

Driving Directions: "8- Connected Neighbors"

NW N NE
W ■ E
SW S SE

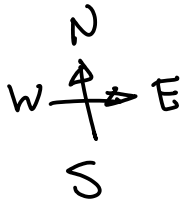
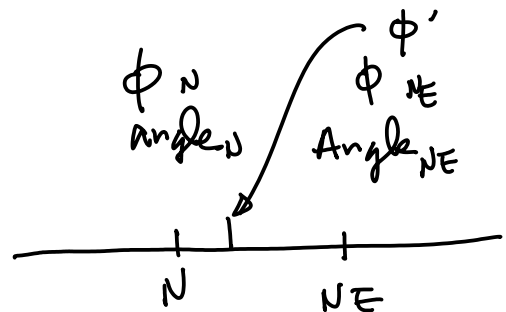


Fig. 4

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

$$\text{Policy } \pi(\underset{\substack{\uparrow \\ \text{Action} \\ (\text{8 Directions})}}{\alpha_{k+1}}, S_{k+1} | S_k) \rightarrow \underset{\substack{\uparrow \\ \text{Reward}}}{\delta_{k+1}}$$

Action	Reward
N	$\delta_N = ?$
NW	$\delta_{NW} = ?$
W	$\delta_W = ?$
SW	\vdots
S	\vdots



NW N NE
W ■ E
SW S SE

Determine Reward Function Based On Moving Direction of Shortest Path.

List of Possible moving Directions

1. From Fig. 4. Only 5 possible Directions

NW N NE
W ■ E
SW S SE

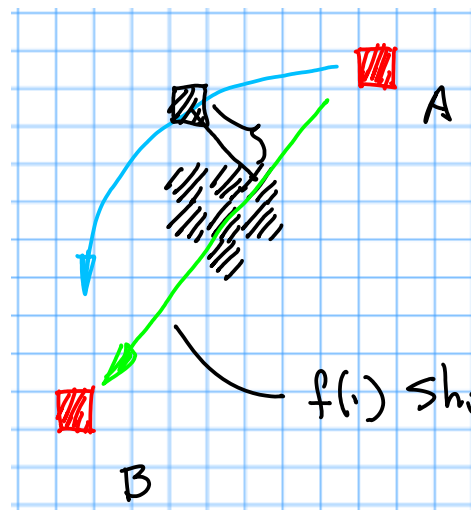
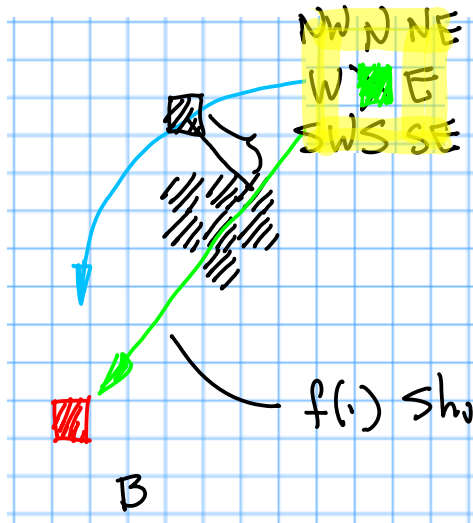


Fig. 5

Example 1. 1. plane 8-Directional 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



$f(\cdot)$ Shortest Path

Fig. 6

$$\pi_{DMR} : \Delta = 0.5$$

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

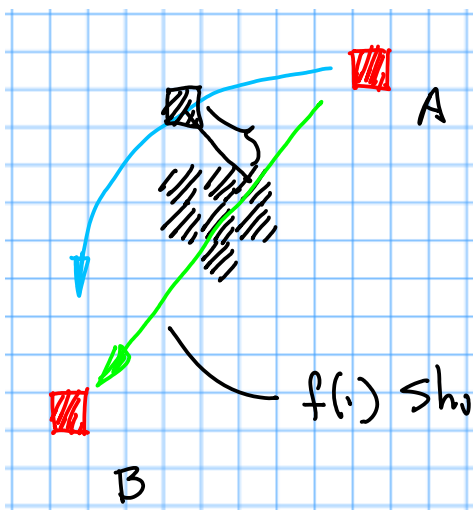
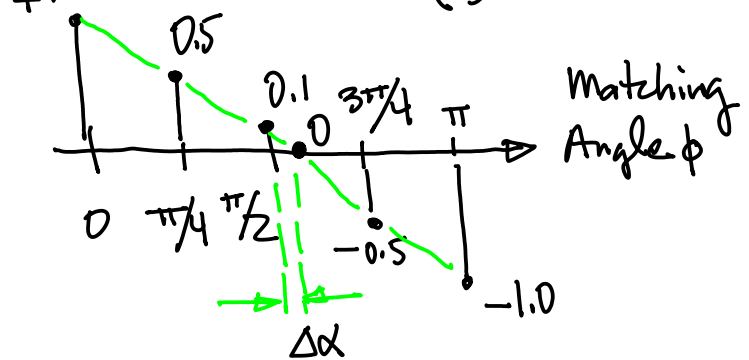
Angle $\approx \pi$

Algorithm: Best Matching Direction. Highest "+" Reward

Worst matching Direction

Smallest "-" Reward

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



$f(\cdot)$ Shortest Path

Fig. 7

Program Implementation:

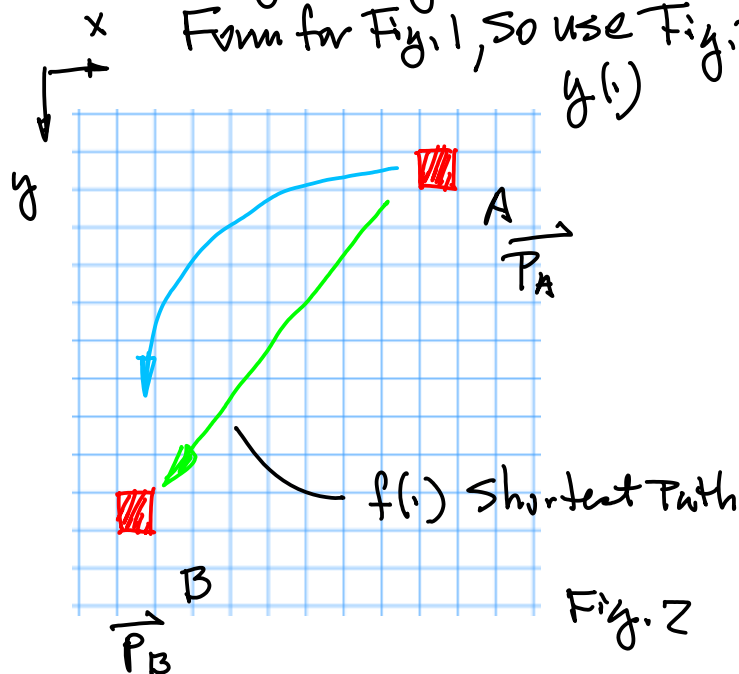
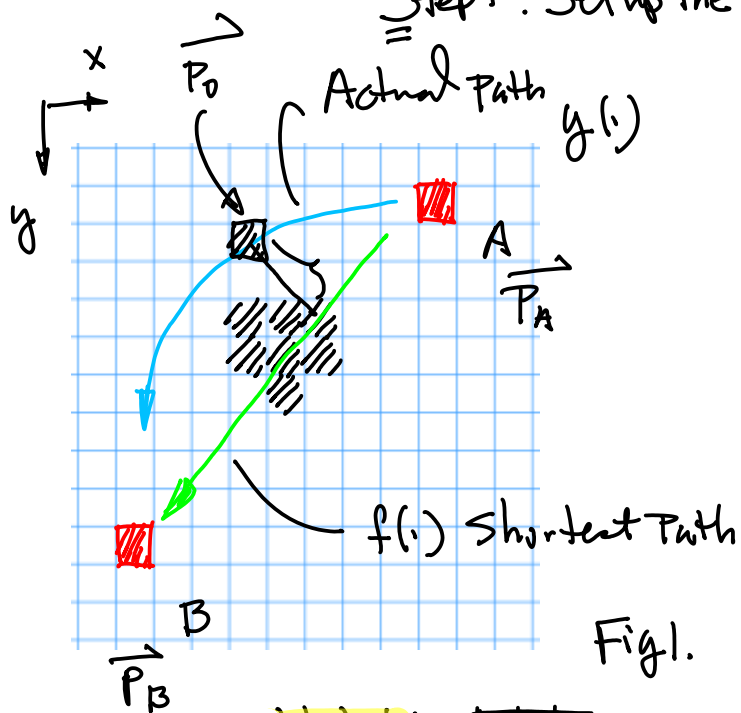
1° Implement Reward Function (1).

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

Compute Reward.

March 12 (Sat), 22

Step 1. Setup the System AS in Fig. 1, Fig. 2 is the Abstract Form for Fig. 1, So use Fig. 2.



Step 2.

NW N NE
W X E
SW S SE

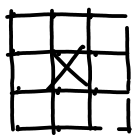
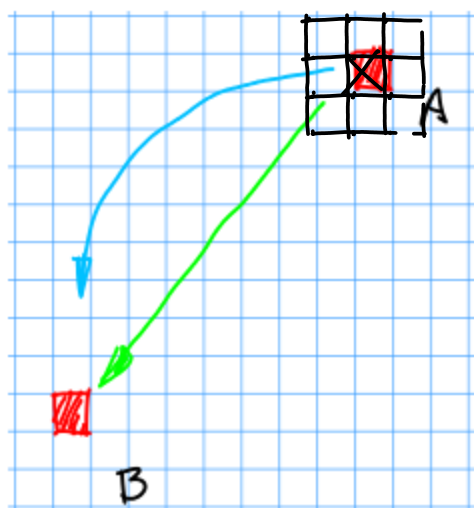


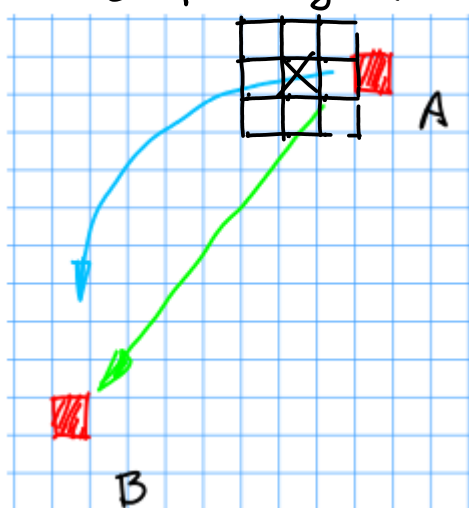
Fig. 2b. 8-Connected Neighbour.

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

Step 3. Compute Angle Between Green and Blue at Location 1.



Compute Angle Between Green and Blue at Location 2.



March 12 (Sat) 22

4

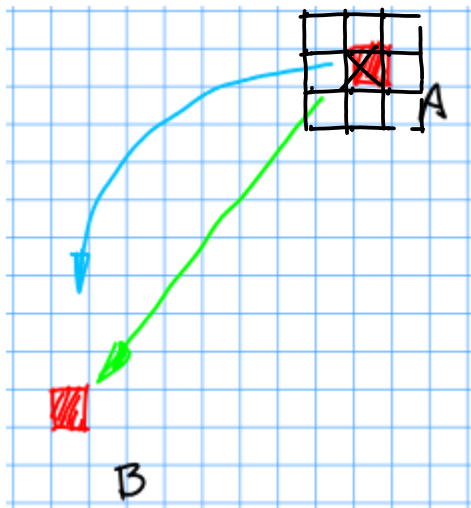


Fig. 3

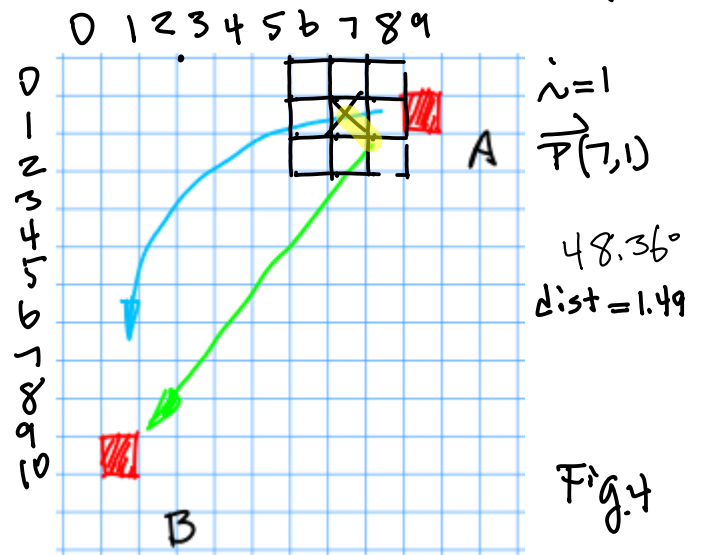


Fig. 4

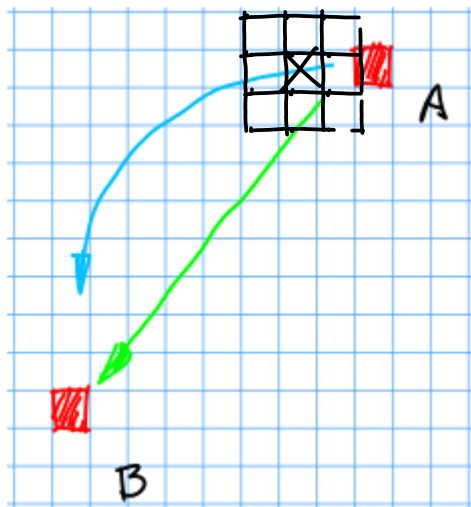


Fig. 5

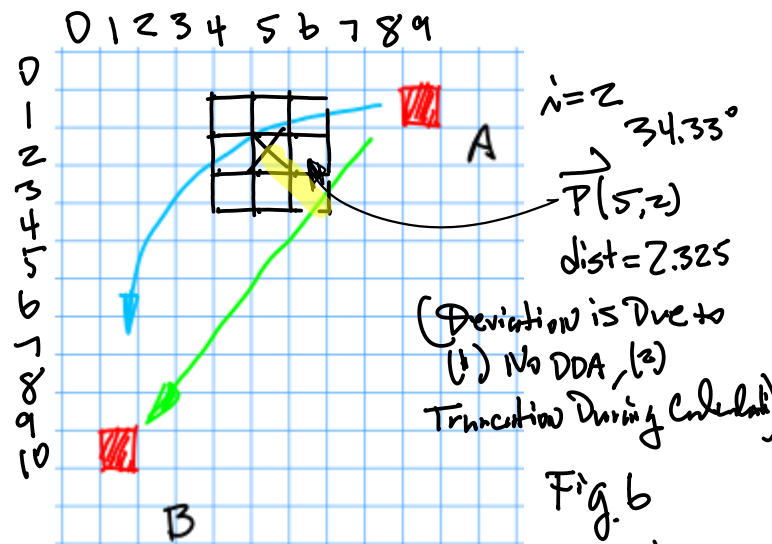


Fig. 6

Compute Angle Between Green and Blue at Location 3.

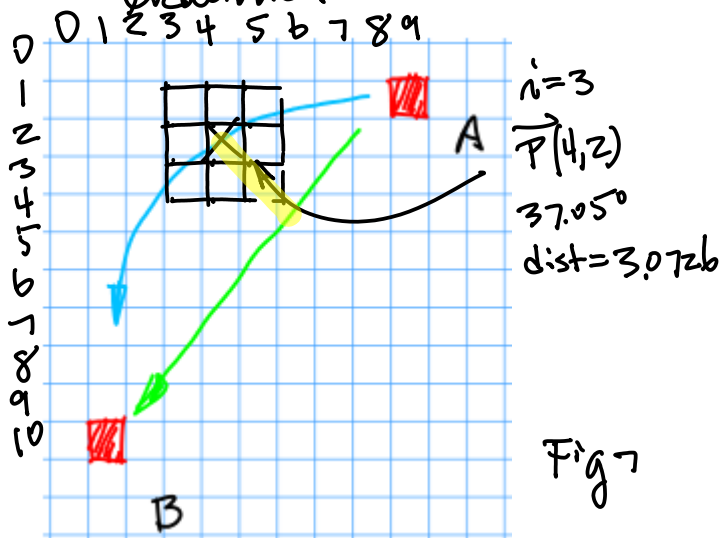


Fig. 7

Compute Angle Between Green and Blue at Location 4.

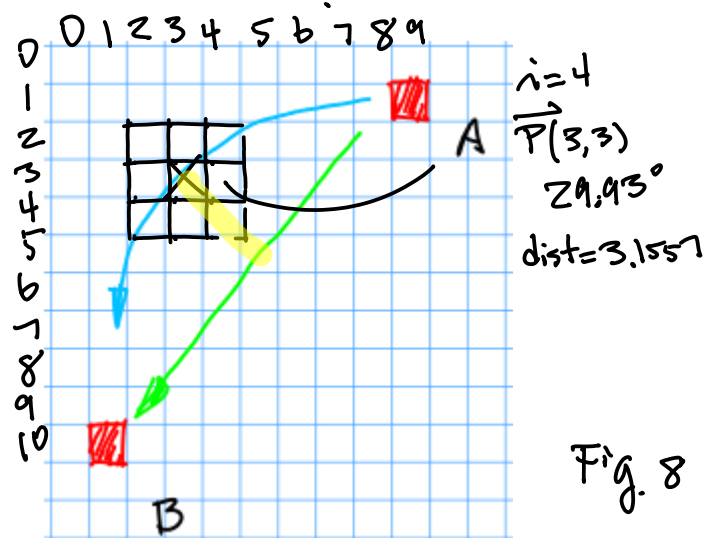


Fig. 8

Compute Angle Between Green and Blue at Location 5.

Compute Angle Between Green and Blue at Location 6.

March 12 (Sat) 22

5

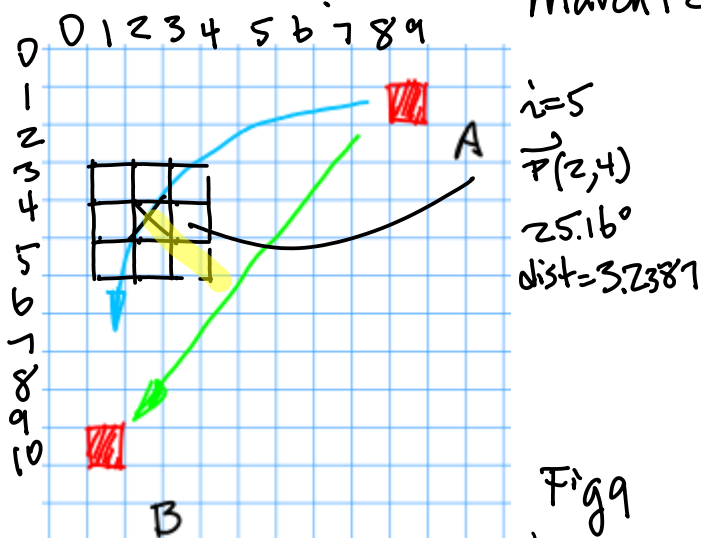


Fig 9

Compute Angle Between green and Blue at Location 7.

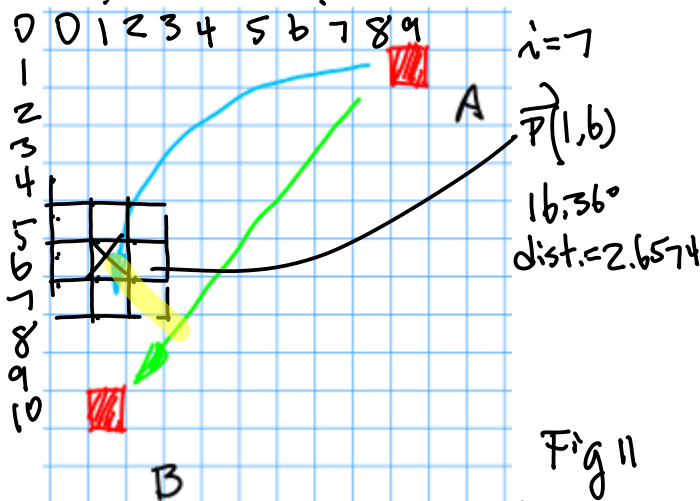


Fig 11

Compute Angle Between green and Blue at Location 9.

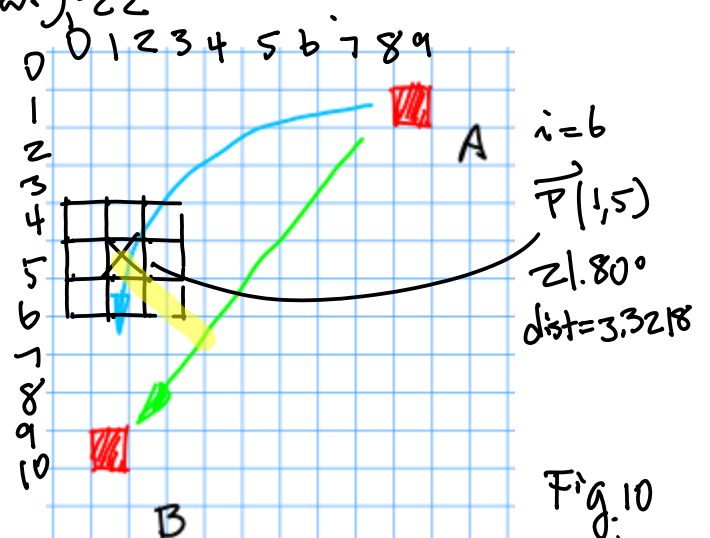


Fig 10

Compute Angle Between green and Blue at Location 8.

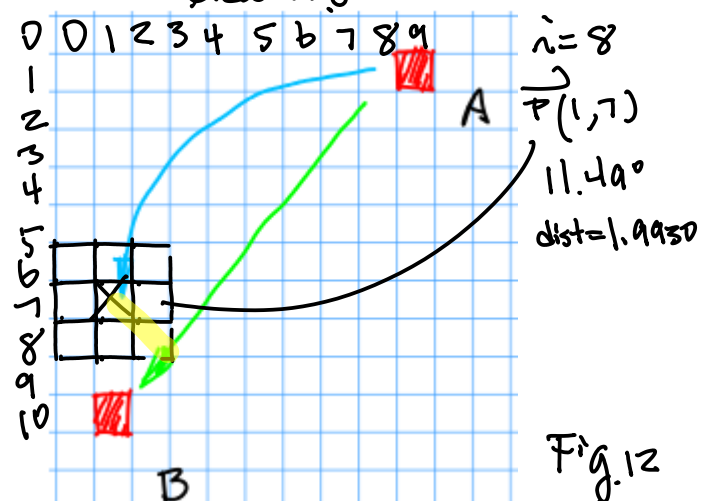


Fig 12

Compute Angle Between green and Blue at Location 10.

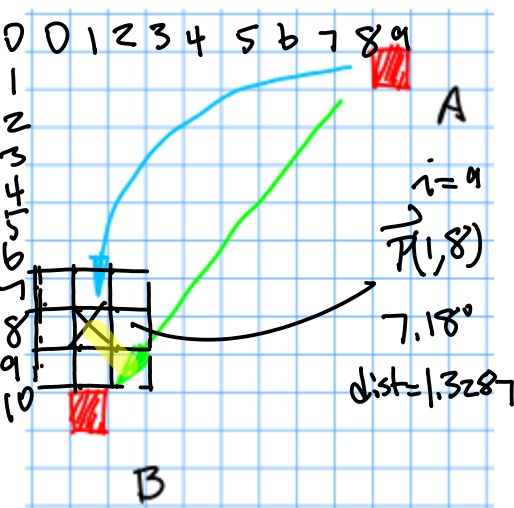


Fig 13

Compute Angle at Location 11.

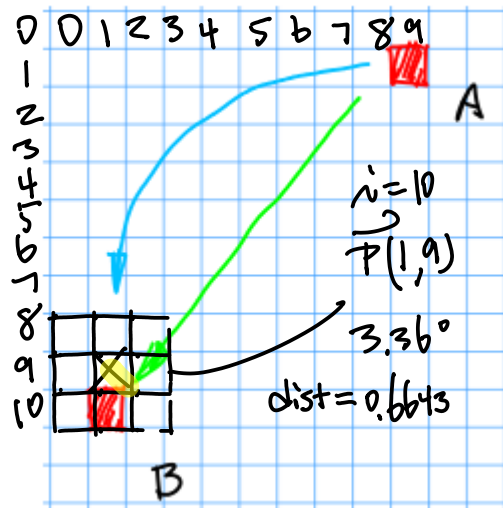


Fig 14

Compute Angle at Location 12.

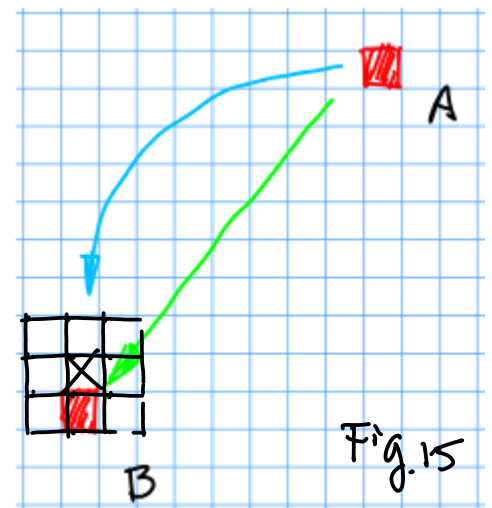


Fig 15

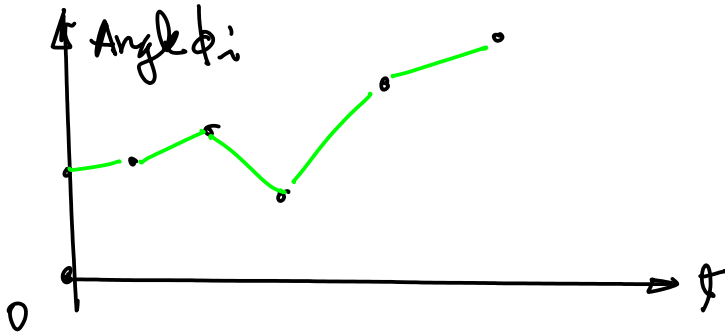
Compute Angle at Location 13.

March 12 (Sat), 22

6

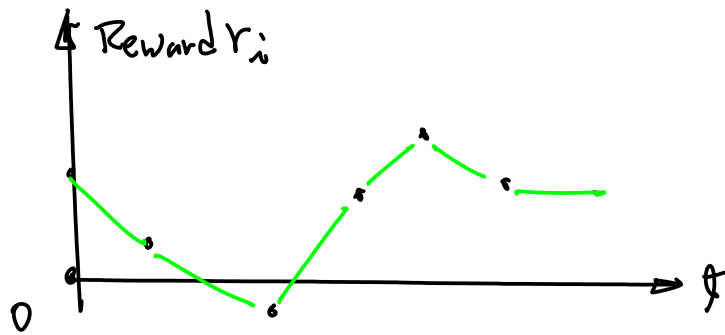
Step 4

Plot All the Angles $\phi_1, \phi_2, \dots, \phi_i, \dots$ in the plot below, plot All Reward Function values $r_1, r_2, \dots, r_i, \dots$ in the plot below. Then find Sum of all Rewards.



Then find Sum of all Rewards.

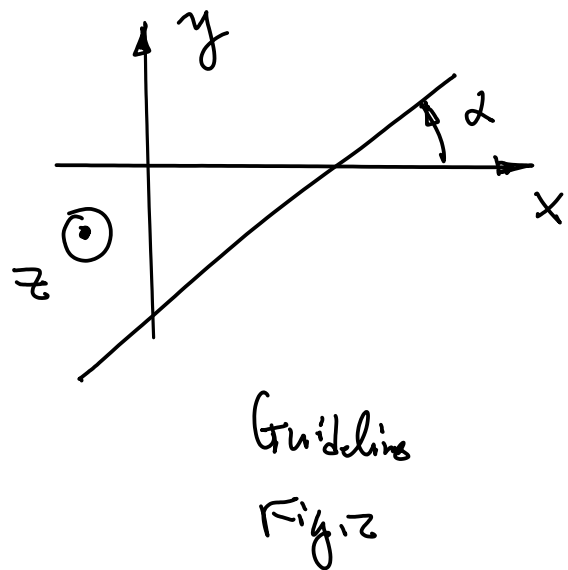
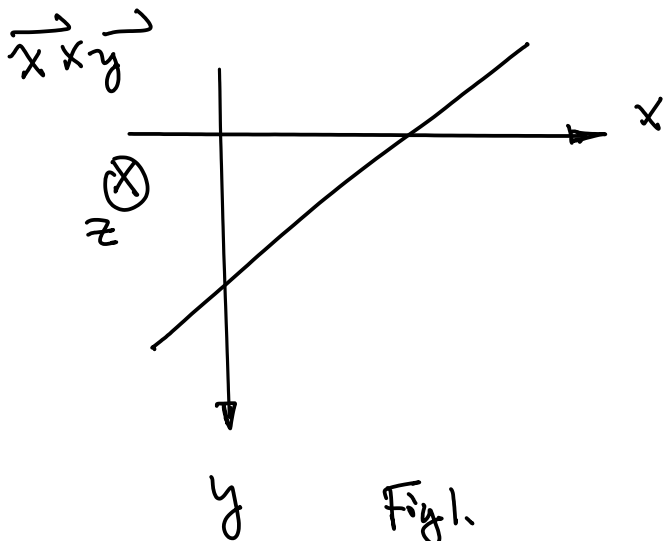
$$R_{\Sigma} = \sum_{i=1}^{N-1} r_i \quad \dots (1)$$



March 13th (Sun) with B.P.

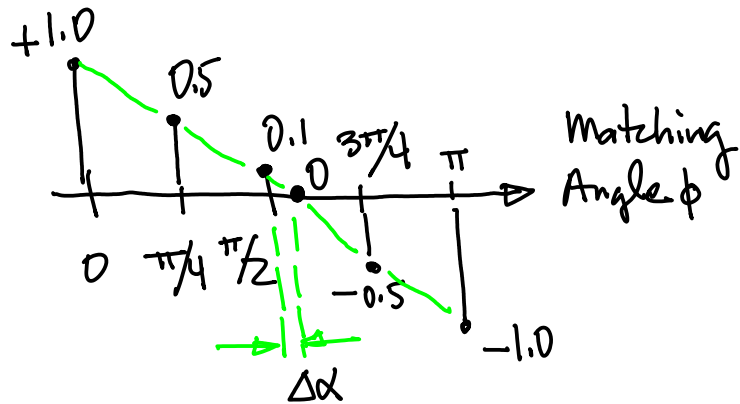
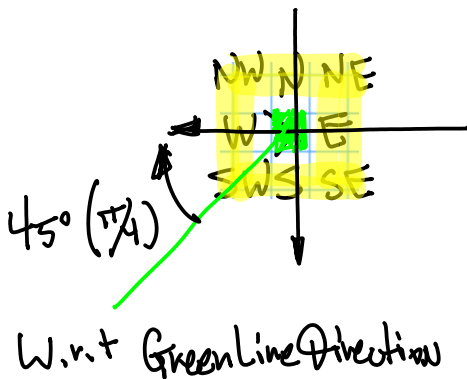
(End)

$$\vec{x} \times \vec{y} = \vec{z}$$



No "World coordinate System" yet

7.



$$\pi_{DMR} : \Delta = 0.5$$

$x-SW$ $+1.0$ 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
 $x-N$ -0.6
 $x-NE$ -1.0

Angle $> \pi/2$
 Angle $> 3\pi/4$
 Angle $\approx \pi$

March 14 (Monday) With. XY, BP.

[robotics-open_abb / aiv200 / 190g-deep-reinforcement-learning / 190g-3-6DoF-Action-State-Reward-SS-2021-03-17.pdf](#)

6 DoF Robot Unity

How to train your Robot Arm?. Training a 6 axis robot arm using Unity... | by Raju K | XRPractices | Medium

[rkandas/RobotArmMLAgentUnity: Training 6 axis robot arm Inverse kinematics using Unity ML Agents \(github.com\)](#)

1. Actions: An array of actions – each action in the array represents the degree of rotation. We have 5 types of actions in total: 1 Rotate and 4 Bends.

1.1. Axis 1: is the bottom-most axis and can rotate 0 to 360 degrees [Rotate]

```
armAxes[0].transform.localRotation =
Quaternion.AngleAxis(angles[0] * 180f, armAxes[0].GetComponent<Axis>().rotationAxis);
```

1.1. Axis 1: is the bottom-most axis and can rotate 0 to 360 degrees [Rotate]

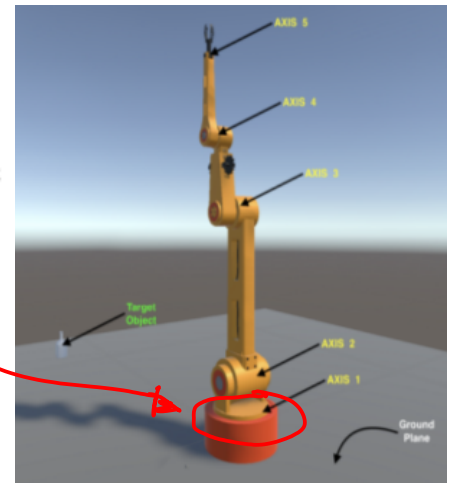
armAxes[0].transform.localRotation = Quaternion.AngleAxis(angles[0] * 180f, armAxes[0].GetComponent<Axis>().rotationAxis);

C#

- a. Physical model (Dimension)
- b. Physics of the model.
- c. Graphical model

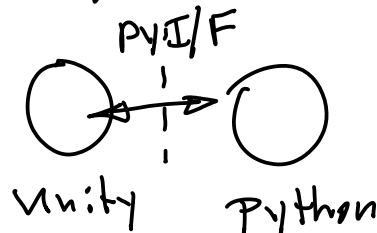
Rotation Direction

Angle



Step 1. a-c : Move OpenCV model to Unity
 Step 2. C# ML Interface

from CTI ONE model, And the implementation code is from CTI One team, especially from Mr. Yusuke Yakuma.



1. Angle
 2. Index
 3. Reward (I)

March 16 (Wed)

1. Verification of YY's Implementation

2. Providet Hand Calculation.

(1) from start position to the end position.

	Position	Angle	Distance	Reward
\vec{P}_A	$(q, 1)$	ϕ_i	d_i	$r(\phi_i, L_i)$
\vdots	\vdots	\vdots	\vdots	\vdots
\vec{P}_i	(i, z)			
\vdots	\vdots			
\vec{P}_B	$(1, 10)$			

(2) Record Heuristic Motion, Path.txt

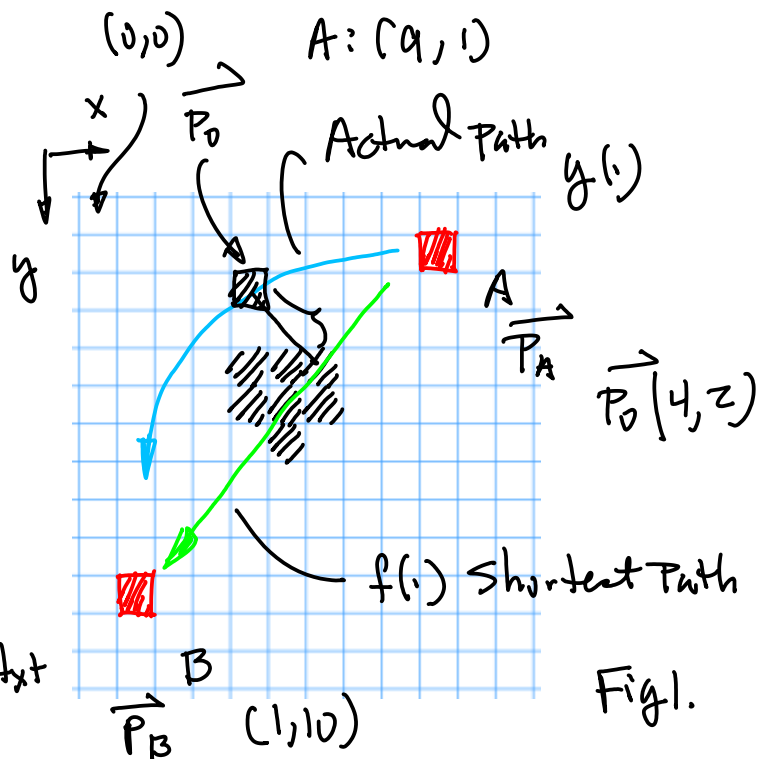


Fig1.

② Run G1 ONE Version 0.1 Code,
Make Comparison for Verification.

March 17 (Thursday) Hand Calculation

Step 1. Given initial condition
 $A = (9, 1), B = (1, 0)$
x-y coordinate system Setup
as shown in the figure below.

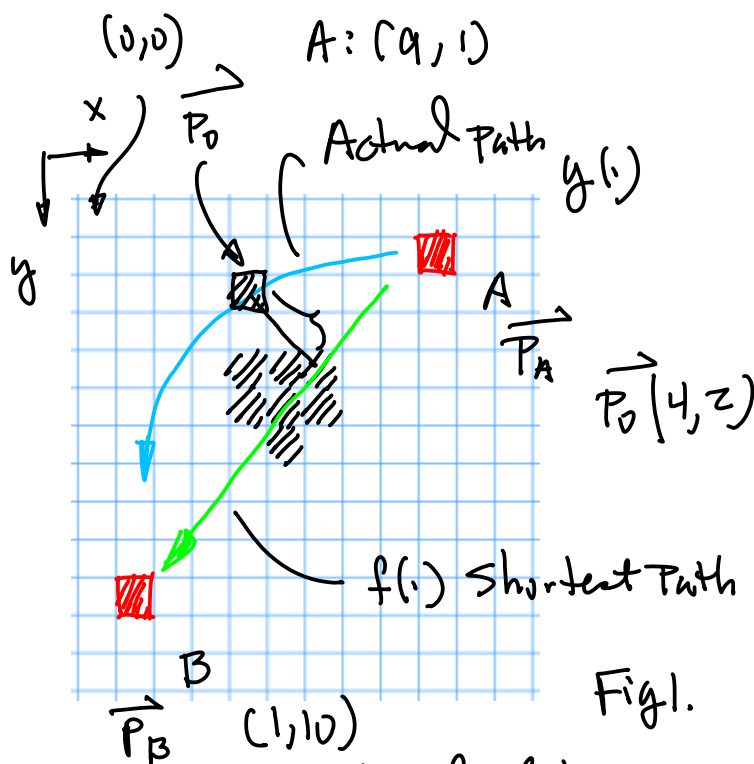


Fig 1.

Step 2. The Angle calculation

Formula:

$$\vec{P_A} - \vec{P_i} = (x_a - x_i, y_a - y_i) \dots (1) \text{ (Blue Line)}$$

$$\vec{P_A} - \vec{P_B} = (x_a - x_b, y_a - y_b) \dots (2) \text{ (Green Line)}$$

$$(\vec{P_A} - \vec{P_i}) \cdot (\vec{P_A} - \vec{P_B}) = \|\vec{P_A} - \vec{P_i}\| \|\vec{P_A} - \vec{P_B}\| \cos \alpha$$

Therefore, for Eqn (3), we have \dots (3)

Eqn (4) below,

$$\cos \alpha = \frac{(\vec{P_A} - \vec{P_i}) \cdot (\vec{P_A} - \vec{P_B})}{\|\vec{P_A} - \vec{P_i}\| \|\vec{P_A} - \vec{P_B}\|} = \frac{(x_a - x_i, y_a - y_i) \cdot (x_a - x_b, y_a - y_b)}{\sqrt{(x_a - x_i)^2 + (y_a - y_i)^2} \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}} = \frac{(x_a - x_i)(x_a - x_b) + (y_a - y_i)(y_a - y_b)}{\sqrt{(x_a - x_i)^2 + (y_a - y_i)^2} \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}} \dots (4)$$

Now, Calculation. Denote Robot Position

as $\vec{P}_i(x_i, y_i)$,

for Position $i=1$, $\vec{P}_1(7,1)$ illustrated in Fig.4

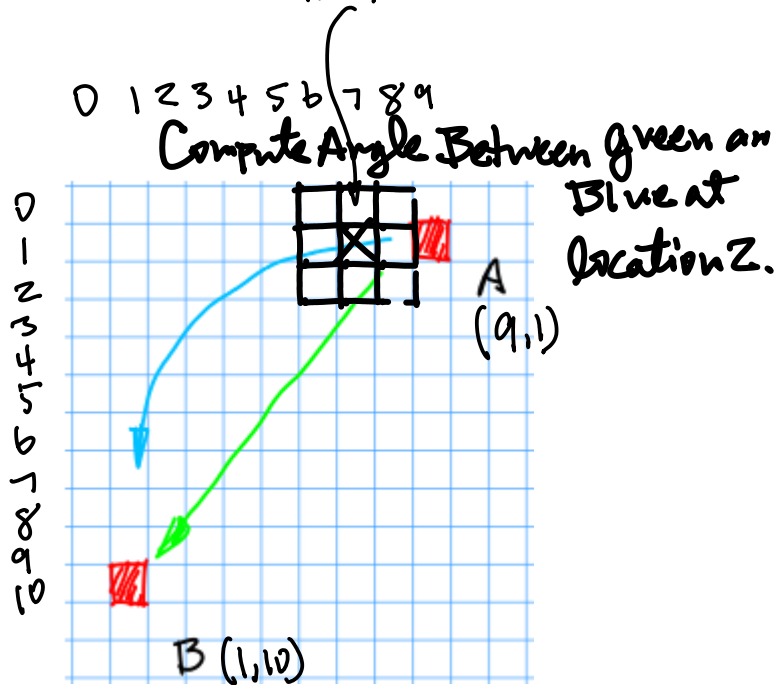


Fig.4

From Eqn (4), we have

$$\cos \theta = \frac{(x_a - x_i)(x_a - x_b) + (y_a - y_i)(y_a - y_b)}{\sqrt{(x_a - x_i)^2 + (y_a - y_i)^2} \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}}$$

see my Spreadsheet for the hand Calculation.

Now, Compute Reward Function, Based on the Formula on PP.

From pp.4.

March 17 (Th), 22

 $\pi_{DMR} : \Delta = 0.5$

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 $\approx \pi$ Algorithm: Best Matching Direction.
Highest "+" RewardWorst matching Direction
Smallest "-" Reward

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

March 18, Fri

Reward Function On Positive Angle (Clockwise) Fig.1

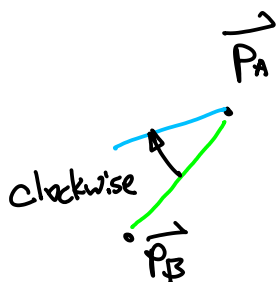


Fig.1

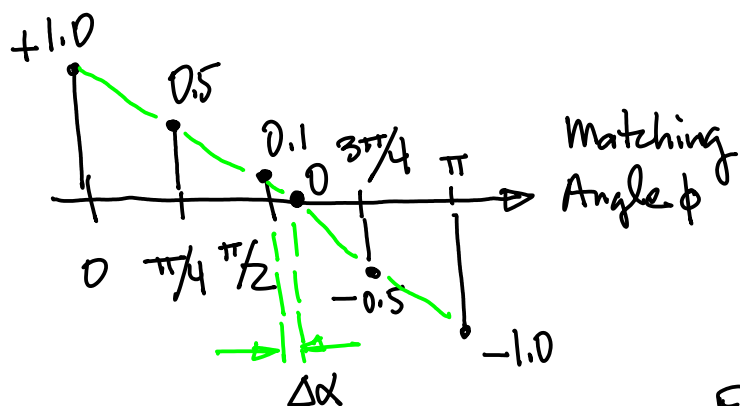
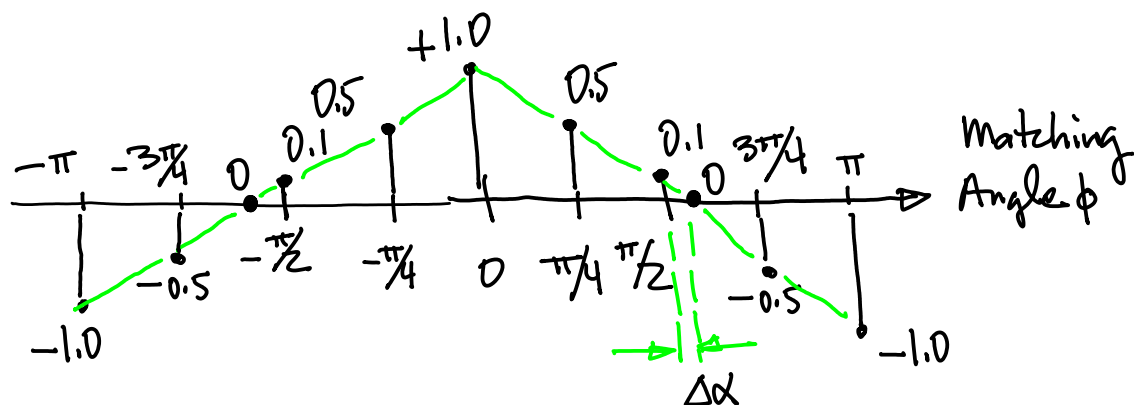
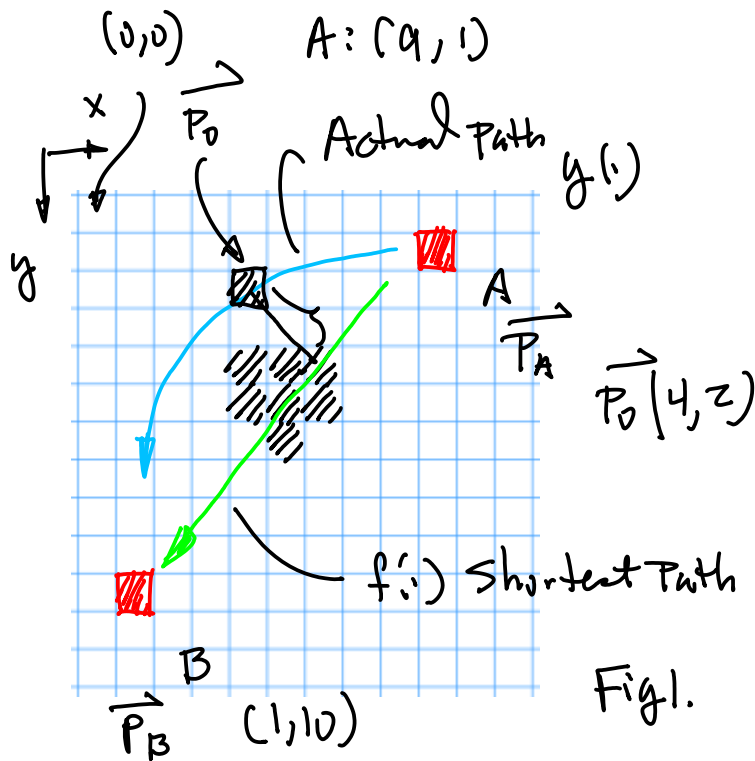


Fig.2



Perpendicular Line Calculation.



Given :

$$\vec{P_A} - \vec{P_i} = (x_a - x_i, y_a - y_i) \dots (1) \text{ (Blue Line)}$$

$$\vec{P_A} - \vec{P_B} = (x_a - x_b, y_a - y_b) \dots (2) \text{ (Green Line)}$$

Direction Vectors:

$$\vec{d_B} = (x_a - x_i, y_a - y_i) = (x_B, y_B) \dots (3)$$

$$\vec{d_G} = (x_a - x_b, y_a - y_b) = (x_G, y_G) \dots (4)$$

Parametric Equation for the lines in

$ax = by + c$ Form,

$$f_B(x, y) = a_B y - (b_B x + c) \dots (5)$$

$$f_G(x, y) = a_G y - (b_G x + c) \dots (6)$$

Where

$$a_B = x_a - x_i, b_B = y_a - y_i \dots (7)$$

$$a_G = x_a - x_b, b_G = y_a - y_b \dots (8)$$

$$\text{And } \vec{d_G} = (a_G, b_G) \dots (8b)$$

The intersecting Point of the Perpendicular line from $\vec{P_i}$ on Blue to Green Line

$$\vec{P}(x, y) = \vec{P_i} + \lambda \vec{d} \dots (9)$$

Where

$$\vec{d} \cdot \vec{d_G} = 0, (x_d, y_d) \cdot (x_G, y_G) = x_d x_G + y_d y_G = 0 \dots (10)$$

$$(x_d, y_d) = (-y_G, x_G) \dots (11)$$

So, Perpendicular line to Green

$$\vec{P}(x, y) = \vec{P_i} + \lambda \vec{d} = \vec{P_i} + \lambda (x_d, y_d) = \vec{P_i} + \lambda (-y_G, x_G) \dots (11b)$$

The distance (Perpendicular Line from Blue Line $\vec{P_i}$ to Green Line):

$$\lambda = - \frac{f_G(x_i, y_i)}{a_G^2 + b_G^2} \dots (12)$$

The distance (Perpendicular Line from Blue Line \vec{P}_i to Green Line):

$$\begin{aligned}\vec{P}(x, y) &= \vec{P}_i + \lambda \vec{d} = \vec{P}_i + \lambda (x_d, y_d) = \vec{P}_i + \lambda (-y_d, x_d) \\ &= (x_i, y_i) + \lambda (-y_d, x_d) \quad \dots (11b)\end{aligned}$$

where

$$\lambda = -\frac{f(x_i, y_i)}{a_g^2 + b_g^2} \quad \dots (12)$$

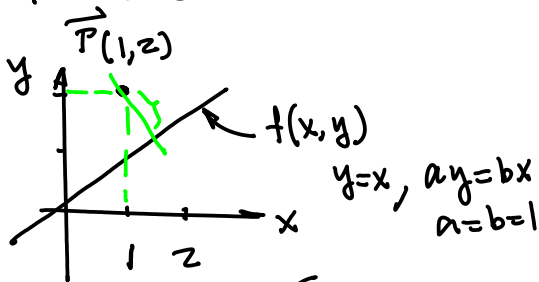
$$f_g(x, y) = a_g y - (b_g x + c_g) \quad \dots (1b)$$

$$a_g = x_a - x_b, \quad b_g = y_a - y_b \quad \dots (8)$$

$$\vec{d}_g = (a_g, b_g) = (x_a, y_a) \quad \dots (8b)$$

$$\text{Then, } \text{dist}(\vec{P}_i, \text{green}) = \sqrt{(x_i - x)^2 + (y_i - y)^2} \quad \dots (13)$$

Testing of the Above Equations By a Special case (for simplicity) Below



Step 1.

Fig 2

For line f (Green Line Equivalent)

$$\vec{d}_g = (x_g, y_g) = (a_g, b_g) = (1, 1)$$

from Eqn (12)

$$\begin{aligned}\lambda &= -\frac{f(x_i, y_i)}{a_g^2 + b_g^2} = -\frac{a_g y_i - (b_g x_i + c_g)}{a_g^2 + b_g^2} = -\frac{y_i - x_i}{1+1} \Big|_{\substack{x_i=1 \\ y_i=2}} \\ &= -\frac{2-1}{2} = -\frac{1}{2}\end{aligned}$$

Step 2. Find Intersection Point of the Perpendicular line on Green From Eqn (11b)

$$\begin{aligned}\vec{P}(x, y) &= (x_i, y_i) + \lambda (-y_d, x_d) \\ &= (1, 2) + \lambda (-1, 1) \Big|_{a_g=b_g=1} \\ &= (1, 2) + \lambda (-1, 1) \Big|_{\lambda=-\frac{1}{2}} \\ &= (1, 2) - \frac{1}{2}(-1, 1) = (1+\frac{1}{2}, 2-\frac{1}{2}) \\ &= (\frac{3}{2}, \frac{3}{2}) = (1.5, 1.5) \quad \dots\end{aligned}$$

From Eqn (10),

$$\begin{aligned}\text{dist}(\vec{P}_i, \text{green}) &= \sqrt{(x_i - x)^2 + (y_i - y)^2} \\ &= \sqrt{(1 - 1.5)^2 + (2 - 1.5)^2} \\ &= \sqrt{0.5^2 + 0.5^2} = 0.5\sqrt{2} = \frac{\sqrt{2}}{2}\end{aligned}$$

Note: C_g Calculation = 0.707

$$f_g(x, y) = a_g y - (b_g x + c_g) \Big|_{\substack{a_g=x_g=x_a-x_b \\ b_g=y_g=y_a-y_b}}$$

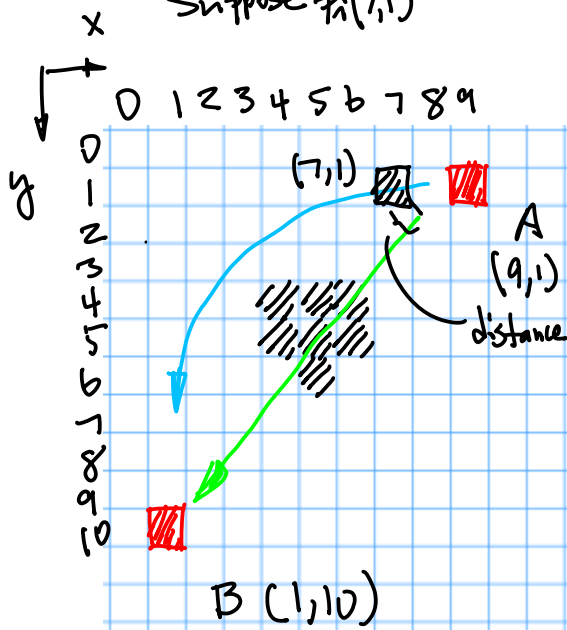
Take $(x, y) = (x_a, y_a)$
Substitute into the above equation.

$$\begin{aligned}0 &= a_g y_a - (b_g x_a + c_g), \quad c_g = a_g y_a - b_g x_a \\ &= (x_a - x_b) y_a - (y_a - y_b) x_a\end{aligned}$$

$$\therefore c_g = (x_a - x_b) y_a - (y_a - y_b) x_a \quad \dots (13)$$

Example: Hand Calculation of Distance

Suppose $\vec{P}_i(7,1)$



Step 1. Find direction vector of green line

$$\vec{d}_G = (x_a - x_b, y_a - y_b) = (x_G, y_G) \quad \dots (4)$$

$$= (9 - 1, 1 - 10) = (8, -9)$$

Step 2. Find green line Equation in Parametric Form.

$$f_G(x, y) = a_G y - (b_G x + c_G)$$

From

$$a_G = x_a - x_b, \quad b_G = y_a - y_b \quad \dots (8)$$

$$\vec{d}_G = (a_G, b_G) = (x_G, y_G) \quad \dots (8b)$$

$$a_G = 8, \quad b_G = -9, \quad \text{hence} \quad \begin{matrix} = (9-1, 1-10) \\ = (8, -9) \end{matrix}$$

$$f_G(x, y) = 8y - (-9x + c_G)$$

From Eqn (13)

$$c_G = (x_a - x_b) y_a - (y_a - y_b) x_a$$

$$= a_G y_a - b_G x_a$$

$$= 8 \cdot y_a - (-9) x_a = 8 \cdot 1 - (-9) \cdot 9$$

$$= 8 + 81 = 89$$

Step 3. Find f_G .

$$f_G(x_i, y_i) = a_G y_i - (b_G x_i + c_G) \quad \dots (b)$$

$$= a_G y_i - b_G x_i - c_G = 8y_i - (-9)x_i - 89 \quad \left| \begin{matrix} x_i = 7 \\ y_i = 1 \end{matrix} \right.$$

$$= 8 \cdot 1 + 9 \cdot 7 - 89 = 71 - 89 = -18$$

Step 4. Find λ

$$\lambda = - \frac{f_G(x_i, y_i)}{a_G^2 + b_G^2} \quad \dots (12)$$

$$= - \frac{-18}{a_G^2 + b_G^2} = \frac{18}{8^2 + (-9)^2} = \frac{18}{64 + 81}$$

$$= \frac{18}{145} = 0.12$$

Step 5. Find intersection point $P(x, y)$ on the green line, Eqn (11b)

$$\vec{P}(x, y) = \vec{P}_i(x_i, y_i) + \lambda \vec{d}$$

$$= (x_i, y_i) + \lambda (-y_G, x_G)$$

$$x = x_i + \lambda (-y_G) \quad \left| \begin{matrix} x_i = 7, \lambda = 0.12 \end{matrix} \right. = 7 + 0.12 \cdot (-(-9)) = 7 + 0.12 \cdot 9 = 7 + 1.08 = 8.08$$

$$y = y_i + \lambda x_G = 1 + \lambda (x_G) = 1 + 0.12 \cdot 8 = 1 + 0.96 = 1.96$$

Step 6. Find distance, from Eqn (12)

$$\text{dist}(\vec{P}_i, \text{green}) = \sqrt{(x_i - x)^2 + (y_i - y)^2}$$

$$= \sqrt{(7 - 8.08)^2 + (1 - 1.96)^2}$$

$$= \sqrt{1.08^2 + 0.96^2} = \sqrt{1.1664 + 0.9216}$$

$$= 1.445$$