16-811 Math Fundamentals for Robotics
Assignment 4

Sajal Maheshwari
sajalm@andrew.cmu.edu

October 24, 2019

1 Collaborations

None

2 Instructions for Running code and Analysis

The code for Question 1 is written in MATLAB.
The code for Question 6 is written in Python

2.1 Question 1

For question 1, initially the startPoint, endPoint, stepSize are declared. The other initializations for Adams-Bashforth algorithm are also initialized here.

There are three functions corresponding to the Euler, RK4 and Adams-Bashforth algorithms, by the names of 'Euler', 'RungeKutta' and 'AdamBatch'. The main code is broken into sub-sections where each method is called and the results are both tabulated and visualized graphically.

2.2 Question 6

The code for Question 6 is added in the file q6.py. The loop for iterations is divided into three parts, with each part corresponding to the (a), (b) and (c) sub-parts in the question. To run the code, the evaluator needs to change the number of iterations in the for loop (line 45) and then uncomment/comment the relevant/irrelavant code in the loop.

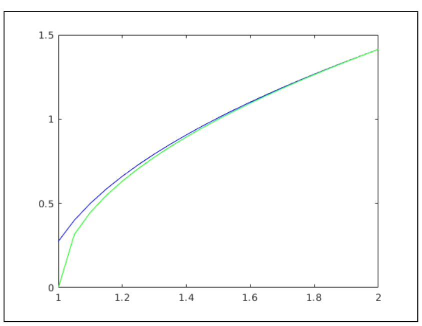
3 Code questions and explanations

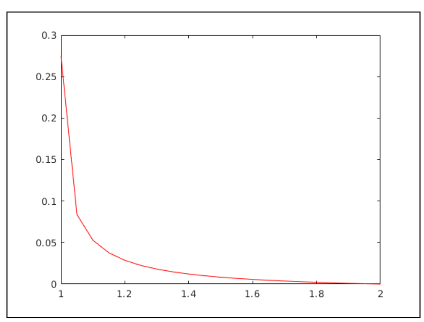
3.1 Q.1

For all the plots, green represents the estimated output and blue represents the real output.

3.1.1 Euler Method

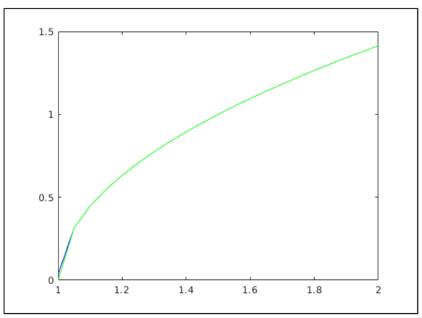
	1 Input	2 EstimatedOutput	3 RealOutput	4 Error	5
1	2	1.4142	1.4142	0	
2	1.9500	1.3789	1.3784	4.5335e-04	
3	1.9000	1.3426	1.3416	9.5555e-04	
4	1.8500	1.3054	1.3038	0.0015	
5	1.8000	1.2671	1.2649	0.0021	
6	1.7500	1.2276	1.2247	0.0028	
7	1.7000	1.1869	1.1832	0.0036	
8	1.6500	1.1447	1.1402	0.0046	
9	1.6000	1.1011	1.0954	0.0056	
10	1.5500	1.0556	1.0488	0.0068	
11	1.5000	1.0083	1	0.0083	
12	1.4500	0.9587	0.9487	0.0100	
13	1.4000	0.9065	0.8944	0.0121	
14	1.3500	0.8514	0.8367	0.0147	
15	1.3000	0.7926	0.7746	0.0181	
16	1.2500	0.7296	0.7071	0.0225	
17	1.2000	0.6610	0.6325	0.0286	
18	1.1500	0.5854	0.5477	0.0377	
19	1.1000	0.5000	0.4472	0.0528	
20	1.0500	0.4000	0.3162	0.0838	
21	1	0.2750	0	0.2750	
22					

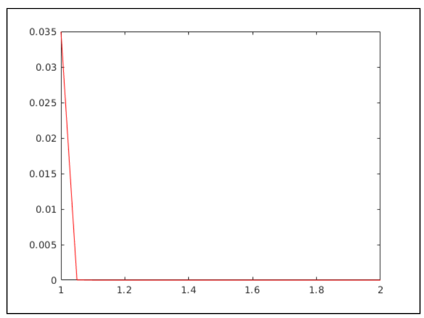




The average error for the Euler method comes out to be: 0.0282

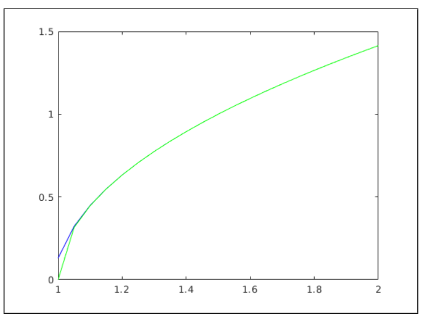
	1 Input	2 EstimatedOutput	3 RealOutput	4 Error	5
1	2	1.4142	1.4142	0	
2	1.9500	1.3784	1.3784	3.1446e-10	
3	1.9000	1.3416	1.3416	7.2106e-10	
4	1.8500	1.3038	1.3038	1.2523e-09	
5	1.8000	1.2649	1.2649	1.9547e-09	
6	1.7500	1.2247	1.2247	2.8968e-09	
7	1.7000	1.1832	1.1832	4.1813e-09	
8	1.6500	1.1402	1.1402	5.9667e-09	
9	1.6000	1.0954	1.0954	8.5051e-09	
10	1.5500	1.0488	1.0488	1.2212e-08	
11	1.5000	1.0000	1	1.7802e-08	
12	1.4500	0.9487	0.9487	2.6563e-08	
13	1.4000	0.8944	0.8944	4.0966e-08	
14	1.3500	0.8367	0.8367	6.6105e-08	
15	1.3000	0.7746	0.7746	1.1350e-07	
16	1.2500	0.7071	0.7071	2.1252e-07	
17	1.2000	0.6325	0.6325	4.5153e-07	
18	1.1500	0.5477	0.5477	1.1674e-06	
19	1.1000	0.4472	0.4472	4.2429e-06	
20	1.0500	0.3162	0.3162	3.1626e-05	
21	1	0.0350	0	0.0350	
22					

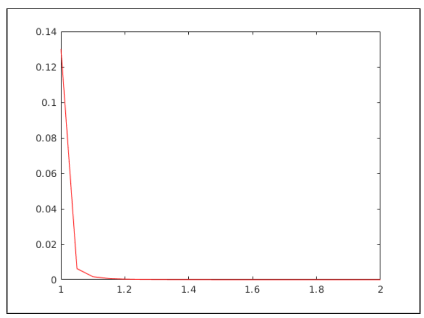




The average error for the RK4 method comes out to be: 0.0017

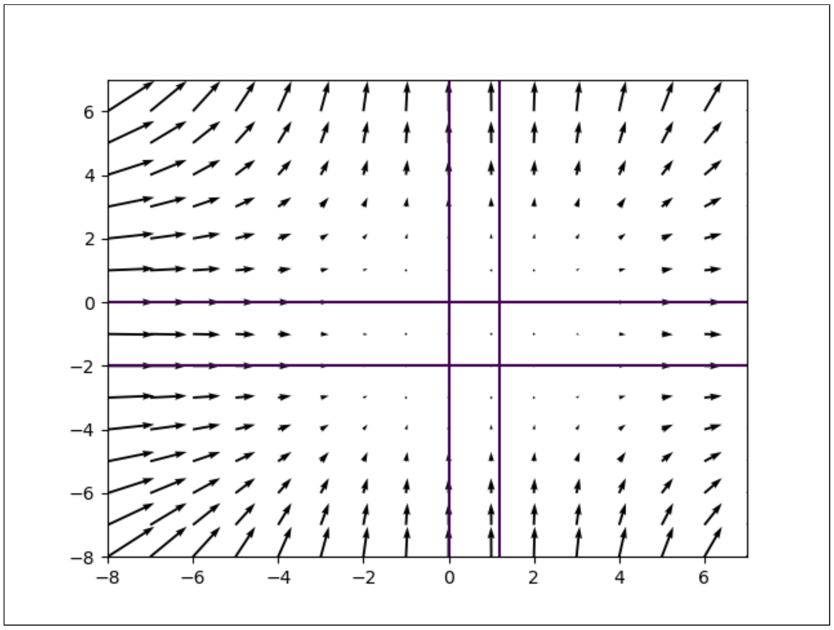
	1 Input	2 EstimatedOutput	3 RealOutput	4 Error	5
1	2	1.4142	1.4142	0	
2	1.9500	1.3784	1.3784	4.0463e-07	
3	1.9000	1.3416	1.3416	9.3316e-07	
4	1.8500	1.3038	1.3038	1.6016e-06	
5	1.8000	1.2649	1.2649	2.4729e-06	
6	1.7500	1.2247	1.2247	3.6192e-06	
7	1.7000	1.1832	1.1832	5.1477e-06	
8	1.6500	1.1402	1.1402	7.2196e-06	
9	1.6000	1.0955	1.0954	1.0082e-05	
10	1.5500	1.0488	1.0488	1.4125e-05	
11	1.5000	1.0000	1	1.9988e-05	
12	1.4500	0.9487	0.9487	2.8759e-05	
13	1.4000	0.8945	0.8944	4.2386e-05	
14	1.3500	0.8367	0.8367	6.4565e-05	
15	1.3000	0.7747	0.7746	1.0283e-04	
16	1.2500	0.7073	0.7071	1.7402e-04	
17	1.2000	0.6328	0.6325	3.2056e-04	
18	1.1500	0.5484	0.5477	6.6866e-04	
19	1.1000	0.4489	0.4472	0.0017	
20	1.0500	0.3225	0.3162	0.0063	
21	1	0.1301	0	0.1301	
22					





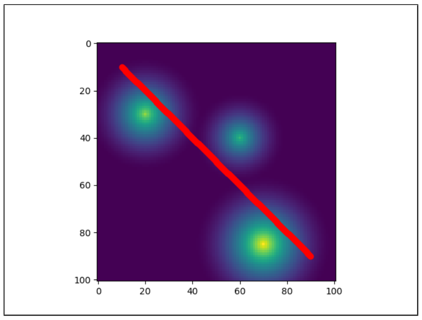
The average error for the Euler method comes out to be: 0.0066

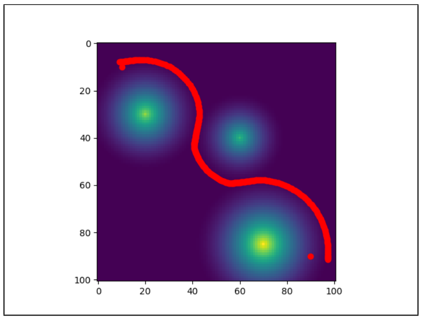
3.2 Q2



3.3 Q6

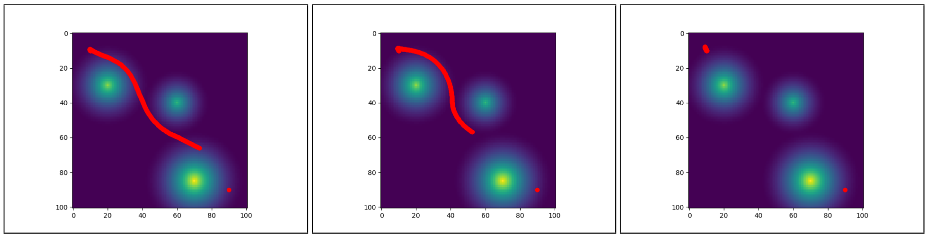
3.3.1 Q6.1





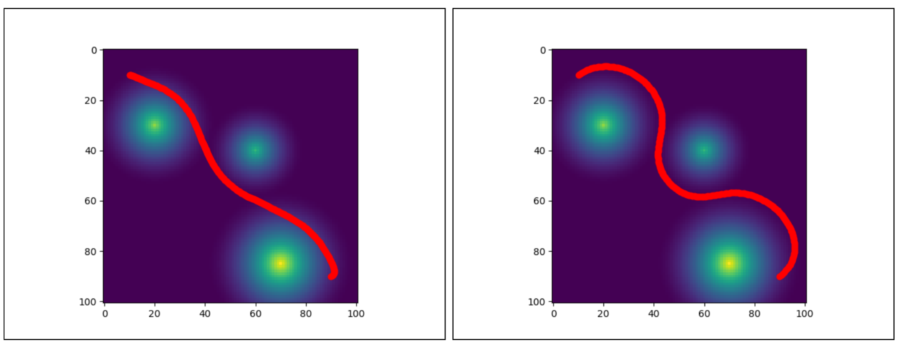
As the number of iterations increased, each individual point reached from its start position to the minimaum position it could reach, and we get a 'discontinous' path from source to destination, but without the obstacles (right graph).

3.3.2 Q6.2



As the number of iterations increased, the points tend to get closer to the point before them, because of the very high weight on smoothness.Since the start and the end points are fixed, all the points tend to get concentrated on one of these two points as the number of iterations increase, which is the reason for failure of this method.(left to right, path points after 100,200 and 500 iterations).

3.3.3 Q6.3



Introducing continuity constraints on both sides means that the points are close to both the next and the previous points. If the points get concentrated on the start and end points, the cost between p_i and p_{i+1}, where p_i converges on the start p_{i+1} converges on the end, will have an extremely high cost. Therefore, the path learned now is optimized such that the points are close(both forward and backward gradients) and not independent of their neighbourhood, giving us a realistic path. The plots show the learned paths after 100 and 5000 iterations in the left and right images respectively.

3.3.4 Q6.4

The final answer will not be the same if we start from different initialization of the path. This is because gradient descent at each of the points can only gaurateee that it will get us till the local minima for that point. Now, we can reach different minima for different initial point positions. For example, if the initial paths given would have been on the different side of the obstacle peaks, we would have got the mirror image of the current path as the optimal path.

3.3.5 Q6.5

We can make use of the concept of interpolation here. First we can do the gradient descent over all the points without the smoothness conditions in place. This will ensure that each point reaches its own minima individually. We can now select the points below a threshold value which we treat as our valid low-cost points. Now, between the valid points that are far away, we can deploy interpolation to get the points in between as a post-processing step, which might increase the overall cost over the entire path because of a 'longer' path consisting of more points, but each individual step has a low cost.

16811 Math fundamental for Robotics Assignment 4

Q.1) (a) $\frac{dy}{dx} = \frac{1}{y}$ $y(2) = \sqrt{2}$

$\int y\,dy = \int dx$

$\frac{y^2}{2} = x + c$

We are given that $y(2) = \sqrt{2}$

$\Rightarrow \frac{2}{2} = 2 + c$

$\Rightarrow c = -1$

$\Rightarrow y^2 = 2x - 2$

$\Rightarrow y = \sqrt{2x-2}$

(b) CODE AND GRAPH SUBMITTED

(c) CODE AND GRAPH SUBMITTED

(d) CODE AND GRAPH SUBMITTED

HW 4

Q.2 (a) $f(x,y) = x^3 + y^3 - 2x^2 + 3y^2 - 8$

$$\frac{\partial f(x,y)}{\partial x} = 3x^2 - 4x$$

extrema $(x=0, \ x=4/3)$

$$\frac{\partial^2 f(x,y)}{\partial x^2} = 6x - 4$$

$$6x-4\big|_{x=0} = -4 \Rightarrow x=0 \rightarrow \text{maxima}$$

$$6x-4\big|_{x=4/3} = 4 \Rightarrow x=4/3 \rightarrow \text{minima}$$

$$\frac{\partial f(x,y)}{\partial y} = 3y^2 + 6y$$

extrema $(y=0, \ y=-2)$

$$\frac{\partial^2 f(x,y)}{\partial y^2} = 6y + 6$$

$$6y+6\big|_{y=0} = 6 \Rightarrow y=0 \rightarrow \text{minima}$$

$$6y+6\big|_{y=-2} = -6 \Rightarrow y=-2 \rightarrow \text{maxima}$$

Critical points for $f(x,y)$ are:

$(0,0) \rightarrow$ Saddle point

$(0,-2) \rightarrow$ maxima

$(4/3, 0) \rightarrow$ minima

$(4/3, -2) \rightarrow$ Saddle point

(b) $\begin{bmatrix} \partial f(x,y)/\partial x \\ \partial f(x,y)/\partial y \end{bmatrix} = \begin{bmatrix} 3x^2 - 4x \\ 3y^2 + 6y \end{bmatrix}$

At $x = 1$ and $y = -1$

$$\begin{bmatrix} 3x^2 - 4x \\ 3y^2 + 6y \end{bmatrix} = \begin{bmatrix} 3 - 4 \\ 3 - 6 \end{bmatrix} = \begin{bmatrix} -1 \\ -3 \end{bmatrix}$$

We need to find a t such that

$f\begin{bmatrix} 1 - t(-1) \\ -1 - t(-3) \end{bmatrix}$ is minimized

$\Rightarrow$ $f(1+t, 3t-1)$ is minimized

$$g(t) = f(1+t, 3t-1) = (1+t)^3 + (3t-1)^3 - 2(1+t)^2 + 3(3t-1)^2 - 8$$

$$\frac{dg(t)}{dt} = 3(1+t)^2 + 9(3t-1)^2 - 4(1+t) + 18(3t-1)$$

$$= 3(1+t)^2 - 4(1+t) + 9(3t-1)^2 + 18(3t-1)$$

$$= (1+t)[3 + 3t - 4] + 9(3t-1)[3t - 1 + 2]$$

$$= (1+t)[3t-1] + 9(3t-1)(3t+1)$$

$$= (3t-1)(1 + t + 9 \times 3t + 9)$$

$$= (3t-1)(28t + 10)$$

$\frac{dg(t)}{dt} = 0$ at $t = 1/3$ and $t = -5/14$

$$\frac{d^2 g(t)}{dt^2} = (3t-1).28 + (28t+10).3$$

$$= 84t - 28 + 84t + 30$$

$$= 168t + 2$$

At $t = 1/3 \rightarrow$ minima

$\Rightarrow$ The desired value of $t = 1/3$

$\Rightarrow$ The next value of $\begin{bmatrix} x \\ y \end{bmatrix}$ becomes $\begin{bmatrix} 1 - (1/3)(-1) \\ -1 - (1/3)(-3) \end{bmatrix}$

$$= \begin{bmatrix} 4/3 \\ 0 \end{bmatrix}$$

Which is a minima

$\therefore$ Steps required $= 1$. Ans

3 (a) Definiⁿ of 2 vectors being Q - orthogonal is :
(v_1 and v_2)

$$v_1^T Q v_2 = 0$$

Now, let us assume the matrix Q has Eigenvectors d_1 and d_2 with Eigenvalues λ_1 and λ_2 ($\lambda_1 \neq \lambda_2$) respectively

$$A = d_1^T Q d_2$$

$$(\because Q d_2 = \lambda_2 d_2)$$

$$A = \lambda_2 \bar{d_1} \cdot \bar{d_2}$$

Now, we know that A is a scaler

$$\Rightarrow A = A^T$$

$$\Rightarrow (d_1^T Q d_2) = d_2^T Q^T d_1$$

$$= d_2^T Q d_1 \quad (\because Q = Q^T)$$

$$= d_2^T \lambda_1 d_1 \quad (\text{Definition of Eigenvector})$$

$$= \lambda_1 \bar{d_1} \cdot \bar{d_2}$$

$$\Rightarrow \lambda_1 \bar{d_1} \cdot \bar{d_2} = \lambda_2 \bar{d_1} \cdot \bar{d_2}$$

$\therefore$ we know that $\lambda_1 \neq \lambda_2$ (Given)

$\bar{d_1} \cdot \bar{d_2} = 0$ Hence, proved.

(b) If we are given two orthogonal basis vectors as Eigenvectors of Q, we know that $\bar{v_i} \cdot \bar{v_j} = 0$

And by previous question, we know that

$$v_i^T Q v_j = \lambda_i \bar{v_i} \cdot \bar{v_j}$$

$\because \bar{v_i} \cdot \bar{v_j} = 0$

the vectors $\bar{v_i}$ and $\bar{v_j}$ are Q - orthogonal.

Q. 4 (a) $\beta_K = \dfrac{g_{K+1}^T Q d_K}{d_K^T Q d_K}$

$$d_{K+1}^T = -g_{K+1}^T + d_K^T \beta_K$$
$$d_{K+1}^T = -g_{K+1}^T + d_K^T \, \frac{g_{K+1}^T Q d_K}{d_K^T Q d_K}$$

Multiplying both sides by $Q d_{K+1}$

$$d_{K+1}^T Q d_{K+1} = -g_{K+1}^T Q d_{K+1} + \frac{g_{K+1}^T Q d_K}{d_K^T Q d_K} \underbrace{d_K^T Q d_{K+1}}_{0} \rightarrow 0$$

$$\Rightarrow d_{K+1}^T Q d_{K+1} = -g_{K+1}^T Q d_{K+1}$$

$\Rightarrow d_K^T Q d_K = -g_K^T Q d_K$. Hence, proved.

Now, we need to find α and β such that they only contain the terms $Q g_K$

$$\alpha = -\frac{g_K^T d_K}{d_K^T Q d_K} = -\frac{g_K^T d_K}{-g_K^T Q d_K} = \frac{g_K^T d_K}{d_K^T Q g_K}$$

IIIly $\beta = \dfrac{g_{K+1}^T Q d_K}{d_K^T Q d_K} = \dfrac{g_{K+1}^T Q d_K}{-d_K^T Q g_K} = \dfrac{d_K^T Q g_{K+1}}{-d_K^T Q g_K}$

Scanned by CamScanner

Q4.(b) $p_k = \nabla f(y_k) = Q y_k + b$

$$= Q(x_k - g_k) + b$$

$$p_k = Q x_k - Q g_k + b$$

$$\Rightarrow Q g_k = Q x_k + b - p_k$$

$$Q g_k = g_k - p_k \quad \text{Proved.}$$

Q4(c) $$x_{k+1} = x_k + \frac{d_k^T g_k}{d_k^T Q g_k} d_k \quad (\text{from } (a))$$

$$= x_k + \frac{d_k^T g_k}{d_k^T (g_k - p_k)} d_k \quad (\text{from } (b))$$

- Now,

$$g_k^T d_k = d_k^T g_k \quad (\text{Scalar value})$$

$$= d_k^T (Q x_k + b)$$

$$= d_k^T (Q(x_0 + \alpha_0 d_0 + \cdots + \alpha_{i-1} d_{i-1}) + b)$$

$$= d_k^T (Q x_0 + b) + \underbrace{\alpha_0 d_k^T Q d_0 + \cdots}_{0}$$

$$= d_k^T g_0$$

Substituting this, we get

$$x_{k+1} = x_k + \frac{d_k^T g_0}{d_k^T g_0 - d_k^T p_k} d_k$$

This method, thus, does not require Hessian and line search.

- Also, any general function can be thought of as quadratic function in a small interval, allowing us to use these formulae.

5) ~~We need~~

Given: Rectangle (assume of dimension x and y

$2x + 2y = P$

Given this constraint, we need to maximize xy

$x + y = P/2$

$\Rightarrow x + y - P/2 = 0$ $\boxed{\begin{array}{l} h(x,y) = x + y - P/2 \\ \text{and } f(x,y) = -xy \end{array}}$

The equation with lagrange multiplier becomes:

$$\overbrace{-xy}^{\text{MINIMIZE}} + \lambda \overbrace{(x + y - P/2)}^{\text{CONSTRAINT}} = 0$$

Since we need to maximize the area, we find the critical points in terms of x, y and λ

$$F = -xy + \lambda(x + y - P/2)$$

$$\frac{\partial F}{\partial x} = -y + \lambda \qquad \frac{\partial F}{\partial y} = -x + \lambda$$

$$\frac{dF}{\partial \lambda} = x + y - P/2$$

$$\nabla F(x,y,\lambda) = \begin{bmatrix} -y + \lambda \\ -x + \lambda \\ x + y - P/2 \end{bmatrix} = \vec{0}$$

$y = +\lambda \qquad x = +\lambda \qquad x + y = P/2$

$\Rightarrow +\lambda + \lambda = P/2$

$\Rightarrow \lambda = +P/4$

$\Rightarrow x = P/4$ and $y = P/4$.

~~Now, regular points will be, the:~~

Scanned by CamScanner

Now, to verify the 2^{nd} order sufficiency

$$\nabla^2 f(x,y) = \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}$$

and $$\nabla^2 h(x,y) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\Rightarrow \nabla^2 f + \lambda^T \nabla^2 h = \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix} = M$$

and $$\nabla h = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

For any vector $\bar{p} = \begin{bmatrix} a \\ b \end{bmatrix}$ $\quad \nabla \bar{h} \cdot \bar{p} = 0$ if $a = -b$.

$\Rightarrow$ if $\nabla h \cdot \bar{p} = 0$ $\quad p = \begin{bmatrix} a \\ -a \end{bmatrix}$

$$p^T M p = [a \ \ -a] \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} a \\ -a \end{bmatrix} = [a \ \ -a] \begin{bmatrix} a \\ -a \end{bmatrix}$$

$$= 2a^2 \geq 0$$

$\Rightarrow$ if $\nabla \bar{h} \cdot \bar{p} = 0$ $\quad p^T M p$ is positive semi-definite.

Proved.