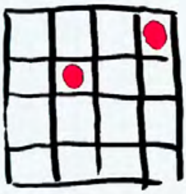
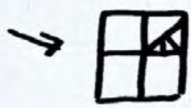
control 03 COGS 300 Feb 03/26

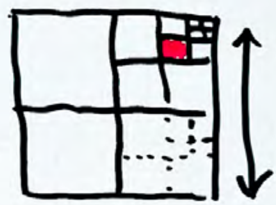
Warm up: discretizations + tiling.

Create different "grids":



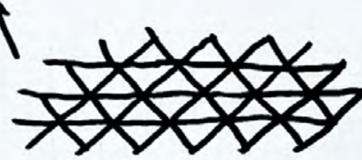
2/16 cartesian





recursive

polar



triangular

1/6 1D linear

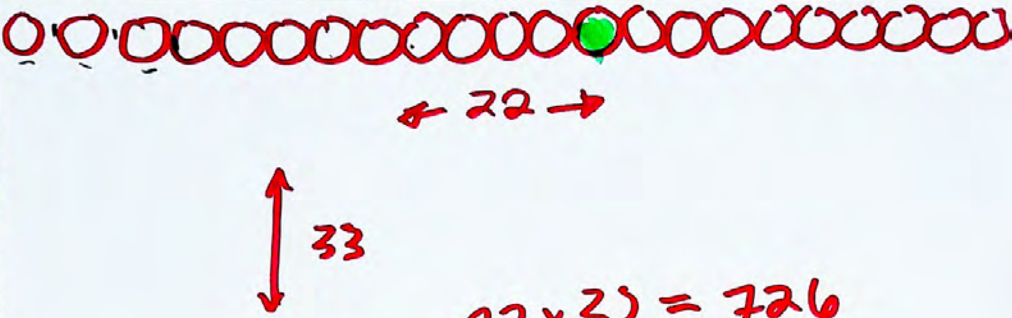
etc...

1/8 pentagon-ish

p(point is covered)?

$r = 1cm$

$A = 21 \times 29.7\ cm^2$



$22 \times 33 = 726$

$1/150$ — $1/726$

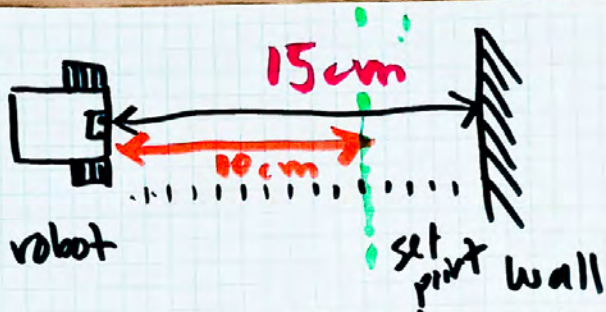
$$\frac{\pi r^2}{A} = 1/623.7$$

$r = 1cm$

$A = 21 \times 29.7 \ cm^2$

0.0016

↑



p(pos) = 1/15

go faster the further we are

slower closer

out = map(dist, min, max, lo, hi);

PID = proportional integral derivative

```
current pos = pos = read();      <- distance
set point = set = 5 cm;
error = pos - set;
output = out = c · error
                ↑ constant
```

"tuning"

err=1cm out = 500

c = 500 err = 10

out = 5000

c = 100 err = 10 out = 1000

c = 100 err = 1 out = 100

c = 17 err = 10 out = 170

c = 17 err = 1 out = 17

simulation + tuning

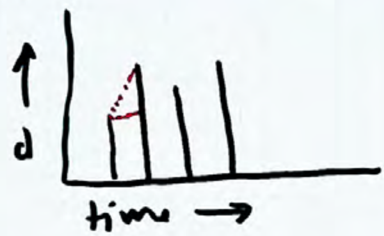
d_0

d_1

d_2

⋮

d_n



$$\frac{d_j - d_i}{1} = d_j - d_i$$

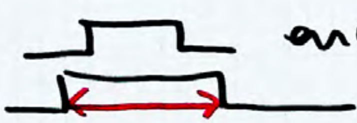
Δd

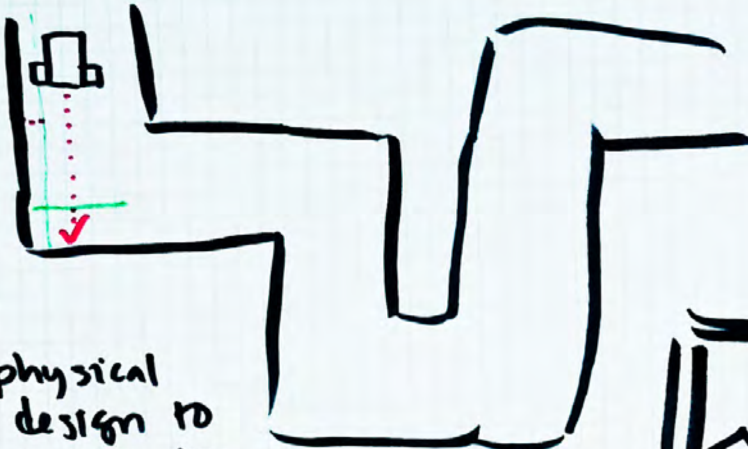
$$c \cdot \text{error} + a \cdot \Delta\text{error} + b \cdot \sum \text{error}$$

P D I

PWM encoding

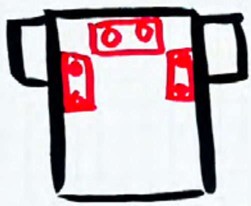
PID control

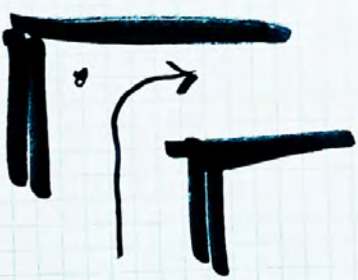




1. physical design to sense in maze

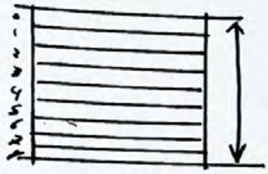
2. Algorithm for control



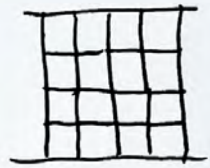


control 03

Warm-up: Discretizations + tiling



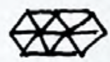
linear



cartesian

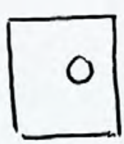
polar

recursive cartesian



triangular

probability: if I throw a coin onto a page, P(point is covered)



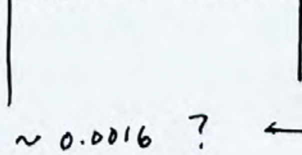
$\frac{\text{area of coin}}{\text{area of page}}$?

$$p = \frac{\pi r^2}{A} = 1/623.7$$

r = 1cm

A = 21 × 29.7 = 623.7

try it!



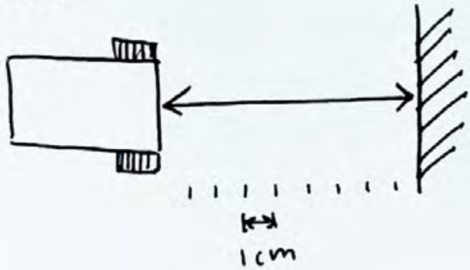
← 10 →

15 ↕

↓ r≈0.5 ↳ closer!

0.0066 ~ 0.0016 ? ⟷ 1/150 ~ $\frac{1}{623.7}$?

Robot.



p (Robot is @ 5cm)?

= 1/10

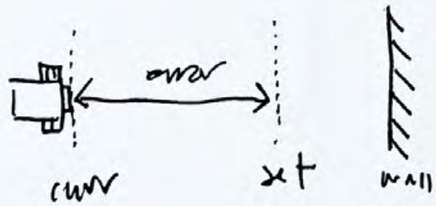
$$\frac{\text{1cm chunk}}{\text{\# of chunks}}$$

↳ robot doesn't "know," it has to guess.

Last time: alg. for driving to set point.

Key insight: go faster, further.

output = map (dist to set point, min dist, max dist, min motor, max motor).




```
error = curr - set ;
output = map(error, 0, 10, 0, 255) ;
```

↳ but what about a diff. curve?

```
curr = read();
set = 5; // cm
error = curr - set;
out = c · error
      ↑
      some constant
```

c = 100   err = 5   out = 500

c = 50   err = 5   out = 250

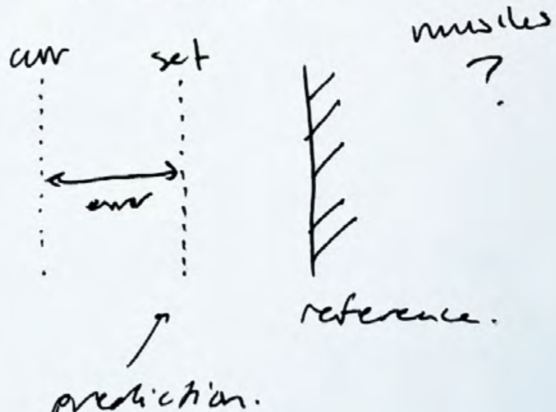
c = 1/5   err = 5   out = 1

depends on <u>your</u> robot.

PWM — encoding

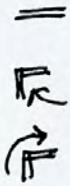
PID — control

control requires sensation + model.
to evaluate.



★ come up with a wall-following algorithm. + Design

Cases:



straight

inside

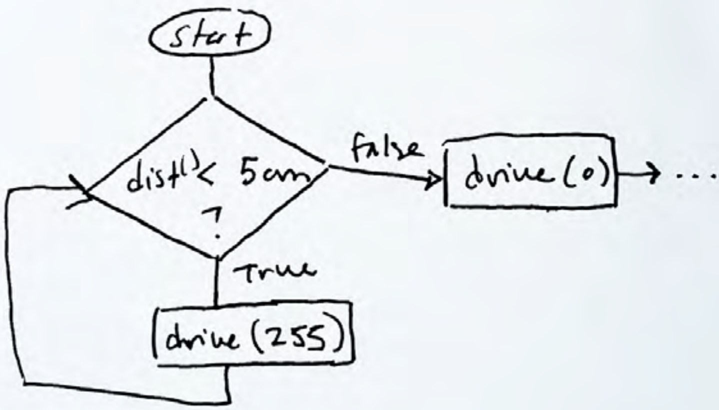
outside.

→ which sensors / where do you put them?

→ which filters do you use, why?

→ what is the algo?

↳ low level

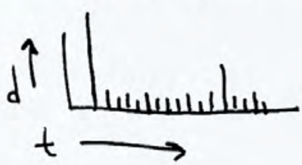


why doesn't this "just work"?

vs. PID version...

Shifting into a probabilistic mindset...

How many measurements until you're ***sure***?



Stability of signal

Δd = change in d = $d_i - d_{i-1}$

small Δd is small variation... but also fragile!

avg$(d_i : d_j)$ var$(d_i : d_j)$ etc.

⋮

$p(\text{pos} \mid \text{measurement})$?

☆ $\dfrac{p(\text{pos})}{p(\text{measurement})}$?