Helical Turns: Part 2, Controls

W. Premerlani and Peter Hollands, May 1, 2015

This document is part two of a three part document describing the theory and implementation of helical turn controls. This part picks up where part one left off, turn conditions, and describes the control algorithms for developing signals for ailerons, elevator, and rudder functions for producing the desired helical turn.

The starting point for the controls is the target value for the tilt vector and rotation rate vector, which are computed from desired turn rate, target pitch, and airspeed according to the equations presented in part 1, some of which are reviewed here:

The desired earth frame vertical axis turn rate in radians per second is specified:

$$\tilde{\omega} = \text{desired rotation rate around earth frame vertical axis} \qquad \text{Equation 1}$$

The desired pitch is specified as the ratio of the vertical component of the air speed vector in the earth frame, divided by the magnitude of the earth frame horizontal air speed:

$$\tilde{p} = \frac{\tilde{S}_V}{\tilde{S}_H} = \text{desired pitch}$$

$$p = \frac{S_V}{S_H} = \text{actual pitch} \qquad \text{Equation 2}$$

$$\tilde{S}_V = \text{desired vertical airspeed, positive is upward in earth frame}$$

$$S_V = \text{actual vertical airspeed}$$

$$S_H = \text{horizontal airspeed}$$

$$S = \sqrt{S_V^2 + S_H^2} = \text{total airspeed}$$

The target body frame rotation rate vector is related to the actual vertical orientation and the desired earth frame rotation rate around the vertical axis by:

$$\tilde{\boldsymbol{\Omega}} = \left[\tilde{\omega} \cdot X \quad \tilde{\omega} \cdot Y \quad \tilde{\omega} \cdot Z\right]$$

$$\tilde{\boldsymbol{\Omega}} = \text{desired rotation rate vector in the body frame} \qquad \text{Equation 3}$$

It was shown in part 1 that the values of the components of the target tilt vector must satisfy the following equations:

$$\frac{\breve{X}}{\breve{Z}} = \frac{-\breve{d}\cdot S}{g}$$
$$\breve{Y} = -\frac{\breve{p}}{\sqrt{1+\breve{p}^2}} \qquad \text{Equation 4}$$

Equation 4 can be implemented in a two steps. The first step is to compute the following:

$$\breve{\mathbf{R}}_1 = \begin{bmatrix}\breve{X}_1 & \breve{Y}_1 & \breve{Z}_1\end{bmatrix} = \frac{\begin{bmatrix}-\frac{\breve{d}\cdot S}{g} & 0 & 1\end{bmatrix}}{\sqrt{\left(\frac{\breve{d}\cdot S}{g}\right)^2+1}} \qquad \text{Equation 5}$$

In other words, the first step computes roll only. Start with the roll parameter placed in the X element, place 1 in the Z element, and normalize. This will result in a first pass result with Y equal to zero, and the sum of the squares of X and Z elements equal to 1. Then, insert the desired value of pitch in the Y element, and renormalize. The ratio of X/Z will be preserved, and the second part of equation 10 will be satisfied:

$$\breve{\mathbf{R}} = \begin{bmatrix}\breve{X} & \breve{Y} & \breve{Z}\end{bmatrix} = \frac{\begin{bmatrix}\breve{X}_1 & -\breve{p} & \breve{Z}_1\end{bmatrix}}{\sqrt{\left(\breve{p}\right)^2+1}} \qquad \text{Equation 6}$$

The actual tilt vector is the bottom row of the direction cosine matrix, denoted here by:

$$\mathbf{R} = \begin{bmatrix}X & Y & Z\end{bmatrix} \qquad \text{Equation 7}$$

At this point we have two conditions the controls must achieve in order to produce the desired helical turn. Equations 5 and 6 prescribe the target tilt vector in the body frame. Equation 3 prescribes the target rotation rate vector in the body frame. These conditions constitute the starting point for the controls.

Start with the controls needed to satisfy equation 6. Tests have shown that using the helical turn approach for prescribing the turn conditions, a simple proportional feedback control can be used to satisfy equation 6 and produce the turn. There are at least two methods that could be used.

The first method is to start with the cross product of the target and actual tilt vectors. The result is a vector whose magnitude is equal to the sine of the angle between the two vectors, and which is perpendicular to both of them:

1. Take the cross product of the target and actual tilt vectors. This produces a vector indicative of a rotation that could be applied to the actual tilt to drive it toward the target tilt.

2. Flip the sign of the resultant cross product. That is because the tilt vector is the image of the earth vertical in the body frame. In order to get the earth vertical to rotate in one direction as seen in the body frame, we need to rotate the aircraft in the opposite direction.
3. The X, Y and Z components of the result of step 2 will be body frame errors that can be reduced by applying a deflection to the corresponding control surface. In the body frame, X maps to the elevator, Y maps to the ailerons, and Z maps to the rudder. So, take each component, multiply it by a corresponding gain, and apply it to the corresponding control surface.

The above method was implemented and tested. It worked well enough during a turn, but not so well going into a turn, coming out of a turn, or during a 180 degree roll from normal to inverted orientation, or vice versa. In particular, during a 180 degree roll in which the target tilt vector was [0 , 0 , 1] or [0 , 0 , -1], the rudder did not deflect the way an expert pilot would use it in that situation. Peter Hollands suggested that pitch control could be improved under conditions of extreme tilt misalignment by defining pitch misalignment in the earth frame rather than in the body frame. So a second method was developed and tested for pitch control and was found to perform well under all conditions. For a more detailed discussion of this second method for pitch control, see Appendix 1.

The final design for the proportional feedback treated roll control and pitch control separately. For roll control, first compute the cross and the dot products of the actual and desired roll vectors:

$$\begin{aligned} rollDot &= X \cdot \overset{\breve{E}}{X} + Z \cdot \overset{\breve{E}}{Z} \\ rollCross &= Z \cdot \overset{\breve{E}}{X} - \overset{\breve{E}}{Z} \cdot X \end{aligned} \qquad \text{Equation 8}$$

Use roll dot and roll cross to compute roll error. Roll error is equal to the cross product when the roll error angle is less than 90 degrees, and is equal to the sign of the cross product if the angle is greater than or equal to 90 degrees. This facilitates moving back and forth between normal and inverted flight:

$$\begin{aligned} & if\,(rollDot > 0) \\ & rollError = rollCross \\ & else \\ & rollError = sign(rollCross) \end{aligned} \qquad \text{Equation 9}$$

Multiply the roll error by a gain and apply it to the ailerons. This will cause the ailerons to drive the actual roll orientation of the plane to match the desired orientation.

For pitch control, start by computing the pitch error in the earth frame:

$$PitchError = \overset{\breve{E}}{Y}\sqrt{X^2 + Z^2} - Y\sqrt{\overset{\breve{E}}{X}{}^2 + \overset{\breve{E}}{Z}{}^2} \qquad \text{Equation 10}$$

The pitch error produced by equation 10 is the sine of the pitch error angle. In this case, we are interested only in values between plus and minus ninety degrees.

The projections of the earth frame pitch error into the body frame produces the body frame pitch error vector:

$$PitchErrorBody = \frac{PitchError}{\sqrt{X^2 + Z^2}} \cdot \begin{bmatrix} -Z & 0 & X \end{bmatrix} \qquad \text{Equation 11}$$

The first component projects onto the elevator. Multiply it by an elevator gain to compute elevator deflection. The third component projects onto the rudder. Multiply it by a rudder gain to compute rudder deflection. The second component is explicitly shown as zero to remind us that the ailerons are not used to control pitch.

Gains can be fixed, which will work for most types of aircraft, or could be inversely proportional to airspeed, which would be appropriate for aircraft that operate over a wide range of speeds.

At this point, using proportional feedback only, tests have shown that control performance is good for both fly-by-wire pilot control and autopilot navigation (waypoint mode), during both normal and inverted flight, including aggressive inverted turns. In waypoint mode, turns are tight and smooth.

However, with proportional only control, tests show that it takes a few seconds for the orientation of the aircraft to attain what is prescribed for the desired turn rate. Response can be sped up by adding feed forward terms. Feed forward has the advantage of not changing the dynamics of the feedback loops, so it does not impact the stability of the feedback. Assuming that the elevator trim is adjusted for airspeed and wing loading, the feed forward deflections should be proportional to the components of the desired rotation rate vector defined in equation 3. Since the steady state rotation rate will be proportional to the air speed, the feed forward deflections for elevator, ailerons, and rudder should be equal to elevator, aileron, and rudder gains times the components the vector defined by equation 3, where the gains are inversely proportional to air speed.

Most model aircraft have inherent damping on each axis because forces on aircraft surfaces depend on angles of attack, which in turn depend on rotation rates. However, in some cases in which there is not much inherent damping and it is desired to limit the rotation rate, it is appropriate to include a feedback term proportional to the difference between the desired body frame rotation rates defined by equation 3, and the actual body frame rotation rates measured by the gyros. By using equation 3 for the damping feedback, any tendency for the rate feedback to conflict with the body frame turn rates required for the turn will be eliminated.

Finally, we consider integral feedback. As will be shown in part 3 of this series of papers, not only is integral feedback not needed, it is counter productive.

At this time all control gains in MatrixPilot described above are constants, but may eventually be made functions of airspeed if there is sufficient demand by users with aircraft that fly over a wide range of airspeeds.

Appendix: Earth frame pitch control

The matrix R transforms from body to earth. R transpose transforms from earth to body. Denote the elements of the matrix stored as a one dimensional array as r0 through r8. The bottom row of the matrix, $\begin{bmatrix} r_6 & r_7 & r_8 \end{bmatrix}$, is also referred to in the helical turn control documents as the vector $\begin{bmatrix} X & Y & Z \end{bmatrix}$. The actual pitch is Y. The desired pitch is $Y\text{Ĕ}$.

Pitch orientation is defined in terms of the fuselage, which maps to a vector in the earth frame equal to the second column of the matrix. Therefore, the unit column vector in the earth frame parallel to the fuselage is given by:

$$\mathbf{F}_{\mathbf{earth}} = \begin{bmatrix} r_1 \\ r_4 \\ r_7 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_4 \\ Y \end{bmatrix} \qquad \text{Equation A.1}$$

However, the desired pitch is $Y\text{Ĕ}$. We would like a rotation vector in the horizontal plane of the earth frame and perpendicular to the fuselage that will move from the actual pitch to the desired pitch. Such a vector will not disturb the yaw orientation in the earth frame. To construct a target vector for the fuselage that does not disturb yaw orientation, we construct the target vector such that its projection onto the horizontal plane is parallel to the projection of the actual fuselage. In other words, the target vector is given by:

$$\mathbf{F}\text{Ĕ}_{\mathbf{earth}} = \begin{bmatrix} \alpha \cdot r_1 \\ \alpha \cdot r_4 \\ Y\text{Ĕ} \end{bmatrix} \qquad \text{Equation A.2}$$

where α is selected to normalize $\mathbf{F}\text{Ĕ}_{\mathbf{earth}}$

The normalization of the target vector leads to the following condition:

$$\alpha^2 \cdot \left[r_1^2 + r_4^2\right] + Y\text{Ĕ}^2 = 1 \qquad \text{Equation A.3}$$

But the magnitude of the actual unit vector parallel to the fuselage is 1. Therefore the first and fourth elements of the matrix satisfy the following:

$$\left[r_1^2 + r_4^2\right] + Y^2 = 1 \qquad \text{Equation A.4}$$

Therefore we can express alpha in terms of the target and actual values of the bottom row of the matrix:

$$\alpha = \frac{\sqrt{1-\breve{Y}^2}}{\sqrt{X^2+Z^2}} = \frac{\sqrt{1-\breve{Y}^2}}{\sqrt{r_6^2+r_8^2}}$$ Equation A.5

Conceptually, the next step is to take the cross product of the target and actual fuselage vectors in the earth frame and then transform into the body frame. Computationally, we can achieve the same result by first transforming the vectors into the body frame and then taking the cross product. Transforming the actual vector into the body frame, we find:

$$\mathbf{F}_{\mathbf{body}} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$ Equation A.6

Equation A.6 should not be too surprising, the Y axis in the body frame is the fuselage by definition.

Transforming the target vector into the body frame we find:

$$\breve{\mathbf{F}}_{\mathbf{body}} = \begin{bmatrix} 0 \\ \alpha \\ 0 \end{bmatrix} + \left(\breve{Y} - \alpha \cdot Y\right) \begin{bmatrix} r_6 \\ r_7 \\ r_8 \end{bmatrix}$$ Equation A.7

To find the pitch vector error in the body frame, take the cross product of equation 6 and equation 7:

$$\breve{\mathbf{F}}_{\mathbf{body}} \times \mathbf{F}_{\mathbf{body}} = \left(\breve{Y} - \alpha \cdot Y\right) \begin{bmatrix} -r_8 \\ 0 \\ r_6 \end{bmatrix}$$ Equation A.8

It remains to substitute for alpha. It is convenient to start by computing the magnitude of the cross product of the target and actual fuselage directions in the earth frame with the following:

$$PitchError = \breve{Y}\sqrt{r_6^2+r_8^2} - Y\sqrt{1-\breve{Y}^2}$$ Equation A.9

Equation 9 is computed by taking the cross product of two earth frame two dimensional pitch vectors. Substitute equations 9 and 5 into equation 8 to yield:

$$\breve{\mathbf{F}}_{\mathbf{body}} \times \mathbf{F}_{\mathbf{body}} = \frac{PitchError}{\sqrt{r_6^2+r_8^2}} \begin{bmatrix} -r_8 \\ 0 \\ r_6 \end{bmatrix}$$ Equation A.10

Equations A.9 and A.10 prescribe a simply way to map the earth frame pitch error onto the elevator and rudder. Use equation A.9 to compute pitch error in the earth frame. Use equation A10 to project the earth frame pitch error onto the elevator and rudder. The first component of the vector in equation A.10 maps to the elevator, the third component maps to the rudder.

Equations A.9 and A.10 can also be expressed in terms of X and Z, and with a row vector instead of a column vector:

$$PitchError = Y\text{Ĕ}\sqrt{X^2 + Z^2} - Y\sqrt{X\text{Ĕ}^2 + Z\text{Ĕ}^2} \qquad \text{Equation A.11}$$

$$\mathbf{F}\text{Ĕ}_{\mathbf{body}} \times \mathbf{F}_{\mathbf{body}} = \frac{PitchError}{\sqrt{X^2 + Z^2}}\begin{bmatrix} -Z & 0 & X \end{bmatrix} \qquad \text{Equation A.12}$$

It is instructive to compare the pitch error vector that we would obtain by taking the cross product of target and actual tilt vector with equations A.11 and A.12. If we use the cross product of the tilt vector, the pitch error vector becomes:

$$PitchErrorVector = \begin{bmatrix} Y \cdot Z\text{Ĕ} - Y\text{Ĕ} \cdot Z & 0 & X \cdot Y\text{Ĕ} - X\text{Ĕ} \cdot Y \end{bmatrix} \qquad \text{Equation A.13}$$

It is instructive to compare equations A.13 and A.12 under various conditions. Start with a simple case in which the target orientation is level, normal orientation, the plane has no roll, it has pitch only. In that case both methods yield the same pitch error vector, and both will respond only with the elevator:

$$PitchErrorVector = \begin{bmatrix} Y & 0 & 0 \end{bmatrix} \qquad \text{Equation A.14}$$

For inverted flight, both methods will yield the vector in equation A.14, with the sign flipped.

Now consider the case in which the target orientation is inverted level flight, and it is halfway through a roll to get there, with Z = 0 and X close to 1. In that case, the pitch error vector using equation A.13 is:

$$PitchErrorVector = \begin{bmatrix} -Y & 0 & 0 \end{bmatrix} \qquad \text{Equation A.15}$$

However, equation A.14 yields:

$$PitchErrorVector = \begin{bmatrix} 0 & 0 & -Y \end{bmatrix} \qquad \text{Equation A.16}$$

In other words, using earth frame pitch error projected onto elevator and rudder produces a sensible result during a snap roll, namel the rudder is used to control pitch with 90 degree roll, whereas using the cross product of the tilt vectors attempts to use the elevator.

There is one more interesting case to consider, the situation in which the plane is in a turn and the controls have achieved the desired roll attitude, the target pitch is zero, and the actual pitch is not zero. In that case the two methods are entirely equivalent and each of them produces the same pitch error vector:

$$PitchErrorVector = Y \cdot \begin{bmatrix} Z' & 0 & -X' \end{bmatrix} \qquad \text{Equation A.17}$$

Equation A.17 shows that either method would properly map the pitch error onto the elevator and rudder, as long the roll error is small. So, it is only the case of a snap roll in which equations A.11 and A.12 have a clear advantage.

The interested reader is invited to verify equation A.17 by making the following substitutions into equations A.11, A.12, and A.13:

$$\begin{aligned} Y' &= 0 \\ X &= X' \cdot \sqrt{1-Y^2} \qquad \text{Equation A.18} \\ Z &= Z' \cdot \sqrt{1-Y^2} \end{aligned}$$