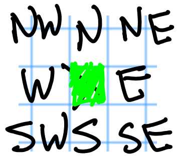
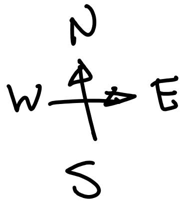
Driving Directions: "8- Connected Neighbors"





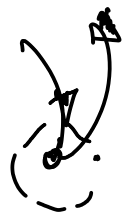


Fig. 4

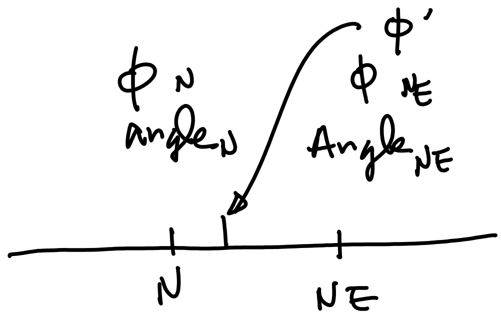
Find the Direction of Driving at Each step that in the end will minize the Objective Function in Eqn (9).

Policy $\pi(\alpha_{k+1}, S_{k+1} | S_k) \to r_{k+1}$

Action (8 Directions)

Reward

Action	Reward
N	$r_N = ?$
NW	$r_{NW} = ?$
W	$r_W = ?$
SW	⋮
S	
⋮	



Determine Reward Function Based ON Moving Direction of Shortest Path.

List of Possible moving Directions

1. From Fig. 4. Only 5 possible Directions

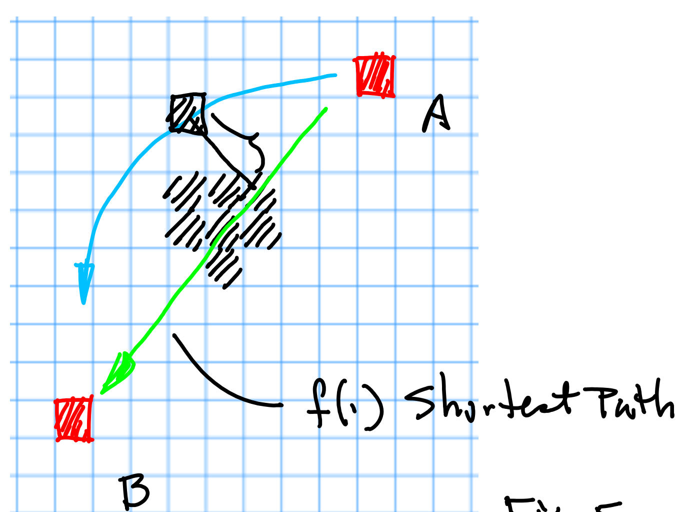


Fig. 5

Example 1. 1. Place 8-Direction Template (N, S, ...) on top of Point A. Use shortest path, Green Line, → as a reference to Define A set of Reward function Based on the direction Matching Reward (DMR) Policy

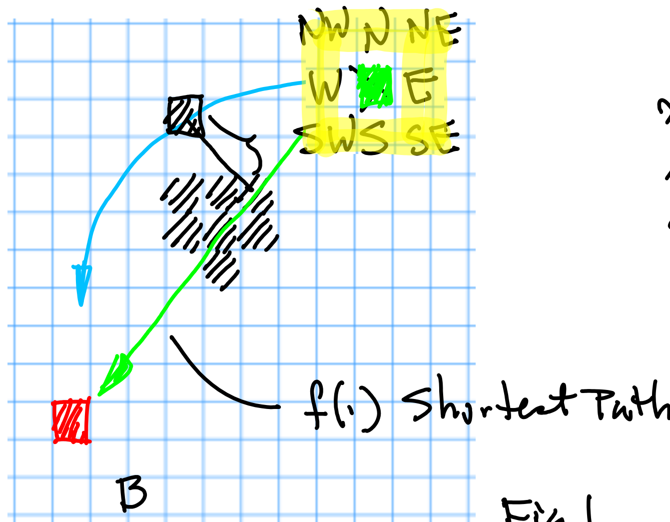


Fig. 6

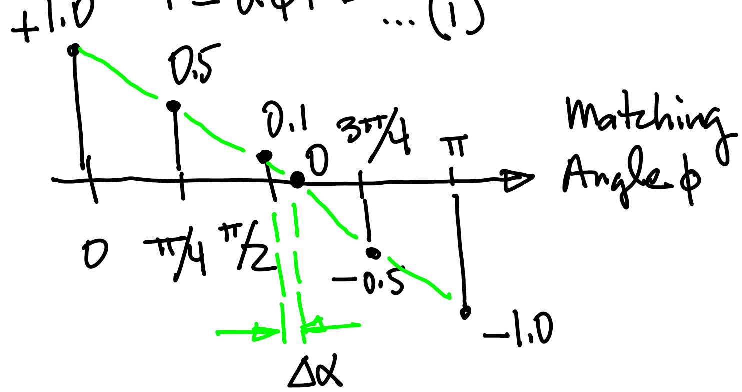
π_{DMR} : $\Delta = 0.5$

x-SW	+1.0	Best Matching	x-SW overlap
x-W	+0.6	Next Best	x-W Angle < $\pi/2$
x-S	+0.6	" "	x-S " < $\pi/2$
x-SE	+0.1		x-SE " < $\pi/2$
x-E	-0.1	Opposite	x-E " > $\pi/2$
x-NW	-0.1		Angle > $\pi/2$
x-N	-0.6		Angle > $3\pi/4$
x-NE	-1.0		Angle $\simeq \pi$

Algorithm: Best Matching Direction. Highest "+" Reward

Worst matching Direction Smallest "−" Reward

$r = a\phi + b$ ... (1)



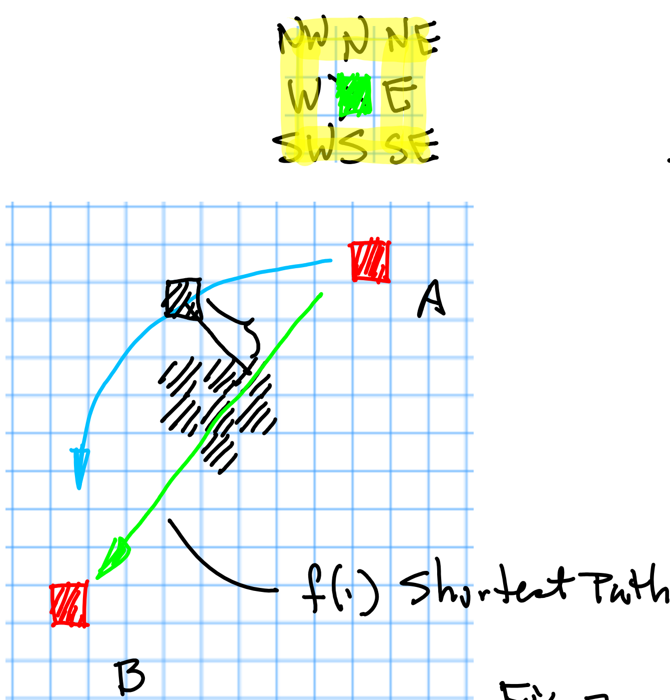


Fig. 7

Program Implementation:

1° Implement Reward Function (1).

Note: Angle ϕ is formed Between Blue line and green Line.

Compute Reward.

March12(Sat), 22

Step1. Setup the System As in Fig 1., Fig.2 is the Abstract Form for Fig.1, So use Fig.2.

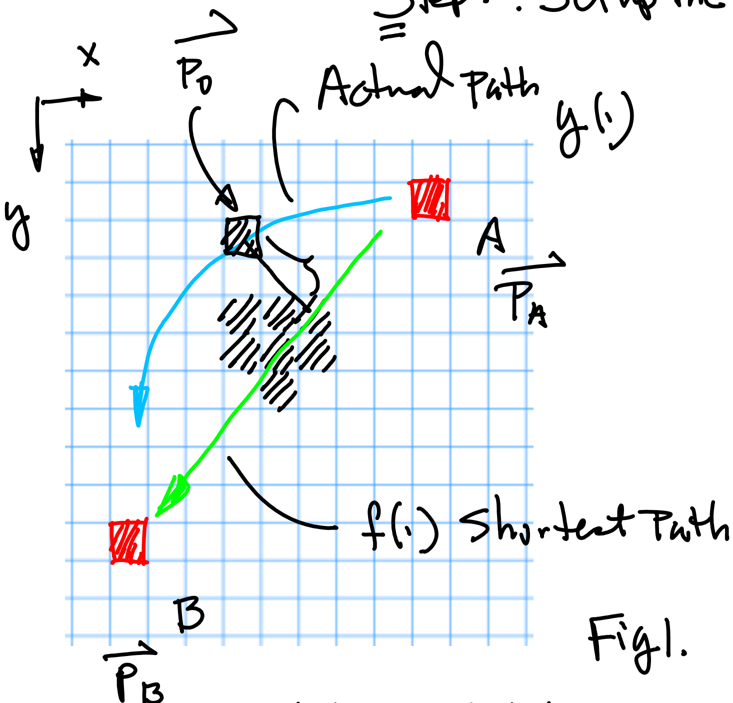


Fig1.

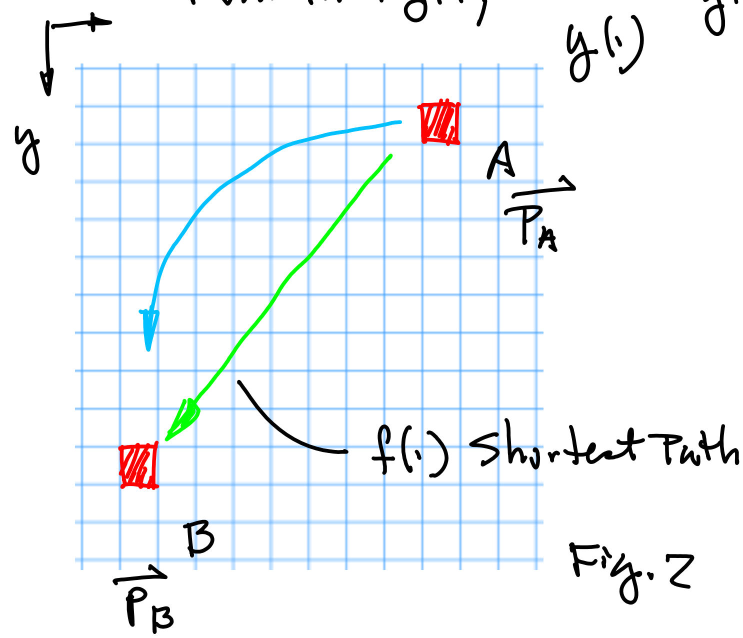
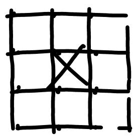


Fig. 2

Step2.

Fig2b.



8-Connected Neighbour.

Define Python module for the Angle Computation for 8-Connected Neighbours in Fig.2b.

Step3. Compute Angle Between green and Blue at location 1.

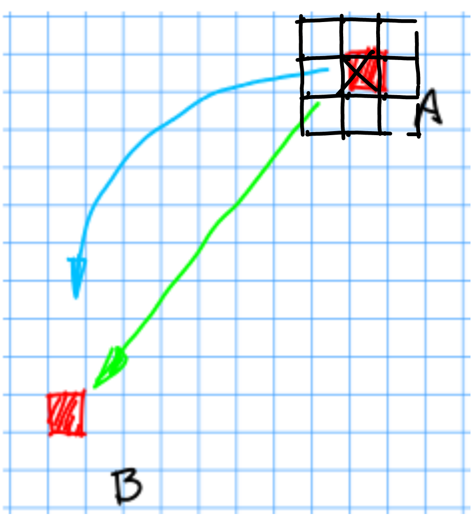


Fig.3

Compute Angle Between green and Blue at location 2.

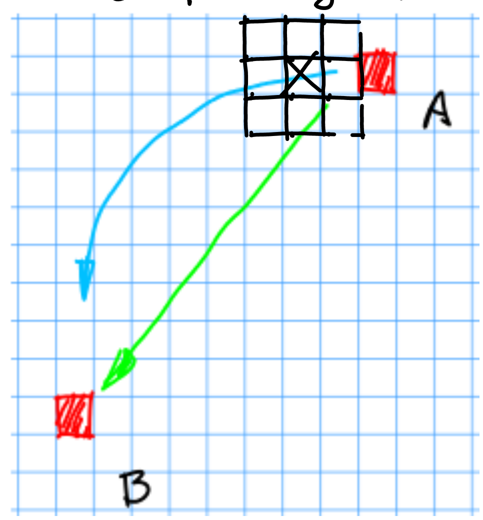


Fig.4

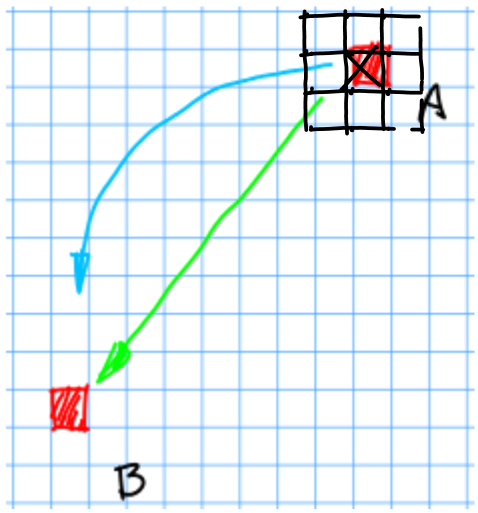


Fig.3

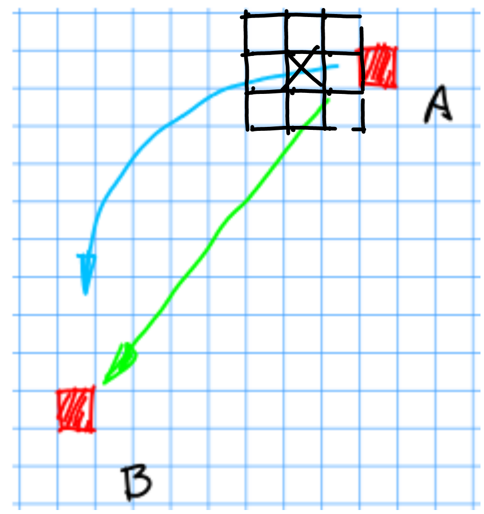


Fig.4

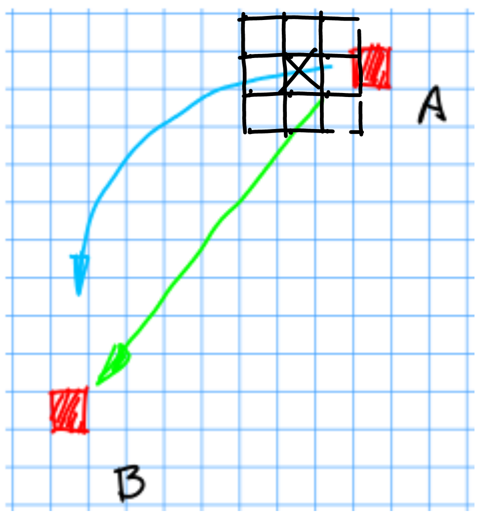


Fig5

Compute Angle Between green and Blue at location3.

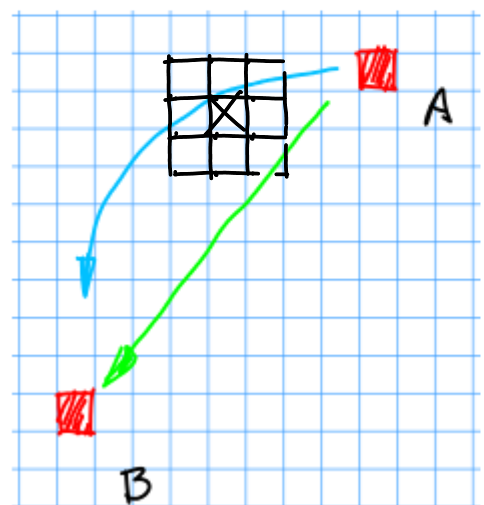


Fig.6

Compute Angle Between green and Blue at location4.

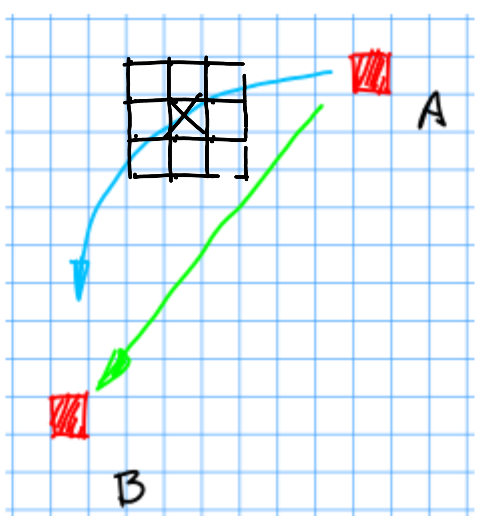


Fig7

Compute Angle Between green and Blue at location5.

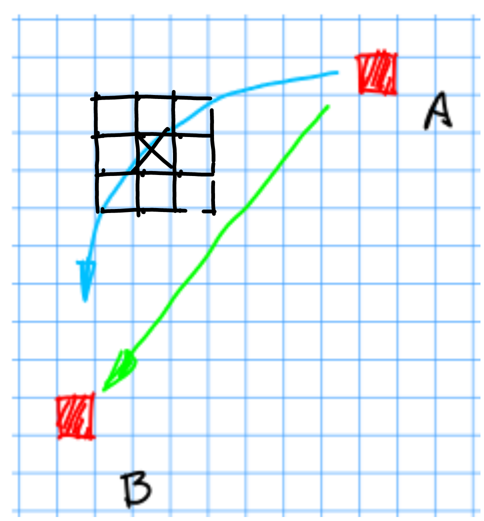


Fig. 8

Compute Angle Between green and Blue at location6.

March 12 (Sat), 22

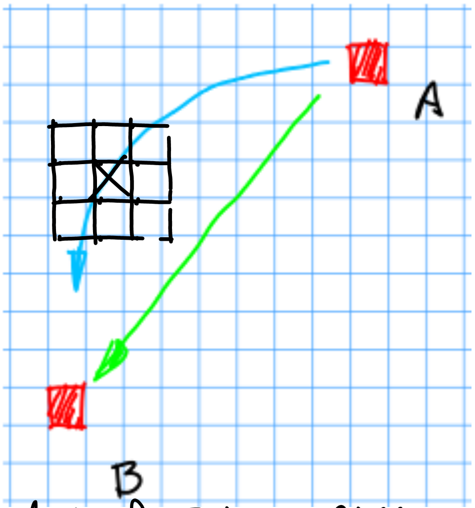


Fig 9

Compute Angle Between green and Blue at location 7.

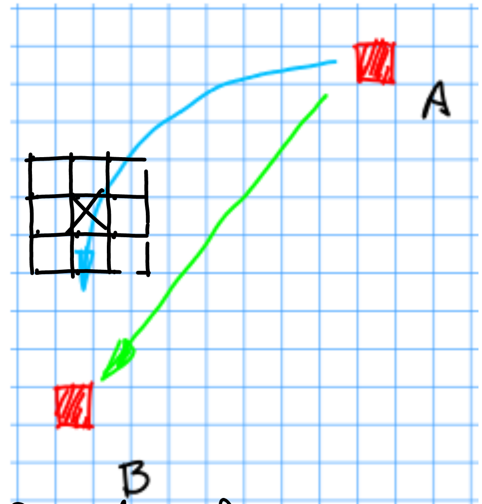


Fig. 10

Compute Angle Between green and Blue at location 8.

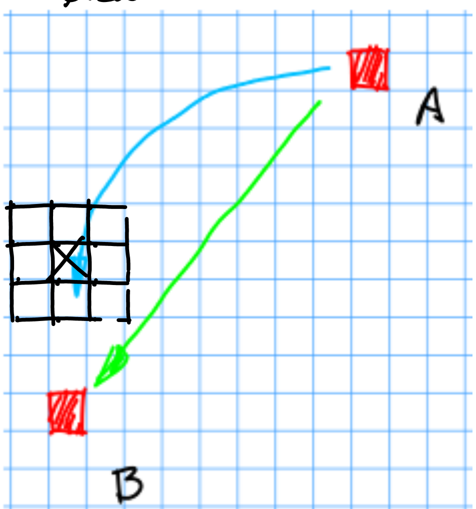


Fig 11

Compute Angle Between green and Blue at location 9.

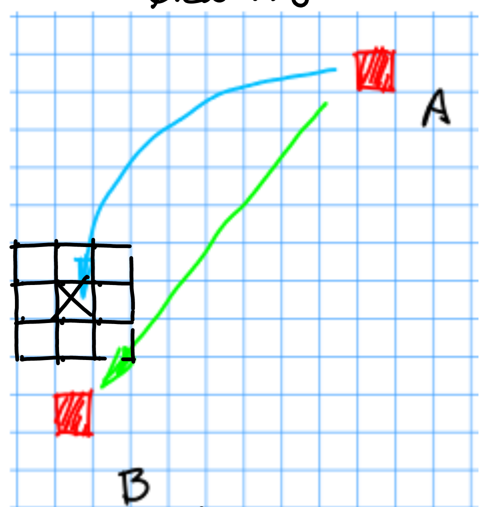


Fig. 12

Compute Angle Between green and Blue at location 10.

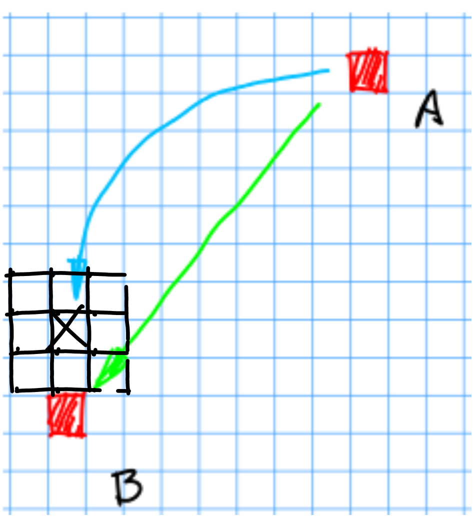


Fig 13

Compute Angle at location 11.

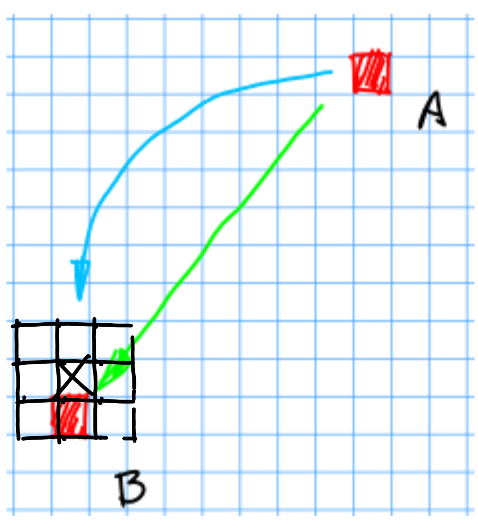


Fig. 14

Compute Angle at location 12.

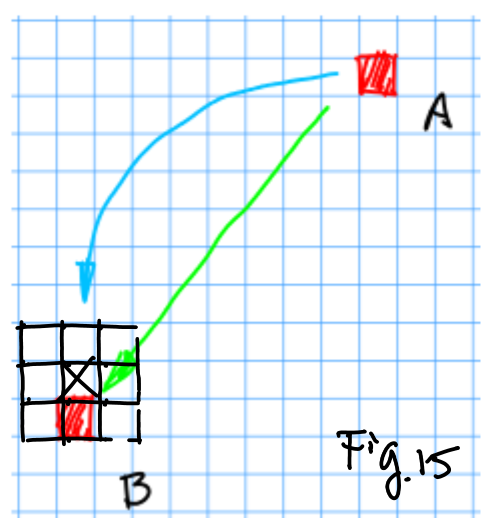
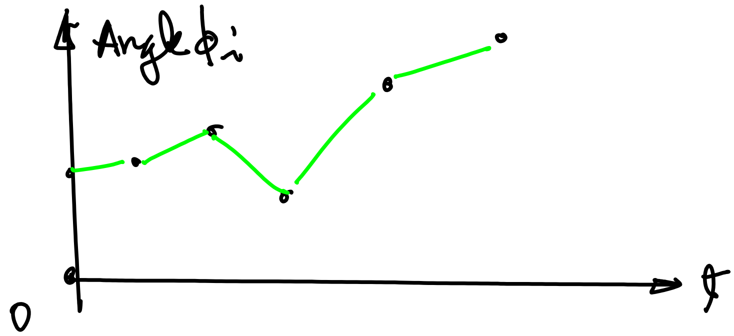


Fig. 15

Compute Angle at location 13.

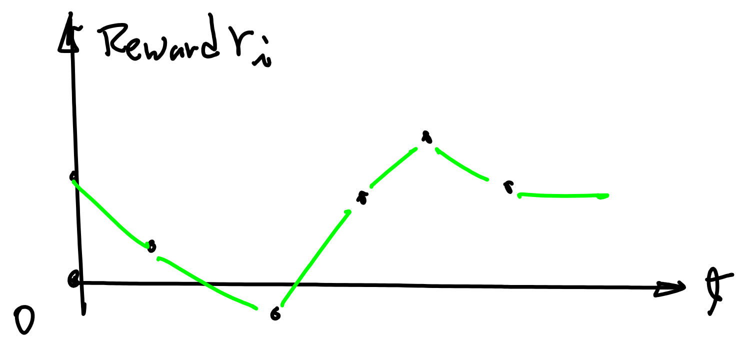
Step 4

Plot All the Angles $\phi_1, \phi_2, \cdots, \phi_i \cdots$ in the plot below, plot All Reward Function Values $r_1, r_2, \cdots, r_i \cdots$ in the plot below.



Then find Sum of all Rewards.

$$R_\Sigma = \sum_{i=1}^{N-1} r_i \quad \cdots (1)$$



(End)