```
function lhs = smooth_mklhs(kpts)
% The lhs of the system of equations smoothest forward curves.
%
% kpts: Time moments associated with the discount factor array.

% First, create the matrix: each knot point will add five more
% coefficients.  Thus the size of the dense matrix that we are interested
% in is [5*length(kpts)]x[5*length(kpts)]. Of course, we use at most ten
% coefficients on any one line.

m = zeros(5 * length(kpts));

% Note that the first and the last block of equations is special.

% Set up the conditions at t = 0.

% fwd'''(0)
lhs(1, 2) = 6;

% fwd''(0)
lhs(2, 3) = 2;

% Base index for the current unknowns. The coefficients of the current
% interval have indices bindex..bindex+4, and the coefficients of the next
% one have indices bindex+5..bindex+9.
bindex = 1;

% Counter that points to the current equation (the one that is currently
% built).
eqn = 3;

% These are the powers of the "preceding" knot point. Recall that 0 is alwa↙
ys
% the first (implicit) knot point.

tprec = 0;

% Now set up the blocks of five equations, one for each knot point,
% except for the last one.

for i = 1 : length(kpts) - 1

  % Compute the powers of kpts(i).
  t1 = kpts(i) - tprec;
  t2 = t1 * t1;
  t3 = t1 * t2;
  t4 = t1 * t3;
  t5 = t1 * t4;
```

```
  % Continuity of the 3rd derivative at the right hand side of the interval↙
;
  % Continuity of the 2nd derivative at the right side of the interval;
  % Continuity of the 1st derivative at the right side of the interval;
  % Continuity of the curve at the right side of the interval;
  % Condition on the price at the right end of the current interval.
  lhs(eqn : eqn + 4, bindex : bindex + 9) = ...
  [ ...
      24*t1  6        0    0  0      0 -6  0  0  0; ...
      12*t2  6*t1     2    0  0      0  0 -2  0  0; ...
       4*t3  3*t2  2*t1    1  0      0  0  0 -1  0; ...
         t4    t3    t2   t1  1      0  0  0  0 -1; ...
       t5/5  t4/4  t3/3 t2/2 t1      0  0  0  0  0  ...
   ];

  % In the next cycle, this knot point will become the previous one.
  tprec = kpts(i);

  % Move to the next equation and to the next set of unknowns.
  eqn = eqn + 5;
  bindex = bindex + 5;

end


t1 = kpts(end) - tprec;
t2 = t1 * t1;
t3 = t1 * t2;
t4 = t1 * t3;
t5 = t1 * t4;

% Now add the equations that refer to the last interval.

% fwd'''(end)
tmp1 = [24*t1 6 0 0 0];

% fwd''(end)
tmp2 = [12*t2 6*t1 2 0 0];

lhs(eqn : eqn + 2, bindex : bindex + 4) = ...
  [ ...
    t5/5 t4/4 t3/3 t2/2  t1; ... % Price at the right end of the interval.
    tmp1;
    tmp2
  ];
```