0^{TI}(1,6) \\ C_0^{TI}(3,1) & C_0^{TI}(3,2) & C_0^{TI}(3,3) & C_0^{TI}(3,4) & C_0^{TI}(3,5) & C_0^{TI}(3,6) \end{bmatrix} \cdot \vec{s}_{0-} \\
&+ \begin{bmatrix} C_{NC}^{TI}(1,4) \\ C_{NC}^{TI}(3,4) \end{bmatrix} \cdot \Delta \dot{x}_{NC} + \begin{bmatrix} C_{NH}^{TI}(1,4) \\ C_{NH}^{TI}(3,4) \end{bmatrix} \cdot \Delta \dot{x}_{NH} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_{TI} \\ \Delta \dot{z}_{TI} \end{bmatrix} \\
\begin{bmatrix} x_{TF} \\ z_{TF} \\ \dot{x}_{TF+} \\ \dot{z}_{TF+} \end{bmatrix} &= \begin{bmatrix} C_0^{TF}(1,1) & C_0^{TF}(1,2) & C_0^{TF}(1,3) & C_0^{TF}(1,4) & C_0^{TF}(1,5) & C_0^{TF}(1,6) \\ C_0^{TF}(3,1) & C_0^{TF}(3,2) & C_0^{TF}(3,3) & C_0^{TF}(3,4) & C_0^{TF}(3,5) & C_0^{TF}(3,6) \\ C_0^{TF}(4,1) & C_0^{TF}(4,2) & C_0^{TF}(4,3) & C_0^{TF}(4,4) & C_0^{TF}(4,5) & C_0^{TF}(4,6) \\ C_0^{TF}(6,1) & C_0^{TF}(6,2) & C_0^{TF}(6,3) & C_0^{TF}(6,4) & C_0^{TF}(6,5) & C_0^{TF}(6,6) \end{bmatrix} \cdot \vec{s}_{0-} \\
&+ \begin{bmatrix} C_{NC}^{TF}(1,4) \\ C_{NC}^{TF}(3,4) \\ C_{NC}^{TF}(4,4) \\ C_{NC}^{TF}(6,4) \end{bmatrix} \cdot \Delta \dot{x}_{NC} + \begin{bmatrix} C_{NH}^{TF}(1,4) \\ C_{NH}^{TF}(3,4) \\ C_{NH}^{TF}(4,4) \\ C_{NH}^{TF}(6,4) \end{bmatrix} \cdot \Delta \dot{x}_{NH} \\
&+ \begin{bmatrix} C_{TI}^{TF}(1,4) & C_{TI}^{TF}(1,6) \\ C_{TI}^{TF}(3,4) & C_{TI}^{TF}(3,6) \\ C_{TI}^{TF}(4,4) & C_{TI}^{TF}(4,6) \\ C_{TI}^{TF}(6,4) & C_{TI}^{TF}(6,6) \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_{TI} \\ \Delta \dot{z}_{TI} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_{TF} \\ \Delta \dot{z}_{TF} \end{bmatrix}
\end{aligned}$$

For the final step in restructuring, stack the 2 equations into a single linear system.

$$\begin{bmatrix} x_{TI} \\ z_{TI} \\ x_{TF} \\ z_{TF} \\ \dot{x}_{TF+} \\ \dot{z}_{TF+} \end{bmatrix} = \begin{bmatrix} C_0^{TI}(1,1) & C_0^{TI}(1,2) & C_0^{TI}(1,3) & C_0^{TI}(1,4) & C_0^{TI}(1,5) & C_0^{TI}(1,6) \\ C_0^{TI}(3,1) & C_0^{TI}(3,2) & C_0^{TI}(3,3) & C_0^{TI}(3,4) & C_0^{TI}(3,5) & C_0^{TI}(3,6) \\ C_0^{TF}(1,1) & C_0^{TF}(1,2) & C_0^{TF}(1,3) & C_0^{TF}(1,4) & C_0^{TF}(1,5) & C_0^{TF}(1,6) \\ C_0^{TF}(3,1) & C_0^{TF}(3,2) & C_0^{TF}(3,3) & C_0^{TF}(3,4) & C_0^{TF}(3,5) & C_0^{TF}(3,6) \\ C_0^{TF}(4,1) & C_0^{TF}(4,2) & C_0^{TF}(4,3) & C_0^{TF}(4,4) & C_0^{TF}(4,5) & C_0^{TF}(4,6) \\ C_0^{TF}(6,1) & C_0^{TF}(6,2) & C_0^{TF}(6,3) & C_0^{TF}(6,4) & C_0^{TF}(6,5) & C_0^{TF}(6,6) \end{bmatrix} \cdot \vec{s}_{0-} \\
+ \begin{bmatrix} C_{NC}^{TI}(1,4) & C_{NH}^{TI}(1,4) & 0 & 0 & 0 & 0 \\ C_{NC}^{TI}(3,4) & C_{NH}^{TI}(3,4) & 0 & 0 & 0 & 0 \\ C_{NC}^{TF}(1,4) & C_{NH}^{TF}(1,4) & C_{TI}^{TF}(1,4) & C_{TI}^{TF}(1,6) & 0 & 0 \\ C_{NC}^{TF}(3,4) & C_{NH}^{TF}(3,4) & C_{TI}^{TF}(3,4) & C_{TI}^{TF}(3,6) & 0 & 0 \\ C_{NC}^{TF}(4,4) & C_{NH}^{TF}(4,4) & C_{TI}^{TF}(4,4) & C_{TI}^{TF}(4,6) & 1 & 0 \\ C_{NC}^{TF}(6,4) & C_{NH}^{TF}(6,4) & C_{TI}^{TF}(6,4) & C_{TI}^{TF}(6,6) & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_{NC} \\ \Delta \dot{x}_{NH} \\ \Delta \dot{x}_{TI} \\ \Delta \dot{z}_{TI} \\ \Delta \dot{x}_{TF} \\ \Delta \dot{z}_{TF} \end{bmatrix} \quad (9)$$

Notice that Eq. (9) has the same general form as Eq. (7). We solve for $\Delta \vec{v}$ in the same manner as Eq. (8).

GENERAL PATTERN

We use Eq. (9) to find the general pattern of the Multi-Maneuver CW Targeting algorithm.

1. The selected maneuver components and constraints are used to form a system of linear equations of the form $\vec{s}_{constraint} = A \cdot \vec{s}_{0-} + B \cdot \Delta \vec{v}$.
2. The number of maneuver components must be equal to the number of constraints, which is denoted as n .
3. $\vec{s}_{constraint}$ is an n -vector that contains the selected constraints.
4. $\Delta \vec{v}$ is an n -vector that contains the selected maneuver components.
5. A is an $n \times 6$ matrix that contains the effects of \vec{s}_{0-} on the constraints. Each row of A contains a row from the CW transition matrix, the specific row being that associated with the selected constraint, and the transfer time equal to $t_{constraint} - t_0$.
6. B is an $n \times n$ matrix that contains the effects of the maneuver components on the constraints. Each row of B contains terms from the CW transition matrix, the specific row being that associated with the constraint, the specific column being that associated with the maneuver component, and the transfer time equal to $t_{constraint} - t_{maneuver}$. If the constraint time is after the maneuver time, the term is 0 (later maneuvers cannot effect earlier constraints).
7. B must be invertible. This is assured if the designer poses a controllable scenario – a scenario where the the constraints can be met by the sequence of maneuvers.
8. The linear system is solved for the unknown maneuver components.

$$\Delta \vec{v} = B^{-1} \cdot (\vec{s}_{constraint} - A \cdot \vec{s}_{0-})$$

ACCELERATION CONSTRAINT

Many trajectory scenarios can be solved with the position and velocity constraints discussed above. Two other useful constraint types are derived in this section and next.

Co-elliptic segments are useful in many rendezvous scenarios. Targeting co-elliptic conditions with only position and velocity constraints is possible, but awkward. When co-elliptic, $z = const$, $\dot{z} = 0$, and $\ddot{z} = 0$, so the addition of an acceleration constraint makes co-elliptic targeting straight forward.

The CW differential equations for coasting flight are:

$$\begin{aligned}\ddot{x} &= 2 \cdot \omega \cdot \dot{z} \\ \ddot{y} &= -\omega^2 \cdot y \\ \ddot{z} &= 3 \cdot \omega^2 \cdot z - 2 \cdot \omega \cdot \dot{x}\end{aligned}$$

Put these equations in matrix form.

$$\begin{aligned}\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 2 \cdot \omega \\ 0 & -\omega^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 \cdot \omega^2 & -2 \cdot \omega & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \vec{r} \\ \vec{v} \end{bmatrix} \\ \ddot{\vec{a}} &= PVA \cdot \begin{bmatrix} \vec{r} \\ \vec{v} \end{bmatrix}\end{aligned}\quad (10)$$

We refer to the 3x6 matrix in Eq. (10) as PVA since it converts a position/velocity vector to an acceleration vector. Notice that PVA is a function only of ω ; it is time independent. Pre-multiply Eq. (1) by PVA .

$$\ddot{\vec{a}}_{n+1} = PVA \cdot \vec{s}_{n+} = PVA \cdot C_0^n \cdot \vec{s}_{0-} + PVA \cdot C_0^n \cdot \begin{bmatrix} \vec{0} \\ \Delta \vec{v}_0 \end{bmatrix} + \dots + PVA \cdot \begin{bmatrix} \vec{0} \\ \Delta \vec{v}_n \end{bmatrix}\quad (11)$$

For scenarios where an acceleration constraint is desired, form the constraint using Eq. (11) and adjoin it to the linear system.

ELEVATION ANGLE CONSTRAINT

Another useful constraint is elevation angle. Figure 4 shows the convention for the in-plane elevation angle.

Specifying an elevation angle constraint imposes a constraint on the relationship between x and z . If θ is the desired elevation angle, then

$$\tan \theta = -\frac{z_\theta}{x_\theta}\quad (12)$$

Form the following.

$$x_\theta = \vec{f}_x \cdot \vec{s}_0 + \vec{g}_x \cdot \Delta \vec{v}\quad (13)$$

$$z_\theta = \vec{f}_z \cdot \vec{s}_0 + \vec{g}_z \cdot \Delta \vec{v}\quad (14)$$

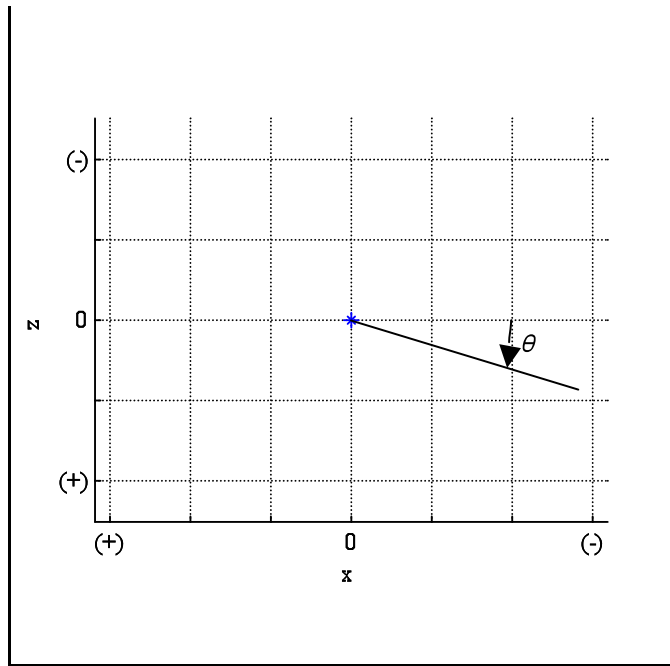


Figure 4. Elevation Angle Convention

Here \vec{f}_x and \vec{g}_x are row vectors that represent the effect of \vec{s}_0 and $\Delta\vec{v}$ on x_θ . Likewise for \vec{f}_z and \vec{g}_z on z_θ . Substitute Eqs. (13) and (14) into Eq. (12) and restructure.

$$0 = \left(\vec{f}_z + \tan \theta \cdot \vec{f}_x \right) \cdot \vec{s}_0 + \left(\vec{g}_z + \tan \theta \cdot \vec{g}_x \right) \cdot \Delta\vec{v} \quad (15)$$

Note that Eq. (15) has a form similar to previous constraints. For scenarios where an elevation angle constraint is desired, form the constraint using Eq. (15) and adjoin it to the linear system.

NC, NH, NSR, TPI, TPF NUMERICAL EXAMPLE

This section provides a numerical example to illustrate the utility of the method and to provide comparison data. Figures 5 and 6 illustrate the desired relative motion.

Initial conditions for the scenario are near the x axis with some out-of-plane position.

$$t_0 = 1000 \text{ sec}, \quad \vec{s}_0 = \begin{bmatrix} -1,000,000 \\ 2000 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ ft}$$

The target orbit period is 90 minutes.

$$w = \frac{2 \cdot \pi \text{ rad}}{5400 \text{ sec}}$$

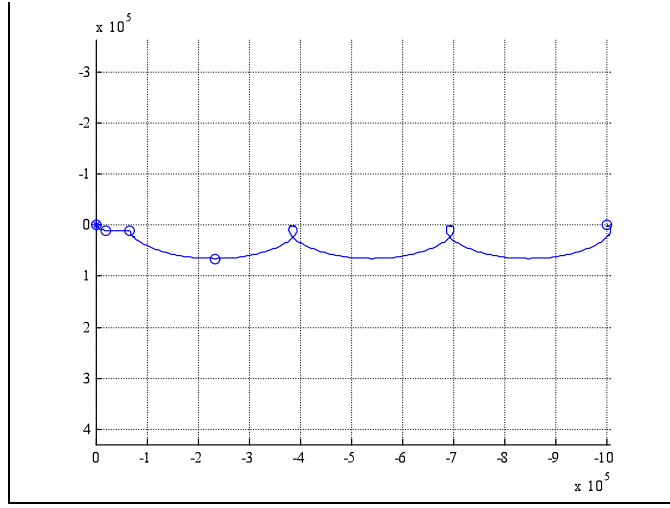


Figure 5. Numerical Example

Event times are as follows.

$$\begin{aligned}
 t_0 &= 1000 \text{ sec} \\
 t_{NC} &= t_0 \\
 t_{NH} &= t_{NC} + 2.5 \cdot \frac{\pi}{w} \quad (\text{i.e. } 2.5 \text{ orbits from } NC) \\
 t_{NSR} &= t_{NH} + .4 \cdot \frac{\pi}{w} * \\
 t_{TPI} &= t_{NSR} + .5 \cdot \frac{\pi}{w} \\
 t_{TPF} &= t_{TPI} + \frac{130}{360} \cdot \frac{\pi}{w} \quad (\text{i.e. } 130 \text{ degree transfer from } TPI)
 \end{aligned}$$

Maneuvers NC and NH are horizontal. NSR has x and z maneuver components. TPI and TPF are 3-axis – x , y , and z maneuver components.

$$\begin{aligned}
 \Delta \vec{v}_{NC} &= \begin{bmatrix} \Delta \dot{x}_{NC} \\ 0 \\ 0 \end{bmatrix}, \quad \Delta \vec{v}_{NH} = \begin{bmatrix} \Delta \dot{x}_{NH} \\ 0 \\ 0 \end{bmatrix}, \quad \Delta \vec{v}_{NSR} = \begin{bmatrix} \Delta \dot{x}_{NSR} \\ 0 \\ \Delta \dot{z}_{NSR} \end{bmatrix} \\
 \Delta \vec{v}_{TPI} &= \begin{bmatrix} \Delta \dot{x}_{TPI} \\ \Delta \dot{y}_{TPI} \\ \Delta \dot{z}_{TPI} \end{bmatrix}, \quad \Delta \vec{v}_{TPF} = \begin{bmatrix} \Delta \dot{x}_{TPF} \\ \Delta \dot{y}_{TPF} \\ \Delta \dot{z}_{TPF} \end{bmatrix}
 \end{aligned}$$

Constraints are imposed at 3 event times – post-NSR co-elliptic, TPI elevation angle, and post-TPF state.

$$\begin{aligned}
 z_{NSR} &= 10,000 \text{ ft}, \quad \dot{z}_{NSR+} = 0, \quad \ddot{z}_{NSR+} = 0 \\
 \theta_{TPI} &= .5 \text{ rad} \\
 \vec{s}_{TPF+} &= \vec{0}
 \end{aligned}$$

*Using a transfer of .5 orbits instead of .4 would result in lower $\Delta \vec{v}$. This example uses .4 to better illustrate the NSR maneuver.

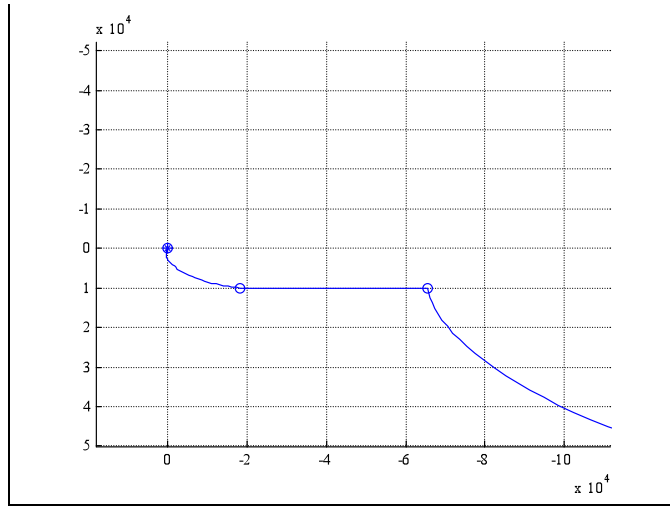


Figure 6. Numerical Example (cont.)

We have everything required to form the linear system, Eq. (7). Doing so results in the following quantities.

$$\vec{s}_{constraint} = \begin{bmatrix} z_{NSR} \\ \dot{z}_{NSR+} \\ \ddot{z}_{NSR+} \\ \text{row associated with } \theta_{TPI} \\ x_{TPF+} \\ y_{TPF+} \\ z_{TPF+} \\ \dot{x}_{TPF+} \\ \dot{y}_{TPF+} \\ \dot{z}_{TPF+} \end{bmatrix}$$

$$\Delta \vec{v} = \begin{bmatrix} \Delta \dot{x}_{NC} \\ \Delta \dot{x}_{NH} \\ \Delta \dot{x}_{NSR} \\ \Delta \dot{z}_{NSR} \\ \Delta \dot{x}_{TPI} \\ \Delta \dot{y}_{TPI} \\ \Delta \dot{z}_{TPI} \\ \Delta \dot{x}_{TPF} \\ \Delta \dot{y}_{TPF} \\ \Delta \dot{z}_{TPF} \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 0 & 0.0002 & -0.0328 & 0 & -0.0505 \\ 0 & 0 & -0.0000 & 0.0001 & 0 & 0.0001 \\ 0 & 0 & 0.0000 & -0.0000 & 0 & 0.0000 \\ 0.0001 & 0 & 0.0075 & -3.2096 & 0 & 0.2204 \\ 0.0001 & 0 & 0.0148 & -6.4359 & 0 & 0.1599 \\ 0 & 0.0000 & 0 & 0 & -0.0857 & 0 \\ 0 & 0 & 0.0004 & -0.1599 & 0 & -0.0857 \\ 0 & 0 & 0.0000 & -0.0003 & 0 & -0.0002 \\ 0 & 0.0000 & 0 & 0 & 0.0000 & 0 \\ 0 & 0 & -0.0000 & 0.0002 & 0 & 0.0000 \end{bmatrix} \cdot 10^4$$

$$B = \begin{bmatrix} -0.0328 & -0.3109 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0001 & -0.0001 & 0 & 0.0001 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.0000 & 0.0000 & -0.0000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -3.2096 & -0.9397 & -0.7863 & 0.1878 & 0 & 0 & 0 & 0 & 0 & 0 \\ -6.4359 & -1.7001 & -1.6583 & 0.0614 & -0.3217 & 0 & 0.2824 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0658 & 0 & 0 & 0 & 0 \\ -0.1599 & -0.1839 & -0.0614 & -0.0658 & -0.2824 & 0 & 0.0658 & 0 & 0 & 0 \\ -0.0003 & -0.0003 & -0.0000 & -0.0002 & -0.0006 & 0 & 0.0002 & 0.0001 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.0001 & 0 & 0 & 0.0001 & 0 \\ 0.0002 & -0.0002 & 0.0002 & 0.0001 & -0.0002 & 0 & -0.0001 & 0 & 0 & 0.0001 \end{bmatrix} \cdot 10^4$$

Solve this using Eq. (8).

$$\Delta \vec{v}_{NC} = \begin{bmatrix} -18.96 \\ 0 \\ 0 \end{bmatrix} \frac{ft}{sec}$$

$$\Delta \vec{v}_{NH} = \begin{bmatrix} -1.21 \\ 0 \\ 0 \end{bmatrix} \frac{ft}{sec}$$

$$\Delta \vec{v}_{NSR} = \begin{bmatrix} 14.36 \\ 0 \\ 20.86 \end{bmatrix} \frac{ft}{sec}$$

$$\Delta \vec{v}_{TPI} = \begin{bmatrix} 3.05 \\ -0.21 \\ -2.09 \end{bmatrix} \frac{ft}{sec}$$

$$\Delta \vec{v}_{TPF} = \begin{bmatrix} 2.76 \\ -2.46 \\ 3.33 \end{bmatrix} \frac{ft}{sec}$$

Applying these $\Delta \vec{v}$ s at the appropriate times results in the relative motion trajectory shown in Figures 5 and 6.

CONCLUSION

When the accuracy associated with CW equations is adequate, the MM_CW_TGT method is useful as is. The non-iterative nature of the method results in fast and robust results. The author has used MM_CW_TGT for preliminary design of rendezvous scenarios.

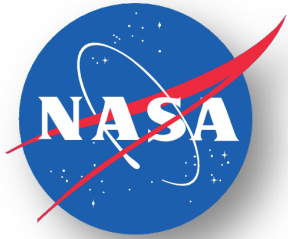
Although not yet implemented, the author expects MM_CW_TGT to be useful as a part of iterative methods. MM_CW_TGT can generate initial guesses for maneuvers, and can calculate corrections between iterations.

ACKNOWLEDGMENT

The author thanks Aaron Brown for suggestions that improved the readability of the paper.

REFERENCES

- [1] W. H. Clohessy and R. S. Wiltshire, "Terminal Guidance System for Satellite Rendezvous," *Journal of the Aerospace Sciences*, Vol. 27, No. 9, 1960, pp. 653–659.



Multi-Maneuver Clohessy-Wiltshire Targeting

David P. Dannemiller

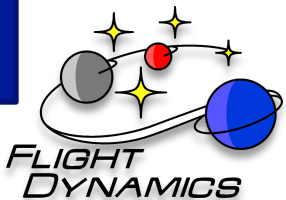
NASA – JSC

8/4/2011

(Astrodynamics Specialist Conference)

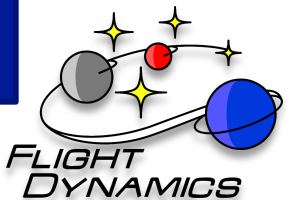
Introduction

- Orbital rendezvous
 - Sequence of maneuvers
 - Bring the chaser to a desired state relative to a target
 - Meet intermediate and final relative constraints
- Effect of maneuvers on constraints is often coupled, so the maneuvers must be solved for as a set
- Multi-Maneuver Clohessy-Wiltshire Targeting



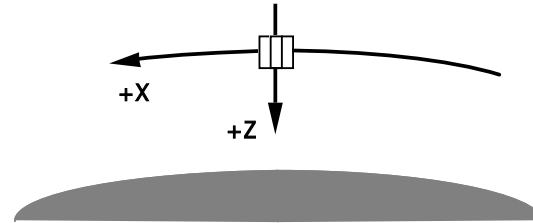
Agenda

- Notation
- Effect of a Series of Impulsive Maneuvers
- Simple Targeting Example
- Moderately Complex Targeting Example
- General Pattern
- Other Constraints
- Numerical Example
- Conclusion



Notation

- Local Vertical Curvilinear (LVC)



- State vector $\vec{s} = \begin{bmatrix} \vec{r} \\ \vec{v} \end{bmatrix}$ $\vec{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$, $\vec{v} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}$
- Clohessy-Wiltshire (CW) transition matrix C_m^n
 - Subscript – initial time
 - Superscript – final time
- Impulsive maneuvers
 - (-) pre-maneuver
 - (+) post-maneuver

$$\Delta \vec{v}_n = \vec{v}_{n+} - \vec{v}_{n-} = \begin{bmatrix} \Delta \dot{x}_n \\ \Delta \dot{y}_n \\ \Delta \dot{z}_n \end{bmatrix}$$

Effect of a Series of Impulsive Maneuvers

- Two maneuvers

$$\vec{s}_{0+} = \vec{s}_{0-} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix}$$

$$\vec{s}_{1-} = C_0^1 \cdot \vec{s}_{0+}$$

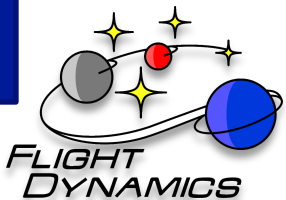
$$= C_0^1 \cdot \vec{s}_{0-} + C_0^1 \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix}$$

$$\vec{s}_{1+} = \vec{s}_{1-} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_1 \end{bmatrix}$$

$$= C_0^1 \cdot \vec{s}_{0-} + C_0^1 \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_1 \end{bmatrix}$$

- General

$$\vec{s}_{n+} = C_0^n \cdot \vec{s}_{0-} + C_0^n \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_0 \end{bmatrix} + C_1^n \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_1 \end{bmatrix} + C_2^n \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_2 \end{bmatrix} + \dots + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_n \end{bmatrix}$$



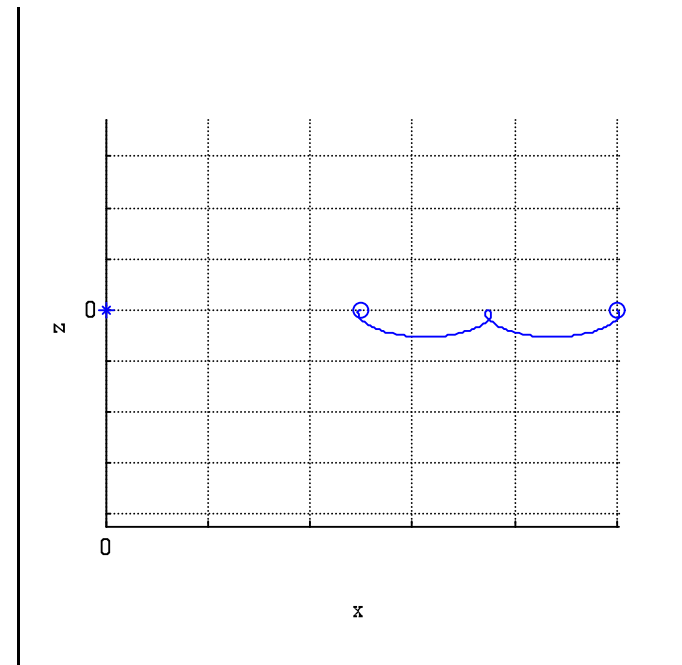
Simple Targeting Example

- Maneuvers

$$\Delta \vec{v}_0 = \begin{bmatrix} \Delta \dot{x}_0 \\ 0 \\ 0 \end{bmatrix}, \Delta \vec{v}_f = \begin{bmatrix} \Delta \dot{x}_f \\ 0 \\ 0 \end{bmatrix}$$

- Constraints

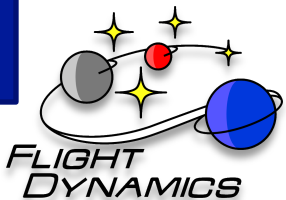
$$\vec{s}_{f+} = \begin{bmatrix} x_f \\ * \\ * \\ 0 \\ * \\ * \end{bmatrix}$$



Simple Targeting Example (cont)

$$\vec{s}_{f+} = C_0^f \cdot \vec{s}_{0-} + C_0^f \cdot \begin{bmatrix} \vec{0} \\ \Delta \vec{v}_0 \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta \vec{v}_f \end{bmatrix}$$

$$\vec{s}_{f+} = C_0^f \cdot \vec{s}_{0-} + \begin{bmatrix} * & * & * & C_0^f(1,4) & * & * \\ * & * & * & C_0^f(2,4) & * & * \\ * & * & * & C_0^f(3,4) & * & * \\ * & * & * & C_0^f(5,4) & * & * \\ * & * & * & C_0^f(5,4) & * & * \\ * & * & * & C_0^f(6,4) & * & * \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta \dot{x}_0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta \dot{x}_f \\ 0 \\ 0 \end{bmatrix}$$



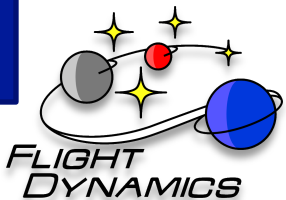
Simple Targeting Example (cont)

$$\vec{s}_{f+} = C_0^f \cdot \vec{s}_{0-} + \begin{bmatrix} C_0^f(1,4) & 0 \\ C_0^f(2,4) & 0 \\ C_0^f(3,4) & 0 \\ C_0^f(4,4) & 1 \\ C_0^f(5,4) & 0 \\ C_0^f(6,4) & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_0 \\ \Delta \dot{x}_f \end{bmatrix}$$

$$\begin{bmatrix} x_f \\ 0 \end{bmatrix} = \begin{bmatrix} C_0^f(1,1) & C_0^f(1,2) & C_0^f(1,3) & C_0^f(1,4) & C_0^f(1,5) & C_0^f(1,6) \\ C_0^f(4,1) & C_0^f(4,2) & C_0^f(4,3) & C_0^f(4,4) & C_0^f(4,5) & C_0^f(4,6) \end{bmatrix} \cdot \vec{s}_{0-} + \begin{bmatrix} C_0^f(1,4) & 0 \\ C_0^f(4,4) & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_0 \\ \Delta \dot{x}_f \end{bmatrix}$$

$$\vec{s}_{constraint} = A \cdot \vec{s}_{0-} + B \cdot \Delta \vec{v}$$

$$\Delta \vec{v} = B^{-1} \cdot (\vec{s}_{constraint} - A \cdot \vec{s}_{0-})$$



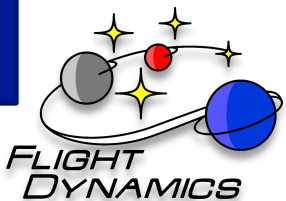
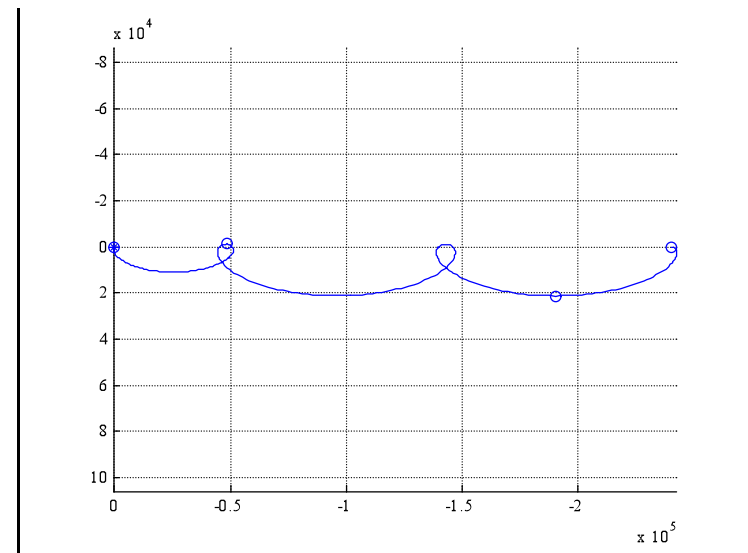
Moderately Complex Targeting Example

- Maneuvers

$$\Delta\vec{v}_{NC} = \begin{bmatrix} \Delta\dot{x}_{NC} \\ 0 \\ 0 \end{bmatrix}, \Delta\vec{v}_{NH} = \begin{bmatrix} \Delta\dot{x}_{NH} \\ 0 \\ 0 \end{bmatrix}, \Delta\vec{v}_{TI} = \begin{bmatrix} \Delta\dot{x}_{TI} \\ 0 \\ \Delta\dot{z}_{TI} \end{bmatrix}, \Delta\vec{v}_{TF} = \begin{bmatrix} \Delta\dot{x}_{TF} \\ 0 \\ \Delta\dot{z}_{TF} \end{bmatrix}$$

- Constraints

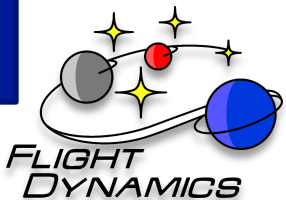
$$\vec{s}_{TI+} = \begin{bmatrix} x_{TI} \\ * \\ z_{TI} \\ * \\ * \\ * \end{bmatrix}, \vec{s}_{TF+} = \begin{bmatrix} x_{TF} \\ * \\ z_{TF} \\ \dot{x}_{TF+} \\ * \\ \dot{z}_{TF+} \end{bmatrix}$$



Moderately Complex Targeting Example (cont)

$$\vec{s}_{TI+} = C_0^{TI} \cdot \vec{s}_{0-} + C_{NC}^{TI} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{NC} \end{bmatrix} + C_{NH}^{TI} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{NH} \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{TI} \end{bmatrix}$$

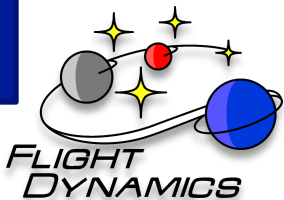
$$\vec{s}_{TF+} = C_0^{TF} \cdot \vec{s}_{0-} + C_{NC}^{TF} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{NC} \end{bmatrix} + C_{NH}^{TF} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{NH} \end{bmatrix} + C_{TI}^{TF} \cdot \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{TI} \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \Delta\vec{v}_{TF} \end{bmatrix}$$



Moderately Complex Targeting Example (cont)

$$\begin{bmatrix} x_{TI} \\ z_{TI} \\ x_{TF} \\ z_{TF} \\ \dot{x}_{TF+} \\ \dot{z}_{TF+} \end{bmatrix} = \begin{bmatrix} C_0^{TI}(1,1) & C_0^{TI}(1,2) & C_0^{TI}(1,3) & C_0^{TI}(1,4) & C_0^{TI}(1,5) & C_0^{TI}(1,6) \\ C_0^{TI}(3,1) & C_0^{TI}(3,2) & C_0^{TI}(3,3) & C_0^{TI}(3,4) & C_0^{TI}(3,5) & C_0^{TI}(3,6) \\ C_0^{TF}(1,1) & C_0^{TF}(1,2) & C_0^{TF}(1,3) & C_0^{TF}(1,4) & C_0^{TF}(1,5) & C_0^{TF}(1,6) \\ C_0^{TF}(3,1) & C_0^{TF}(3,2) & C_0^{TF}(3,3) & C_0^{TF}(3,4) & C_0^{TF}(3,5) & C_0^{TF}(3,6) \\ C_0^{TF}(4,1) & C_0^{TF}(4,2) & C_0^{TF}(4,3) & C_0^{TF}(4,4) & C_0^{TF}(4,5) & C_0^{TF}(4,6) \\ C_0^{TF}(6,1) & C_0^{TF}(6,2) & C_0^{TF}(6,3) & C_0^{TF}(6,4) & C_0^{TF}(6,5) & C_0^{TF}(6,6) \end{bmatrix} \cdot \vec{s}_0 -$$

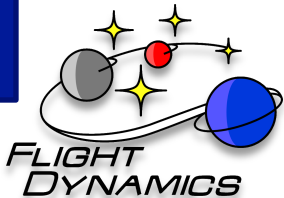
$$+ \begin{bmatrix} C_{NC}^{TI}(1,4) & C_{NH}^{TI}(1,4) & 0 & 0 & 0 & 0 \\ C_{NC}^{TI}(3,4) & C_{NH}^{TI}(3,4) & 0 & 0 & 0 & 0 \\ C_{NC}^{TF}(1,4) & C_{NH}^{TF}(1,4) & C_{TI}^{TF}(1,4) & C_{TI}^{TF}(1,6) & 0 & 0 \\ C_{NC}^{TF}(3,4) & C_{NH}^{TF}(3,4) & C_{TI}^{TF}(3,4) & C_{TI}^{TF}(3,6) & 0 & 0 \\ C_{NC}^{TF}(4,4) & C_{NH}^{TF}(4,4) & C_{TI}^{TF}(4,4) & C_{TI}^{TF}(4,6) & 1 & 0 \\ C_{NC}^{TF}(6,4) & C_{NH}^{TF}(6,4) & C_{TI}^{TF}(6,4) & C_{TI}^{TF}(6,6) & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta \dot{x}_{NC} \\ \Delta \dot{x}_{NH} \\ \Delta \dot{x}_{TI} \\ \Delta \dot{z}_{TI} \\ \Delta \dot{x}_{TF} \\ \Delta \dot{z}_{TF} \end{bmatrix}$$



General Pattern

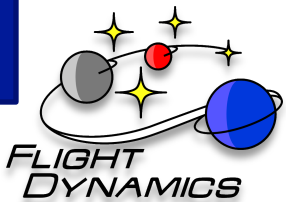
1. The selected maneuver components and constraints are used to form a system of linear equations of the form $\vec{s}_{constraint} = A \cdot \vec{s}_{0-} + B \cdot \Delta\vec{v}$.
2. The number of maneuver components must be equal to the number of constraints, which is denoted as n .
3. $\vec{s}_{constraint}$ is an n -vector that contains the selected constraints.
4. $\Delta\vec{v}$ is an n -vector that contains the selected maneuver components.
5. A is an $n \times 6$ matrix that contains the effects of \vec{s}_{0-} on the constraints. Each row of A contains a row from the CW transition matrix, the specific row being that associated with the selected constraint, and the transfer time equal to $t_{constraint} - t_0$.
6. B is an $n \times n$ matrix that contains the effects of the maneuver components on the constraints. Each row of B contains terms from the CW transition matrix, the specific row being that associated with the constraint, the specific column being that associated with the maneuver component, and the transfer time equal to $t_{constraint} - t_{maneuver}$. If the constraint time is after the maneuver time, the term is 0 (later maneuvers cannot effect earlier constraints).
7. B must be invertible. This is assured if the designer poses a controllable scenario – a scenario where the constraints can be met by the sequence of maneuvers.
8. The linear system is solved for the unknown maneuver components.

$$\Delta\vec{v} = B^{-1} \cdot (\vec{s}_{constraint} - A \cdot \vec{s}_{0-})$$



Other Constraints

- Acceleration Constraint
 - Useful to specify co-elliptic constraint
- Elevation Angle Constraint
 - Often used in terminal phase

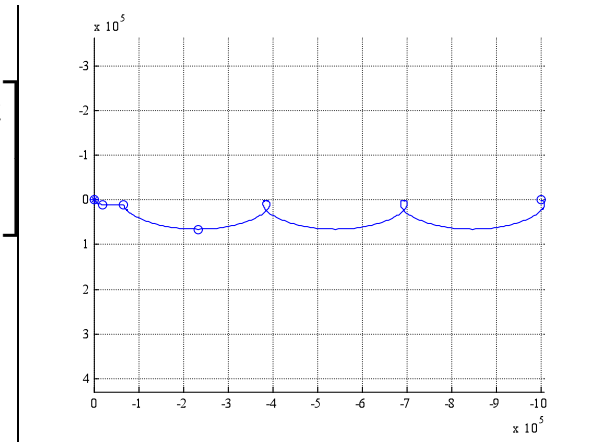


NC, NH, NSR, TPI, TPF Numerical Example

- Maneuvers

$$\Delta \vec{v}_{NC} = \begin{bmatrix} \Delta \dot{x}_{NC} \\ 0 \\ 0 \end{bmatrix}, \quad \Delta \vec{v}_{NH} = \begin{bmatrix} \Delta \dot{x}_{NH} \\ 0 \\ 0 \end{bmatrix}, \quad \Delta \vec{v}_{NSR} = \begin{bmatrix} \Delta \dot{x}_{NSR} \\ 0 \\ \Delta \dot{z}_{NSR} \end{bmatrix}$$

$$\Delta \vec{v}_{TPI} = \begin{bmatrix} \Delta \dot{x}_{TPI} \\ \Delta \dot{y}_{TPI} \\ \Delta \dot{z}_{TPI} \end{bmatrix}, \quad \Delta \vec{v}_{TPF} = \begin{bmatrix} \Delta \dot{x}_{TPF} \\ \Delta \dot{y}_{TPF} \\ \Delta \dot{z}_{TPF} \end{bmatrix}$$



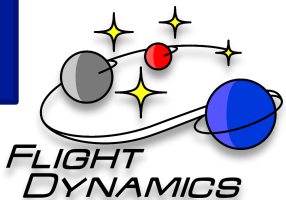
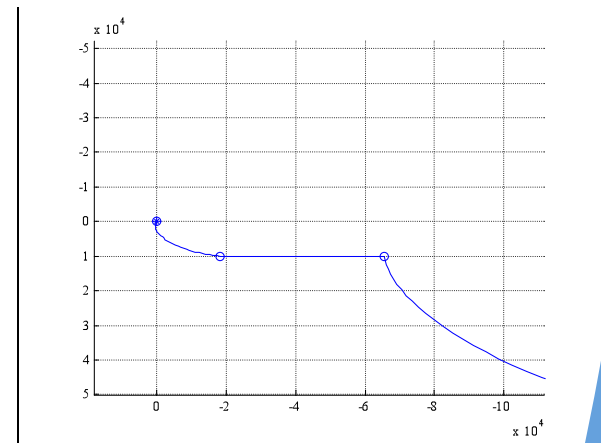
- Constraints

$$z_{NSR} = 10,000 \text{ ft}, \quad \dot{z}_{NSR+} = 0, \quad \ddot{z}_{NSR+} = 0$$

$$\theta_{TPI} = .5 \text{ rad}$$

$$\vec{s}_{TPF+} = \vec{0}$$

- Numerical solution in paper



Conclusion

- Non-iterative nature of the method results in fast and robust results
- Method has been used for preliminary design of rendezvous scenarios
- Method expected to be useful as a part of iterative methods
 - Generate initial guesses for maneuvers
 - Calculate corrections between iterations

$$\vec{s}_{constraint} = A \cdot \vec{s}_{0-} + B \cdot \Delta \vec{v}$$

