

# AI4R: 2nd Half Notes

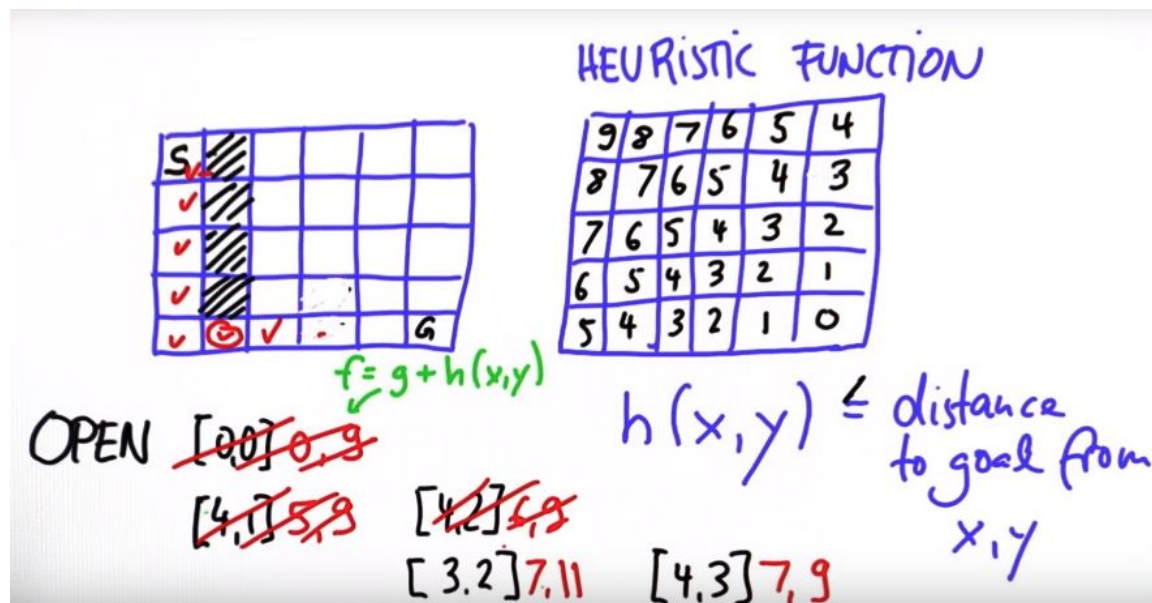
By Brandon Kalashian

For a table of content, click [View](#) -> [Show Document Outline](#)

## Lesson 12 - Search

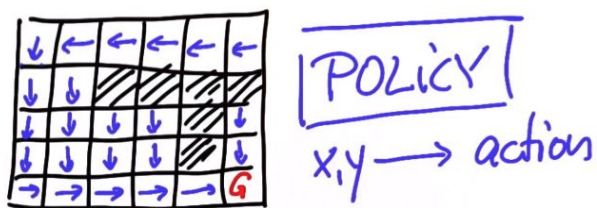
### A\*

- Uses heuristic function
- Deterministic path
- Admissible if  $h(x) \leq \text{goal}$  (numbers too large compared to surrounding ones)

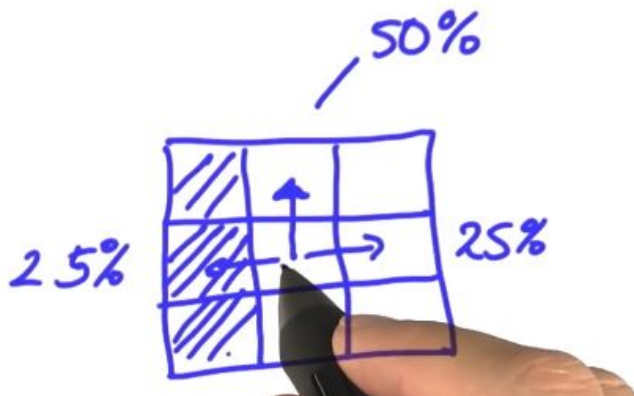


### Dynamic Programming

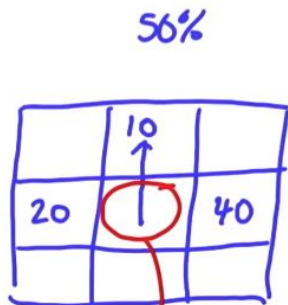
- Best path from anywhere
- Stochastic (adaptive to world) policy



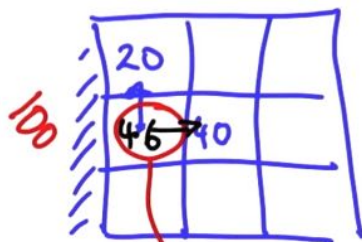
- Stochastic Motion example
  - Provide clearance
  - Movement succeeds based on probability



50% chance of succeeding, 25% chance of hitting left wall



$$\text{Value}(\uparrow) = \begin{array}{r} 50\% \cdot 10 \\ 25\% \cdot 20 \\ 25\% \cdot 40 \\ + 1 \end{array} \left. \begin{array}{l} 5 \\ 5 \\ 10 \\ 1 \end{array} \right\} = 21$$

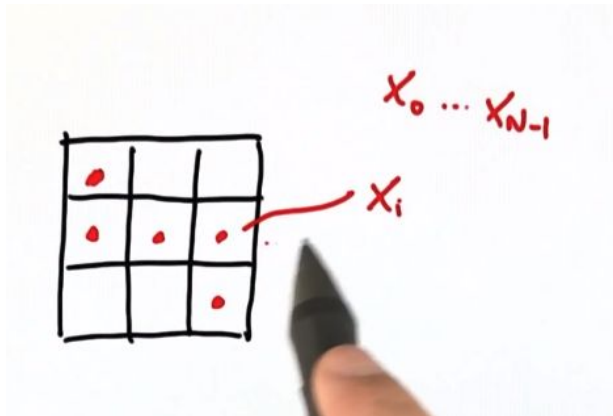


$$\begin{array}{r} V(4) = 50\% \cdot 20 \quad 10 \\ + 25\% \cdot 40 \quad 10 \\ + 25\% \cdot 100 \quad 25 \\ + 1 \quad 1 \\ \hline 46 \end{array}$$

## Lesson 15 - PID Control

### Smoothing Algorithm

Original points:



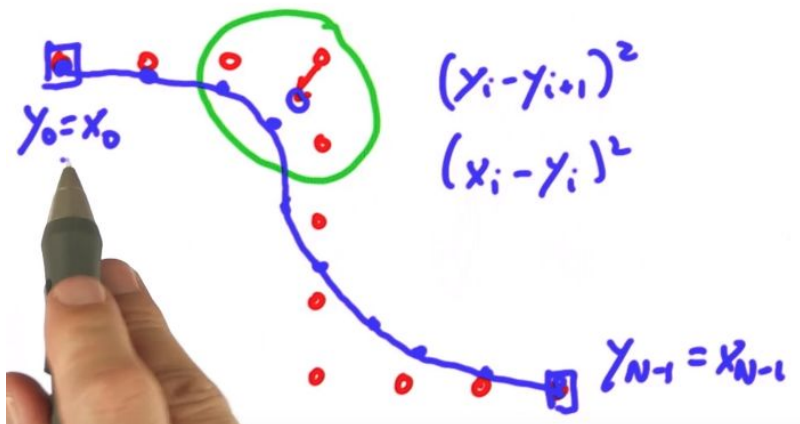
### SMOOTHING ALGORITHM

①  $y_i = x_i$

② OPTIMIZE

$$(x_i - y_i)^2 \rightarrow \min$$

$$(y_i - y_{i+1})^2 \rightarrow \min$$



## Smoothing by gradient descent

Gradient Descent

$(x_i - y_i)^2 \rightarrow \text{Min}$   
 $(y_i - y_{i+1})^2 \rightarrow \text{Min}$

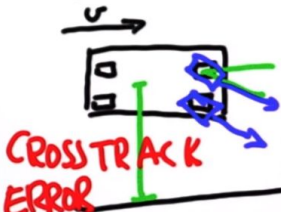
$y_i = y_i + \alpha (x_i - y_i)$   
 $y_i = y_i + \beta (y_{i+1} + y_{i-1} - 2y_i)$   
 $\alpha = 0.5$   
 $\beta = 0.1$

$\begin{bmatrix} 0, 0 \\ 0, 1 \\ 0, 2 \\ 1, 2 \\ 2, 2 \\ 3, 2 \\ 4, 2 \\ 4, 3 \end{bmatrix}$

## P-Control

- "Proportional" - Controller (P-Controller)
- Cross track error = willingness to turn towards target trajectory (larger = more willing)

PID - CONTROL



o steering constant  
 o random steering commands  
 x steer in proportion to CTE.

CROSS TRACK ERROR

REFERENCE TRAJECTORY

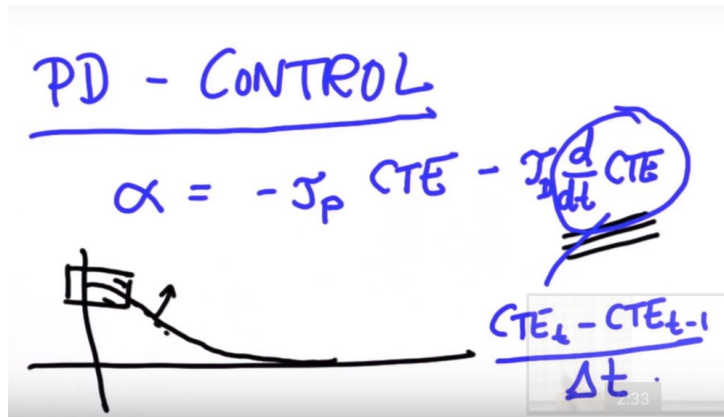
- Applying  $[\alpha = -J * \text{CTE}]$  results in marginal stable:

$$\alpha = -J_p \text{CTE}$$



## PD Control

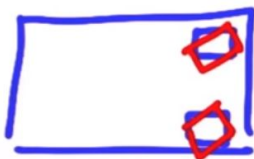
- A way to remove the oscillations of P-Control
- $\alpha = -J_p \cdot CTE - J_d \frac{d}{dt} CTE$



## Systematic Bias

(IE wheels manufactured not perfectly straight)

### SYSTEMATIC BIAS



WHAT HAPPENS?

o JUST AS BEFORE  
x CAUSES BIG CTE

- solution = adjust to bias

$$\alpha = \underbrace{-J_p CTE}_P - \underbrace{J_d \frac{d}{dt} CTE}_D - \underbrace{J_i \sum CTE}_I$$

Proportional differential Integral

PID

## TWIDDLE (Coordinate Ascent)

TWIDDLE

```
run() → goodness
p = [0, 0, 0]
dp = [1, 1, 1]
best_err = run(p)
for i in range(3):
    p[i] += dp[i]
    err = run(p)
    if err < best_err:
        best_err = err
        dp[i] *= 1.1
    else:
        p[i] -= 2 * dp[i]
        p[i] += dp[i]
        dp[i] *= 0.9
```

TWIDDLE





## Lesson 19 - SLAM

Putting it all together:

|              |               |
|--------------|---------------|
| LOCALIZATION | PROBABILITIES |
| TRACKING     | FILTERS       |
| PLANNING     | A*, DP        |
| CONTROL      | PID CONTROL   |

| <u>Quiz</u> | MULTI<br>MODAL | EXPONENTIAL | USEFUL                  |
|-------------|----------------|-------------|-------------------------|
| KALMAN      | ○              | ○           | <del>✗</del>            |
| HISTOGRAM   | ✗              | ✗           | <del>✗</del> <b>YES</b> |
| PARTICLES   | ✗              | ✗           | <del>✗</del>            |

(universal means solution can be applied to arbitrary states)

| <u>Quiz</u>         | Continuous | Optimal | Universal | Local |
|---------------------|------------|---------|-----------|-------|
| Breadth First       | ○          | ✗       | ○         | ○     |
| A*                  | ○          | ✗       | ○         | ○     |
| Dynamic Programming | ○          | ✗       | ✗         | ○     |
| Smoothing           | ✗          | ○       | ○         | ✗     |

| Quiz  | AVOIDS<br>OVERSHOOT | MINIMIZING<br>ERROR | COMPENSATING<br>DRIFT |
|---|---------------------|---------------------|-----------------------|
| PID $\begin{cases} P \\ I \\ D \end{cases}$ | o                   | x                   | o                     |
|   | o                   | o                   | x                     |
|   | x                   | o                   | o                     |

Omega and Xi

$$\mu = \Omega^{-1} \cdot \xi$$

### LINEAR GRAPHSLAM

INITIAL POSITION } add  $\Omega, \xi$   
 MOTION }  
 MEASUREMENT } strength  $\frac{1}{\sigma}$

$$\mu = \Omega^{-1} \cdot \xi$$

→ PATH + MAP