

structState.disp is a matrix of [U (ndf x 1), DU (ndf x 1), dU (ndf x 1)]

P = applied load matrix (ndf x num_step)

for n = 1:num_step - 1

 conv_flag = 0

 iter = 0

 while conv_flag is 0 and iter < iter_max

 [K, R] = SSD(structState.disp, ...)

 unb = P(:, n+1) - R

 if |unb| < tol

 save structState.disp(:, 1) for nth load step

 set structState.disp(:, 2) and structState.disp(:, 3) to 0

 conv_flag = 1

 else

 structState.disp(:, 3) = inv(K) * unb

 update structState.disp(:, 1) += structState.disp(:, 3)

 update structState.disp(:, 2) += structState.disp(:, 3)

 iter += 1

 end

 end

end