**function [Kx, Kr] = placePoles(A, B, C, poles)**

**sz = size(A);**

**N = sz(1);**

**if N <= 1**

**error('System needs to have an order greater than 1.');**

**elseif rank(A) ~= N**

**error('System is not controllable');**

**end**

**if isControllerCanonical(A)**

**Pd = poly(poles);**

**P = poly(eig(A));**

**dP = Pd - P;**

**Kx = dP([N+1:2]);**

**else**

**P = poly(eig(A));**

**% T1 is controllability matrix [B AB A^2B ... A^N-1B]**

**T1 = zeros(N,N);**

**for i=0:N-1**

**T1(:,i+1) = (A^i) \* B;**

**end**

**% T2 involves the characteristic polynomial coefficients of A**

**T2 = zeros(N,N);**

**for i=0:N-1**

**T2(i+1,:) = [zeros(1,i), P(1:end-i-1)];**

**end**

**% Swap rows of identity matrix to get T3**

**T3 = eye(N);**

**for i=1:floor(N/2)**

**ii = T3(i,:);**

**T3(i,:) = T3(N-i+1,:);**

**T3(N-i+1,:) = ii;**

**end**

**% Similarity transform matrix**

**T = T1\*T2\*T3;**

**% Obtain state and input matrices of similar system**

**Az = inv(T)\*A\*T;**

**Bz = inv(T)\*B;**

**% Now obtain Kz needed for similar system**

**Pd = poly(poles);**

**dP = Pd - P;**

**Kz = dP([N+1:-1:2]);**

**% Perform similarity transform to obtain corresponding Kx**

**Kx = Kz\*inv(T);**

**end**

**Kr = -1 / (C\*inv(A-B\*Kx)\*B);**

**end**

**function [val] = isControllerCanonical(A)**

**sz = size(A);**

**N = sz(1);**

**if sz(1)~= sz(2)**

**error("Matrix is not square!");**

**end**

**col1 = A(:,1);**

**col1\_excluding\_bottom = col1([1:N-1]);**

**zero\_vec = zeros(N-1,1);**

**if ~isequal(col1\_excluding\_bottom,zero\_vec)**

**val = false;**

**return;**

**end**

**B = A(1:N-1,2:N);**

**if isequal(B,eye(N-1))**

**val = true;**

**return;**

**else**

**val = false;**

**return;**

**end**

**end**