Kojima - Kojimouse straight attitude control

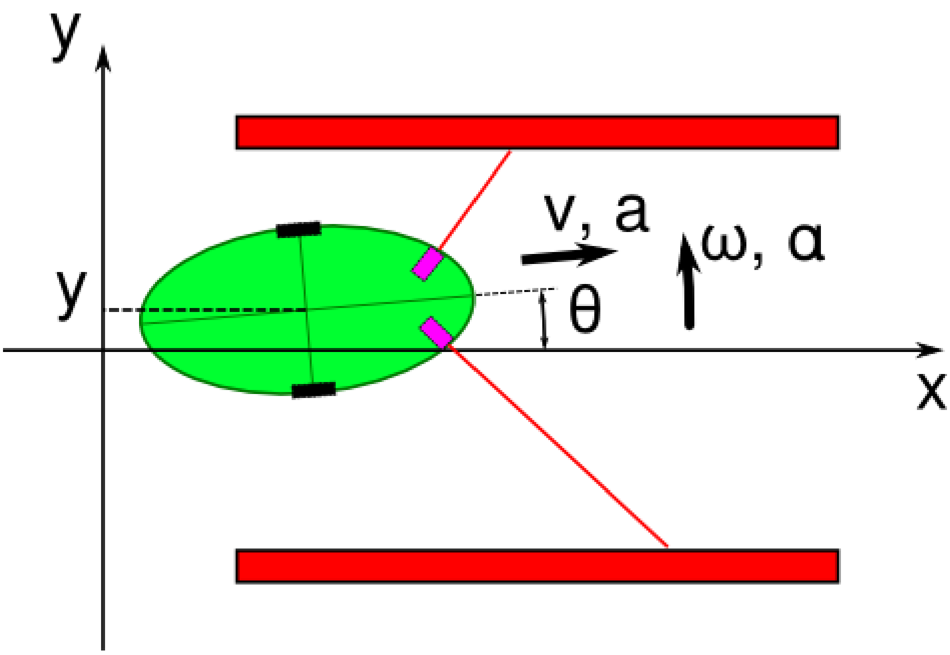
Kojima - Kojimouse straight attitude control method

Translated loosely from blog posts made by Japanese mouse builder Kojima-san in 2011

DRAFT - 16-MAR-2023

Coordinate system:

https://kojimousenote.blogspot.com/2011/12/blog-post_04.html



y is the lateral displacement form the maze centre
θ is the mouse angle wrt the wall
v is the mouse speed in the direction of travel
a is the acceleration in the direction of travel.
ω is the mouse angular velocity
α is the mouse angular acceleration

Measurement of v is from the encoders and measurement of ω is done by the gyro.

Control when y an θ are known

https://kojimousenote.blogspot.com/2011/12/y.html

The goal of the control method is to drive y to zero by manipulating the angular acceleration α. If both y and $theta$ are known the following can be done.

First consider the time derivative of y. Angles are measured in radians and we can use the small angle approximation

$$\dot{y} = v sin\theta \approx v\theta \tag{1}$$

Differentiate again to get

$$\begin{aligned} \ddot{y} &= a\theta + v\dot{\theta} \\ &= a\theta + v\omega \end{aligned} \tag{2}$$

Using the concept of a PD controller where P, D and D_2 are gains

$$\frac{d}{dt}\ddot{y} = -(D_2\ddot{y} + D\dot{y} + Py) \tag{3}$$

We can also differentiate (2) to get

$$\frac{d}{dt}\ddot{y} = \dot{a}\theta + 2a\omega + v\alpha \tag{4}$$

Combining (3) and (4) gives

$$\alpha = -\frac{(D_2\ddot{y} + D\dot{y} + Py + 2a\omega)}{v} \tag{5}$$

We can assume that the change in forward acceleration a has a negligible effect.

If y, v, a, θ, ω are all known then we can calculate an angular acceleration. α that will let us run safely even at high speed and acceleration.

BUT - y and θ cannot be obtained from simple measurements.

Photo Sensor Signal

https://kojimousenote.blogspot.com/2011/12/blog-post_8223.html

The problem with the ability to easure quantities directly means we have to look for another technique. Suppose instead that we try to change the angular acceleration in such a way as to drive the wall sensor value s to its reference value, s_0 when the mouse is centered in the maze. Let the error seen by the sensor be S such that

$$S = s - s_0 \tag{6}$$

The value of s is a function of both position y and angle θ and here we will assume it can be described by the simple linear equation:

$$S = k_1 y + k_2\theta \tag{7}$$

Differentiate wrt time and substitute from above to give

$$\begin{aligned} \dot{S} &= k_1 \dot{y} + k_2 \dot{\theta} \\ &= k_1 v \theta + k_2 \omega \end{aligned} \tag{8}$$

For the PD controller we have

$$\ddot{S} = -(D\dot{S} + PS) \tag{9}$$

Differentiating (7) gives

$$\ddot{S} = k_1(a\theta + v\omega) + k_2 \alpha \tag{10}$$

Now combine (9) and (10) to get

$$\alpha = -\frac{(D\dot{S} + PS + k_1(a\theta + v\omega)}{k_2} \tag{11}$$

This is still not ideal because there remains a term for θ **(PH - not that he suggests what to do about it - what if we treated it as zero?)** Values for k_1 and k_2 must be found empirically. **(PH - and what are P and D in this?)**

For some time, Kojimouse has been controlled using (11). When there is a wall, the sensor error S is used. When there is no wall, S can be calculated from (7). **(PH - but where do we get y and θ from?)**

The controller simplified

https://kojimousenote.blogspot.com/2011/12/blog-post_06.html

Kojima hints that the control law is not so good at higher speeds. A simplification is possible if we assume that $\theta - 0$ and we eliminate S in 98) and (11)**(PH - what does he mean by 'eliminate'?)**

This gives us a new control law

$$\alpha = -\left[\left(D + \frac{k_1}{k_2} v\right)\omega + \frac{P}{k_2} S\right] \tag{12}$$

In this way the sensor value and angular velocity are fed back as is often used. Note that the gain changes with the angular velocity

(PH - this is conjecture because he suggests getting the 'junior' to implement it to see what happens)

OTHER QUESTIONS - PH

Is the sensor reading a distance or an uncorrected reading?

Simulation at the micromouse training camp.

https://kojimousenote.blogspot.com/2011/12/blog-post_11.html
File is at: http://www.kikaiken.org/lib/micromouse/111211/PDs_sim.xls
Kojima provides a spreadsheet simulatingthe controller
kojimas-steering-correction-PDs-sim.xls

the simulation does not use S as an input though