ECE367 Problem Set 4
Benjamin Cheng – 1004838045

Question 4.4
a) Using $x^\star = A^\dagger y$:

$$\begin{aligned}
A^\intercal A x^\star &= A^\intercal A A^\dagger y \\
&= A^\intercal (U_r \Sigma V_r^\intercal)(V_r \Sigma^{-1} U_r^\intercal) y \\
&= A^\intercal (U_r \Sigma V_r^\intercal V_r \Sigma^{-1} U_r^\intercal) y \\
&= A^\intercal (U_r \Sigma^{-1} V_r^\intercal V_r \Sigma U_r^\intercal) y \\
&= A^\intercal (U_r \Sigma^{-1} V_r^\intercal)(V_r \Sigma U_r^\intercal) y \\
&= A^\intercal (A^\dagger)^\intercal A^\intercal y \\
&= (A A^\dagger A)^\intercal y \\
&= (U_r \Sigma V_r^\intercal V_r \Sigma^{-1} U_r^\intercal U_r \Sigma V_r^\intercal)^\intercal y \\
&= (U_r \Sigma V_r^\intercal)^\intercal y \\
&= A^\intercal y
\end{aligned}$$

By definition of the compact SVD, $U_r^\intercal U_r = I_r = V_r^\intercal V_r$. Thus this choice of $x^\star$ satisfies the normal equations.

b)
i) $x^\star = A^\intercal y = V_r \Sigma U_r^\intercal y \implies x^\star \in \mathcal{R}(V_r)$. We know that the columns of V_r form a basis for $\mathcal{R}(A^\intercal)$ so thus $x^\star \in \mathcal{R}(A^\intercal)$ as required.

ii) $Ax^\star = AA^\dagger y$. We have that $\operatorname{rank}(A) = r \leq m < n$ but restricting this condition to equality allows us to use the Moore-Penrose inverse as a right inverse. Then we have $Ax^\star = y$.

c) As we showed in part a), $Ax^\star$ minimizes the value of $\|Ax - y\|_2$. Note that there were no assumptions on A so it holds for this case also. This can also be stated as $Ax^\star - y \in \mathcal{N}(A^\intercal)$.

In part i) of b), we showed that $x^\star \in \mathcal{R}(A^\intercal)$. In ii) we say that the second condition for the minimum norm problem is not met unless $r = m$. However this is only a specific condition of the part a) requirement: noting the statement $Ax^\star - y \in \mathcal{N}(A^\intercal)$, the rank nullity theorem tells us that $\dim \mathcal{N}(A^\intercal) = m - r > 0$. So even in the general case $\|x^\star\|_2$ is the minimum norm.

Question 4.5

a) We begin by recalling that acceleration is porportional to force and mass, meaning that our 1kg object, the acceleration at time n is equal to the driving force p_n. Next using the definition of acceleration as the time rate of change in velocity, we can write that:

$$\dot{x}(n) - \dot{x}(n-1) = p_n \iff \dot{x}(n) = \dot{x}(n-1) + p_n$$

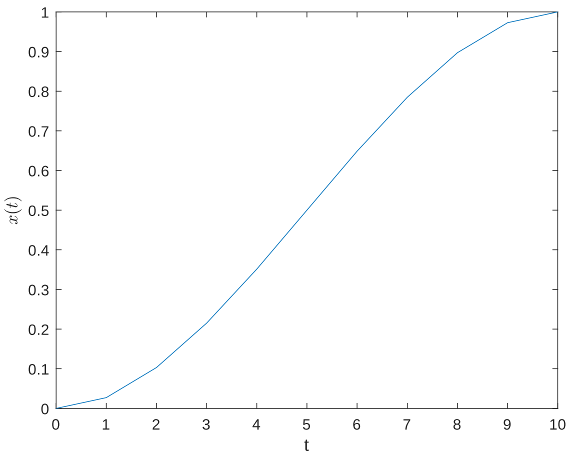
Note that the change in time is 1 by the problem statement. We can then use the definition of position as a time rate of change in velocity, and equating it to an average velocity over that period:

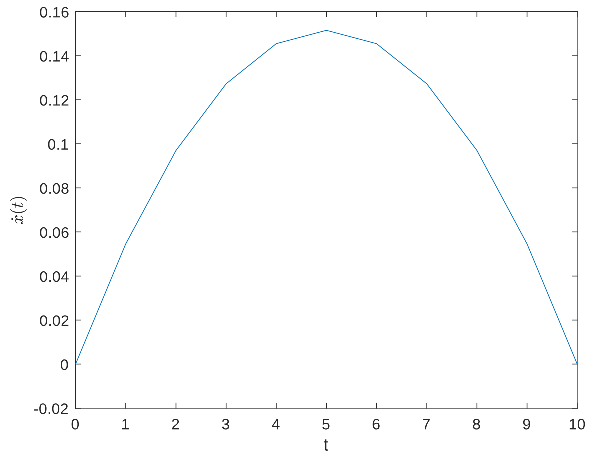
$$\begin{aligned} x(n) - x(n-1) &= \frac{\dot{x}(n) + \dot{x}(n-1)}{2} \\ &= \frac{\dot{x}(n-1) + \dot{x}(n-1) + p_n}{2} \\ &= \dot{x}(n-1) + \frac{p_n}{2} \end{aligned}$$

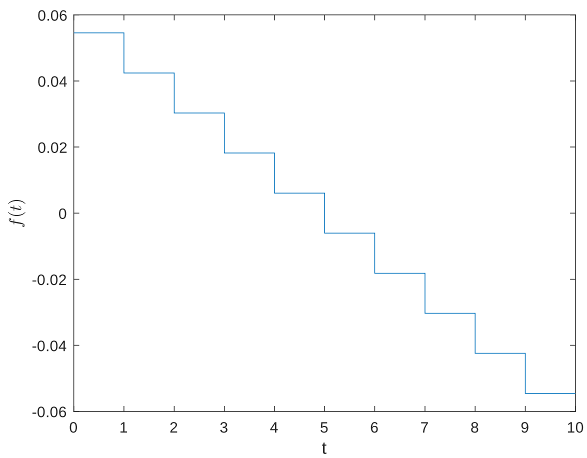
Rearranging we arrive at the first equation:

$$x(n) = x(n-1) + \dot{x}(n-1) + \frac{p_n}{2}$$

b)





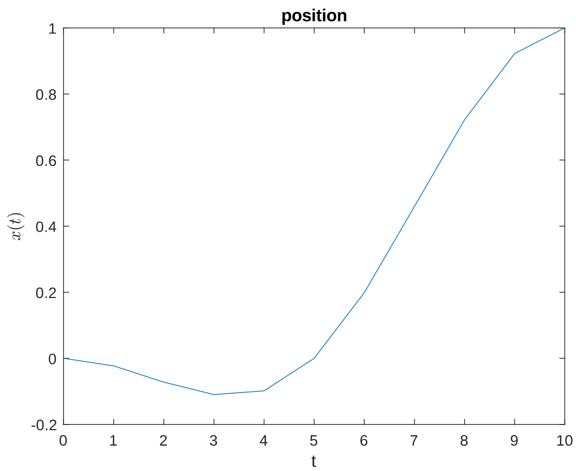


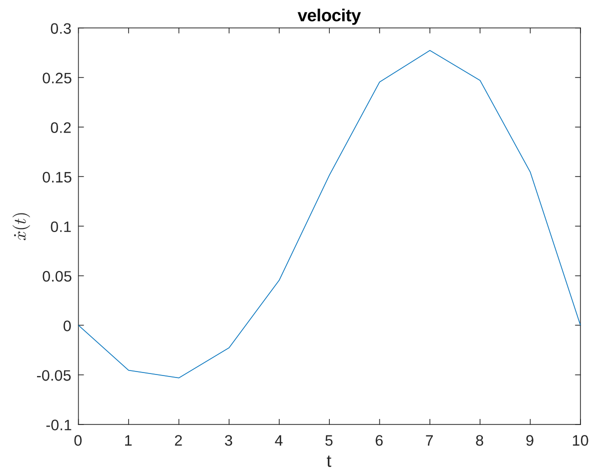
The position increases from zero to 1 in a cubic fashion and satisfies all the conditions given. Since the velocity is parabolic, it makes sense that the position is cubic. The control input is generated so that there are no pulses and instead gradually varies. This is expected from the l2-norm which will penalize bursts.

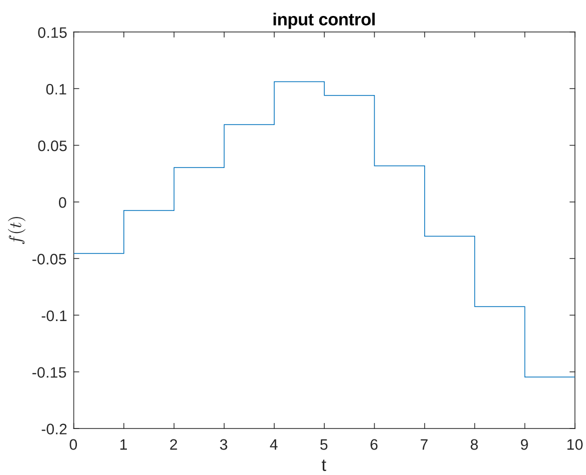
Source (MATLAB):

```
N = 10;
H = zeros(2, N);
b = [0.5; 1];
A = [1 1; 0 1];
yN = [1; 0];

for i = 1:N
    H(:,i) = A^(N-i) * b;
end

u = H' * ((H * H') \ yN);

y = zeros(2, 10);
y(:,1) = b*u(1);
for i = 2:N
    y(:,i) = A * y(:,i-1) + b*u(i);
end
```

c)







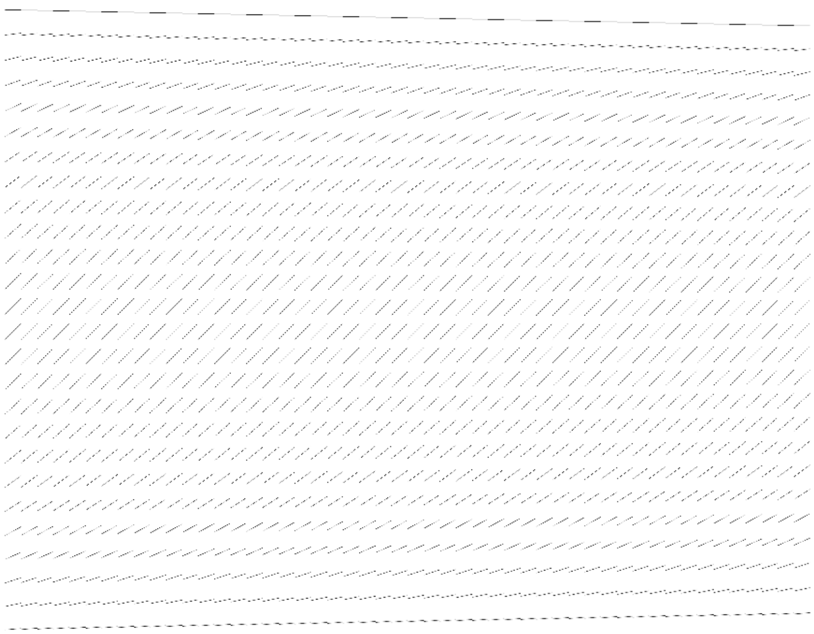
The figures show that the extra condition was met, as x(5) = 0. To do this the control goes negative at the beginning and then comes back up, balancing out the initial inputs. Note that the velocity is not constrained at t=5, so the inputs actually spend more time being positive to allow the position to reach up without large inputs. The velocity follows this trend as expected.

Source (MATLAB):

```
N = 10;
H = zeros(3, N);
b = [0.5; 1];
A = [1 1; 0 1];
yN = [1; 0; 0]; % x(10), v(10), x(5)

for i = 1:N
    H(1:2,i) = A^(N-i) * b;
end
H(3,:) = [9/2 7/2 5/2 3/2 1/2 0 0 0 0 0]; % x(5) = 0 constraint

u = H' * ((H * H') \ yN);

y = zeros(2, 10);
y(:,1) = b*u(1);
for i = 2:N
    y(:,i) = A * y(:,i-1) + b*u(i);
end
```

Question 4.6

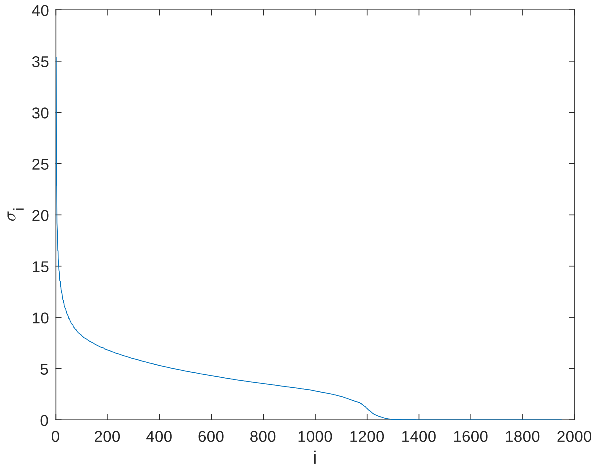
a)



Source (Matlab):

```
h = 50;
w = 50;
n = h * w;
m = 1950;

A = zeros(m, n);
I = eye(n);

for i = 1:n
    A(:,i) = scanImage(reshape(I(i, :), h, w));
end

imshow(-A, [])
```

b) The singular values of the estimated matrix A from part a) is plotted below.



The rank chosen was 1225, at which the singular value is approximately 0.5.

The text in the scanned image is “Winter is Coming”.

Source (MATLAB):

```
Yun = scanImage();
[U,S,V] = svd(A);

title('Singular values of estimated A')
plot(diag(S))
xlabel('i')
ylabel('\sigma_i')

r = 1225;
A_pr = V(:,1:r) * diag(1./diag(S(1:r,1:r))) * U(:,1:r)';

M = reshape(A_pr * Yun, w, h);
imshow(M, [])
```