```
function [tSeg,xSeg,uSeg,cmdSeg] = UAVSim( t, x0, data, p )
% Usage: Simulate a point mass aircraft model from an initial state w/ steering.
%   State:     x = [V;gama;psi;x;y;h;Tbar]
%   -------------------------------------
%     V       true airspeed
%     gamma   air relative flight path angle
%     psi     air relative flight heading angle
%     x       East position
%     y       North position
%     h       altitude
%     Tbar    normalized excess thrust
%
%   Control:   u = [Lbar;phi;Tcbar]
%   -------------------------------
%     Lbar    normalized excess lift
%     phi     bank angle
%     Tcbar   normalized excess thrust command
%
%   Command:   cmd = [v;psi;h;x;y]
%   ------------------------------
%     v       velocity command (true airspeed, m/s)
%     psi     heading command (rad)
%     h       altitude command (m)
%     x       eastward position (m)
%     y       northward position (m)
%
%-------------------------------------------------------------------------
%   Form:
%   [tSeg,xSeg,uSeg,cmdSeg] = UAVSim( time, x0, data, p );
%-------------------------------------------------------------------------
%
%   ------
%   INPUTS
%   ------
%   t       (1,N)     Time vector
%   x0      (7,1)     Initial state vector
%   data              Data structure with fields:
%                       g     Gravitational acceleration
%                       Kh    Altitude control gains
%                       KL    Lateral control gains
%                       Ks    Longitudinal control gains
%   p       (.)       Flight parameters. This function uses the following
%                     fields:
%                       wp        (3,1)   Target waypoint position (x,y,h)
%                       Rmin      (1,1)   Minimum turn radius (m)
%                       hDotMax   (1,1)   Maximum climb rate (abs val)
%                       dT        (1,1)   Time step
%                       stopSim = @(t,x)  Anonymous function. Sim terminates
%                                         when this evaluates to true.
%
%   -------
%   OUTPUTS
%   -------
%   tSeg    (1,M)     Time vector for this segment. Equivalent to "t" input.
%   xSeg    (7,M)     State vector across time for this segment.
%   uSeg    (3,M)     Control vector across time for this segment.
%   cmdSeg  (3,M)     Commands (v,h,psi) across time for this segment.
%
% ** Note: M>=N. The simulation may terminate before it reaches the end of
```

```
% the original input time vector.
%
%-------------------------------------------------------------------------
  nt = length(t);
  ns = size(x0,1);

  if nargin < 4
    p = [];
  end

  if ~isfield(p,'stopSim')
    p.stopSim = @(t,x) 0;
  end

  if( nargin < 3 )
    data.a   = zeros(3,1);
    data.W   = data.a;
    data.g   = 9.81;
    data.tau = 5;
    wn = 0.1;
    zeta = 0.6;
    data.Kh  = [2*wn*zeta, wn^2]; % altitude control gains
    data.KL  = [.1, .005];        % lateral control gains
    data.Ks  = [.1, .001];        % longitudinal control gains
  end

  tSeg = t;
  xSeg = zeros(ns,nt);
  uSeg = zeros(3,nt);
  cmdSeg = zeros(5,nt);

  dT = diff(t);

  % initial state
  xSeg(:,1) = x0;

  xtmp = x0;
  cmdDot = zeros(5,1);
  for k=1:nt-1

    % compute commands
    %=================
    % fprintf('iter %d\n',n)
%     disp(k)
%     disp('state vec')
%     disp(xtmp)
    [cmd,cmdDot] = UAVGuidance(t(k), xtmp, p );

    cmdSeg(:,k) = cmd;
%     disp('post guidance')
%
%     disp(xSeg(:,k))
%     disp(cmdSeg(:,k))
%     disp(cmdDot)

    % compute controls
    %=================
    u = UAVControl( xSeg(:,k), cmdSeg(:,k), cmdDot, data );
%     disp('post control')
%     disp(u)
%     uSeg(:,k) = u;
```

```
    % integrate state with controls
    %==============================
    rhs = @(t,x) UAVRHS(x,u,data.g,data.tau);
    xx = ODENumIntRK4(rhs,[0 dT(k)],xtmp')';
    xtmp = xx(2,:); % Get only the new part of xtmp

    % if (mod(k, 1000) == 0) % Print every 1000 iterations
    %   fprintf('iter %d\n',k)
    %   disp('state')
    %   disp(xtmp)
    %   disp('cmd')
    %   disp(cmd)
    %   disp('cmdDot')
    %   disp(cmdDot)
    %   disp('u')
    %   disp(u)
    % end

    % store state data
    %=================
    xSeg(:,k+1) = xtmp;

    % terminate if we are close enough to the target waypoint
    %========================================================
    if p.stopSim(t(k),xtmp)
      % disp('stop activated at:')
      % disp(t(k))
      break
    end

  end

  k = k+1;
  uSeg(:,k) = UAVControl( xSeg(:,k), cmdSeg(:,k), cmdDot, data );

  % Trim any excess columns
  tSeg = tSeg(1:k);
  xSeg = xSeg(:,1:k);
  uSeg = uSeg(:,1:k);
  cmdSeg = cmdSeg(:,1:k);

end
```

Published with MATLAB® R2020a