Mobile Robots Odometry and Errors.

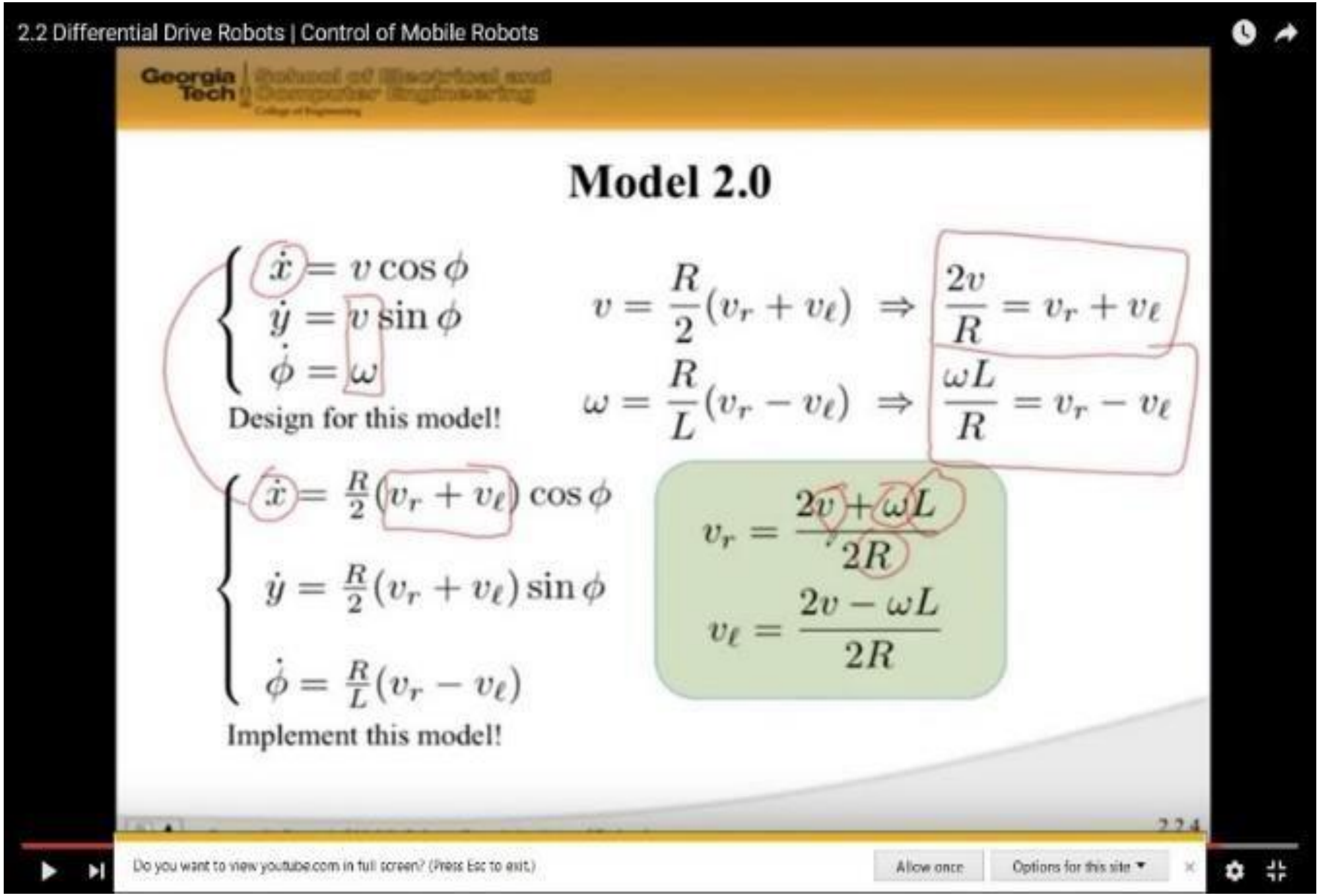
NO ERRORS ANYWHERE yet!

CONTROL – WHEEL VELOCITY, ROTATION

WE DO NOT KNOW DISTANCE. THAT IS INTEGRAL OF VELOCITY

Assuming ω= 0 and v is constant along line at angle Φ,
 we have

x = v cos(Φ) · t meters
y = v sin(Φ) · t   meters



MAGNUS

https://www.youtube.com/watch?app=desktop&v=aE7RQNhwnPQ&sns=em

We have already noted that **odometry is not accurate because inconsistent measurements and irregularities in the surface can cause errors.** In this section we show that even small changes in the direction of the robot's movement can cause errors that are much larger than those caused by changes in its linear motion.

To simplify the presentation, let us assume that a robot is to move **10 m** from the origin of a coordinate system along the *x*-axis and then check its surroundings for a specific object.

What is the effect of an error of *up to* p%? If the error is in the measurement of *x*, the distance moved, then Δx, the error in *x* is:

$$\Delta x \leq \pm 10 \text{ m} \cdot p/100 = \pm p/10 \text{ meters},$$

where the value is negative or positive because the robot could move up to p% before or after the intended distance.

Check: In 10 m, p=10% is +- 1 m in error. This checks.

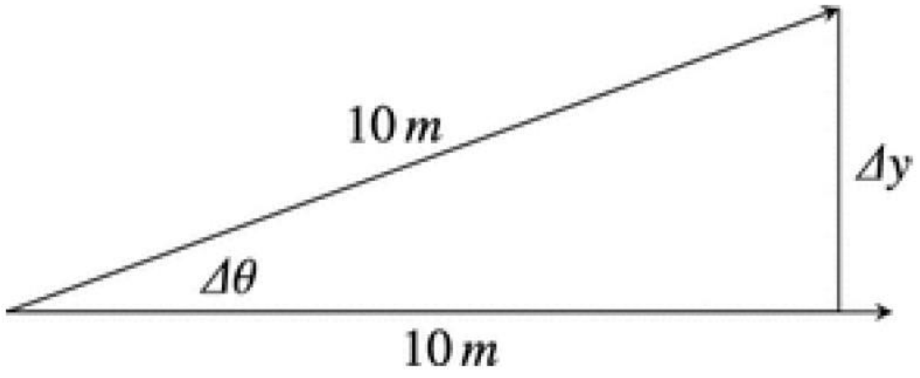
https://link.springer.com/chapter/10.1007/978-3-319-62533-1_5

Robotic Motion and Odometry

Elements of Robotics pp 63-93 | Cite as

- Mordechai Ben-Ari (1) Email author (moti.ben-ari@weizmann.ac.il)
- Francesco Mondada (2)

Suppose now that there is an error $p\%$ in the *heading* of the robot, and, for simplicity, assume that there is no error in the distance moved. The geometry is:



The robot intended to move 10 m along the x-axis, but instead it moved slightly to the left at an angle of $\Delta\theta$. Let us compute the left-right deviation Δy. By trigonometry, $\Delta y = 10 \sin \Delta\theta$. An error of $p\%$ in heading is:

$$\Delta\theta = 360 \cdot \frac{p}{100} = (3.6p)^\circ \,,$$

so the left-right deviation is:

$$\Delta y \leq \pm 10 \sin(3.6p) \,.$$

The following tables compare the difference between a linear error of $p\%$ (left) and an error in heading of $p\%$ (right):

$p\%$	Δx (m)
1	0.1
2	0.2
5	0.5
10	1.00

$p\%$	$\Delta\theta$ (°)	$\sin \Delta\theta$	Δy (m)
1	3.6	0.063	0.63
2	7.2	0.125	1.25
5	18.0	0.309	3.09
10	36.0	0.588	5.88

For a very small error like 2%, the distance error after moving 10 m is just 0.2 m, which should put the robot in the vicinity of the object it is searching for, but a heading error of the same percentage places the robot 1.25 m away from the object. For a more significant error like 5% or 10%, the distance error (50 or 100 cm) is still possibly manageable, but the heading error places the robot 3.09 or 5.88 m away, which is not even in the vicinity of the object.

https://link.springer.com/chapter/10.1007/978-3-319-62533-1_5

5.3 From Segments to Continuous Motion

As the size of the segments becomes smaller, we obtain the instantaneous velocity of the robot at a single point in time, expressed as a derivative:

$$v(t) = \frac{ds(t)}{dt} \,.$$

Similarly, the instantaneous acceleration of the robot is defined as:

$$a(t) = \frac{dv(t)}{dt} \,.$$

For constant acceleration the velocity can be obtained by integrating the derivative:

$$v(t) = \int a \, dt = a \int dt = at \,,$$

and then the distance can be obtained by integrating again:

$$s(t) = \int v(t) dt = \int a \, t \, dt = \frac{at^2}{2} \,.$$

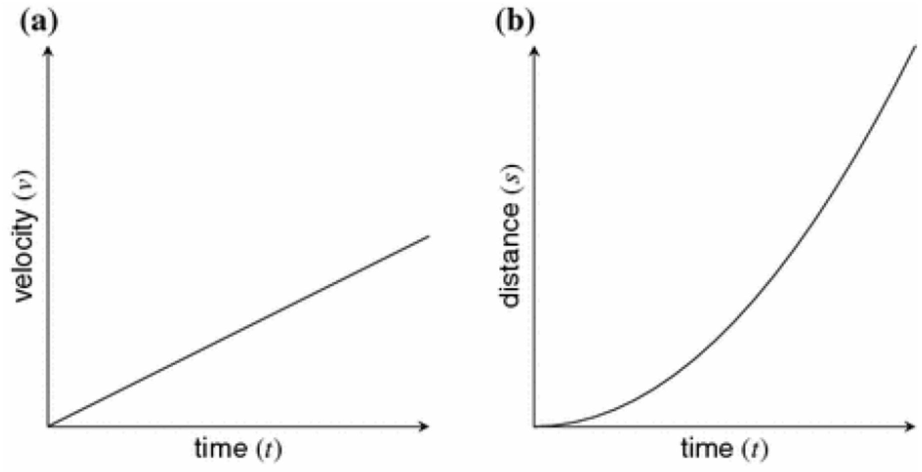


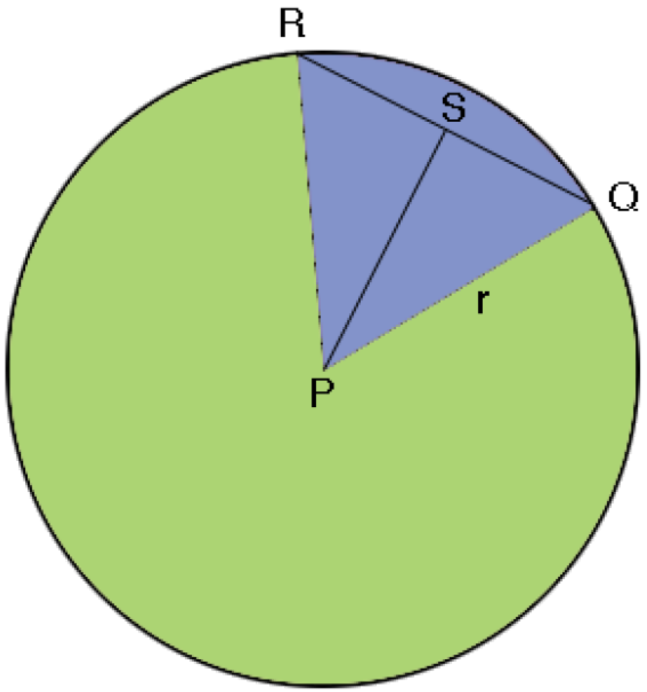
Fig. 5.2

a Velocity for constant acceleration. **b** Distance for constant acceleration

So constant error in acceleration is increasing as t^2

https://link.springer.com/chapter/10.1007/978-3-319-62533-1_5 http://mathcentral.uregina.ca/qq/database/qq.09.09/h/wayne1.html

Find the length **c** of the chord RQ in the diagram



S is the midpoint of RQ so |SQ| = **c**/2 and the measure of the angle SPQ is **t**/2. Also QSP is a right angle so sin(**t**/2)= |SQ|/**r**. Hence |SQ| = **r** × sin(**t**/2) and thus

c = 2 × **r** × sin(**t**/2).

If t is a small angle < 6 deg (0.1 radian),
Error in course is
C/2 = r * t/2, t in radians

For 1 degree off straight (0.0174 r),
Error in km is 8.7 meters.

Example An average car accelerates from 0 to 100 km/h in about 10 s. First, we convert units from km/h to m/s:

$$v_{max} = 100\,\mathrm{km/h} = \frac{100 \cdot 1000}{60 \cdot 60}\,\mathrm{m/s} = 27.8\,\mathrm{m/s}\,.$$

Assuming constant acceleration, $v_{max} = 27.8 = at = 10a$, so the acceleration is 2.78 m/s^2 (read, 2.78 meters per second per second, that is, every second the speed increases by 2.78 meters per second). The distance the car moves in 10 s is:

Don’t forget stopping:
100 km/hr = 60 mph
D = 88ft/sec or about 30 meters/second

REACTION TIME HUMAN - ½ to ¾ second
Full braking in ideal conditions – about 125 ft or 40 meters

TOTAL = 20 METERS + 40 METERS OR 200 FEET.

https://upcommons.upc.edu/bitstream/handle/2099.1/20236/Adaptive%20motion%20planning%20for%20a%20mobile%20robot.pdf?sequence=1&isAllowed=y

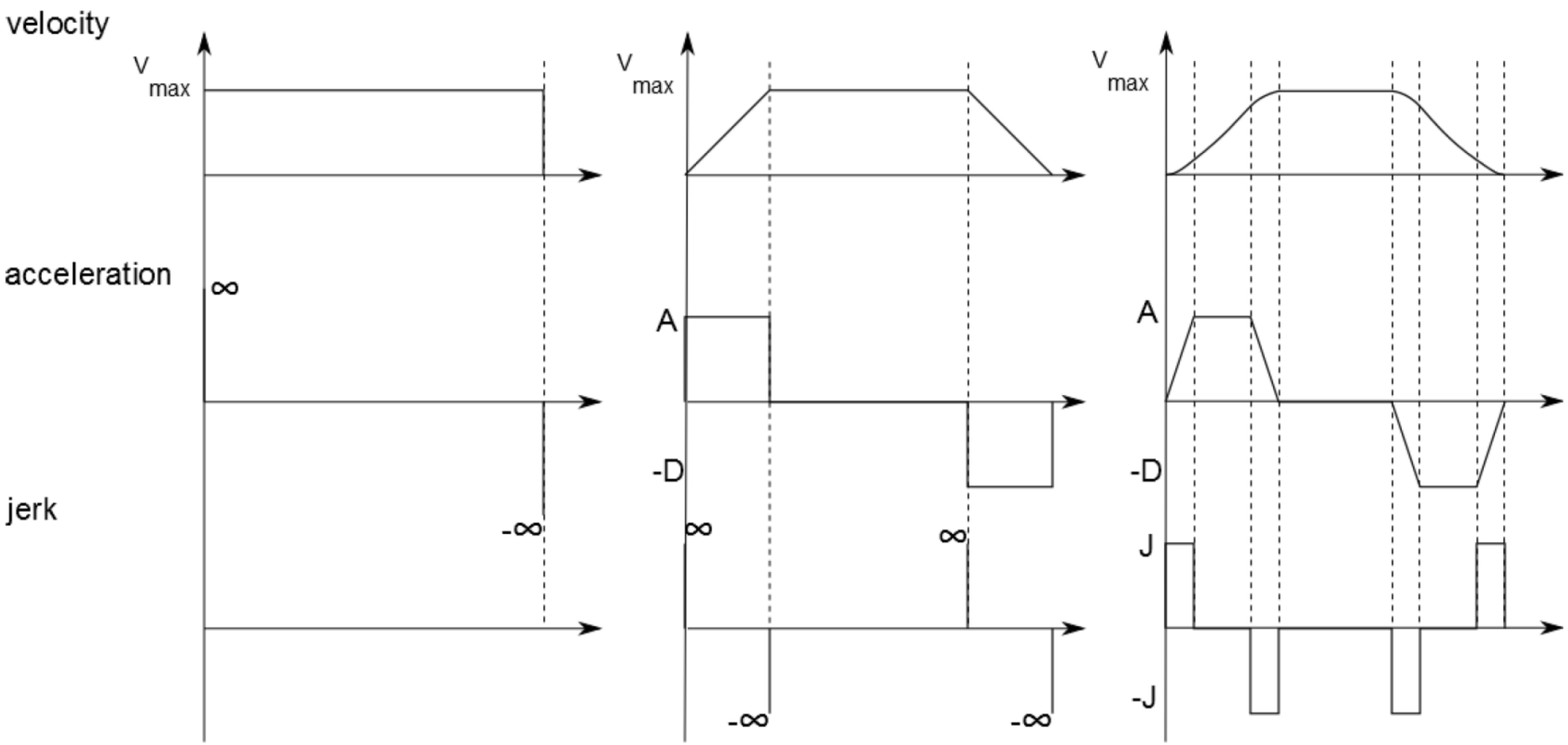
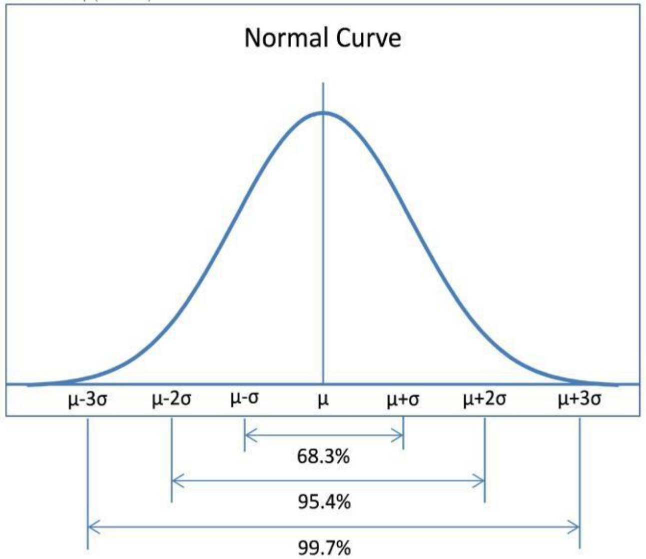


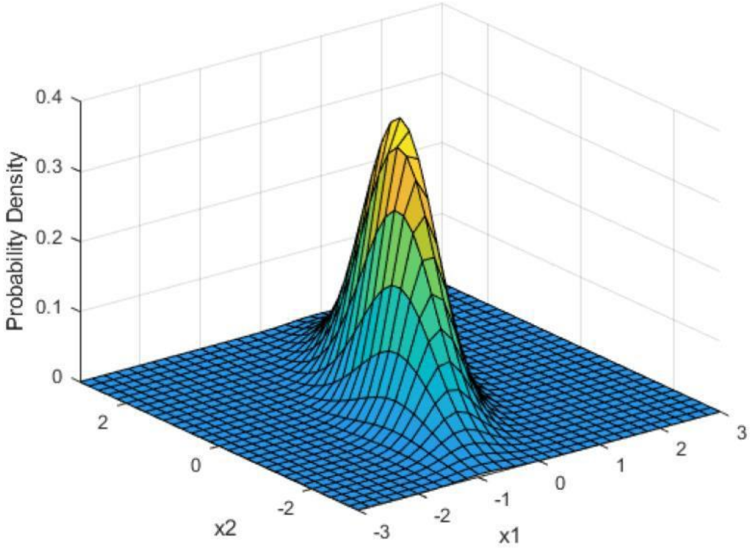
Figure 1. Constant, trapezoidal and s-curve velocity profiles.

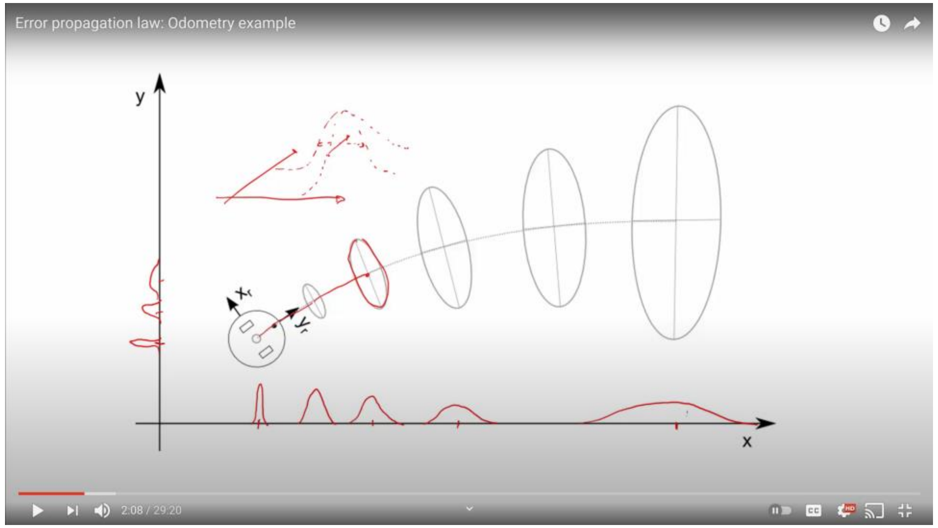
Odometry Errors & Variance & Covariance

deviation = sqrt(variance).



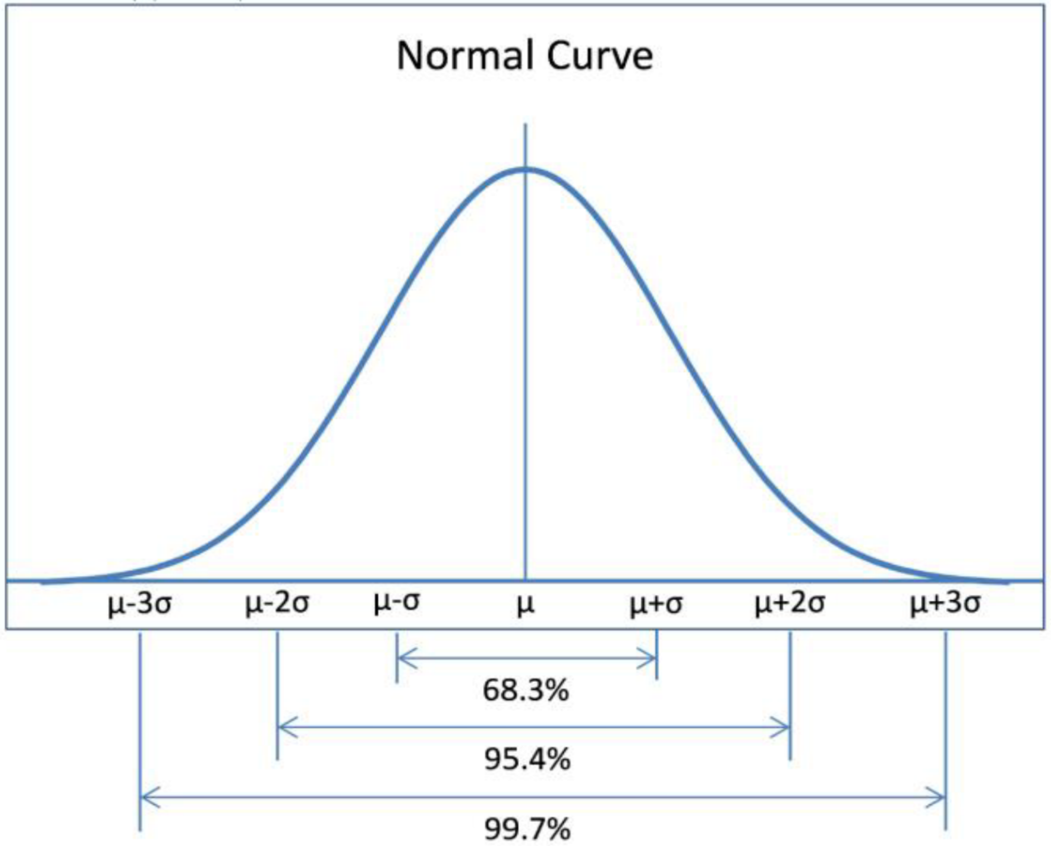
Consider u in the graph as your true value. The value in your Twist message has some error associated with it. If you set the s.d. for linear.x as 1 (cov(linear.x, linear.x) = 1^2 = 1), it means if you take 100 measurements of linear.x, 68.3% will lie between [truevalue-1, truevalue+1] i.e. [u-s.d. , u + s.d.]





LET'S LOOK AT A STANDARD MODEL
FOR VARIABLES WITH RANDOM ERRORS.

deviation = sqrt(variance).



Consider u in the graph as your true value. The value in your Twist message has some error associated with it. If you set the s.d. for linear.x as 1 (cov(linear.x, linear.x) = 1^2 = 1), it means if you take 100 measurements of linear.x, 68.3% will lie between [truevalue-1, truevalue+1] i.e. [u-s.d. , u + s.d.]

Una

Loa

In probability theory, a **normal** (or **Gaussian** or **Gauss** or **Laplace–Gauss**) **distribution** is a type of continuous probability distribution for a real-valued random variable. The general form of its probability density function is

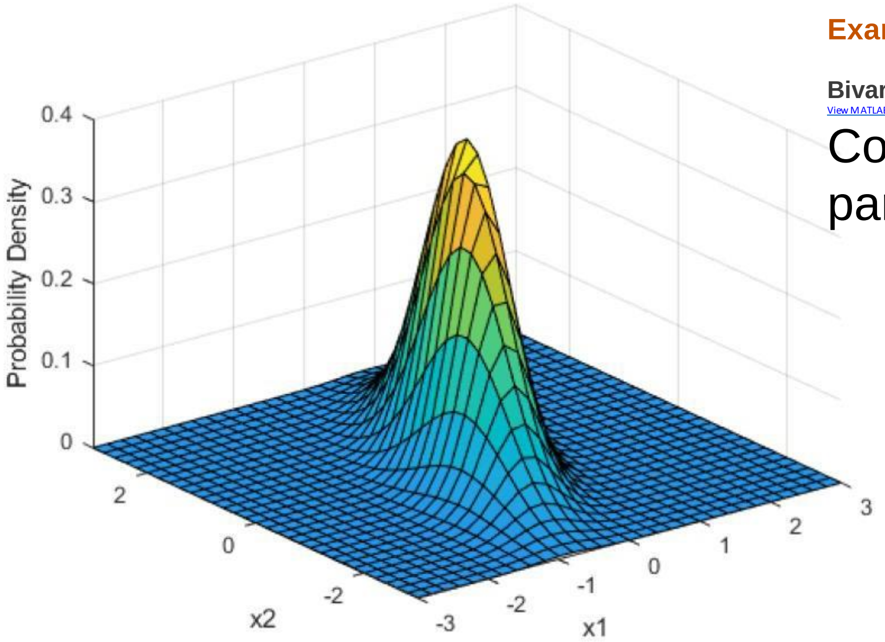
$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

The parameter μ is the mean or expectation of the distribution (and also its median and mode), while the parameter σ is its standard deviation.[1] The variance of the distribution is σ^2.[2] A random variable with a Gaussian distribution is said to be **normally distributed**, and is called a **normal deviate**.

SUPPOSE THERE ARE RANDOM ERRORS IN SEVERAL VARIABLES OF INTEREST i.e. x,y, Φ

Multivariate Gaussian distributions
173,941 views •Nov 19, 2012 14:48

https://www.youtube.com/watch?v=eho8xH3E6mE



Examples

Bivariate Normal Distribution pdf

View MATLAB Command

Compute and plot the pdf of a bivariate normal distribution with parameters mu = [0 0] and Sigma = [0.25 0.3; 0.3 1].

NOTE: variables x1 and y1 are not independent since Covariance matrix Sigma is not diagonal.

https://www.mathworks.com/help/stats/multivariate-normal-distribution.html

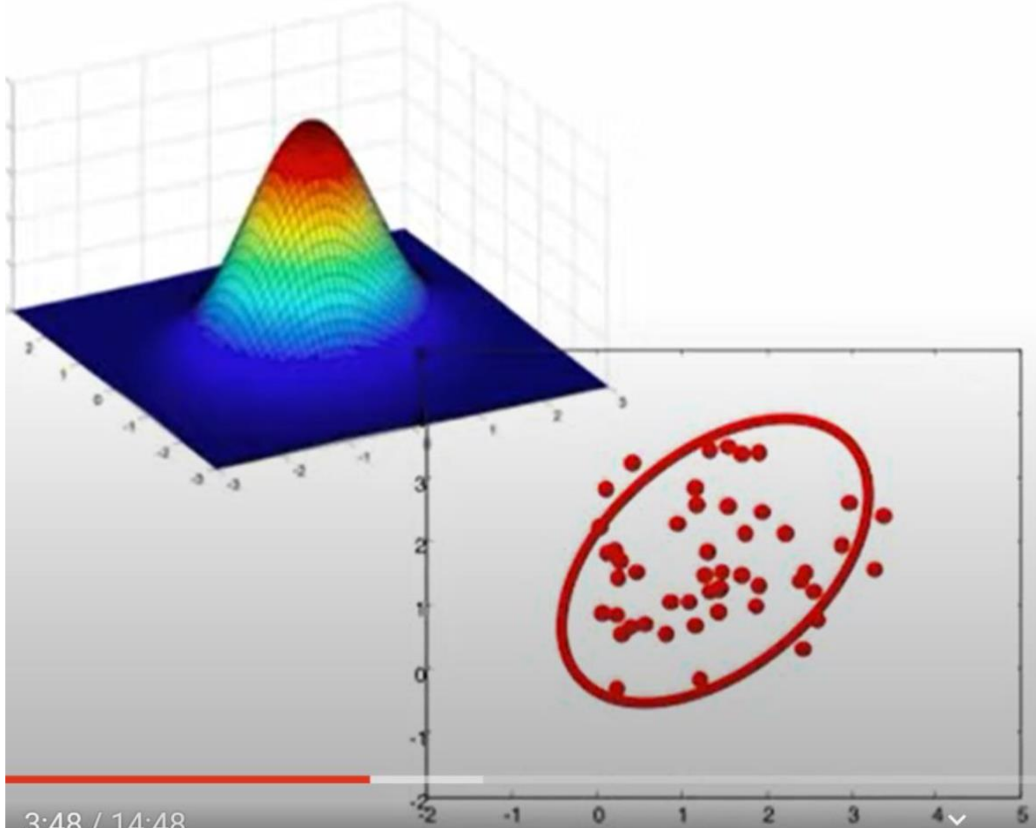
Multivariate Gaussian models

- Similar to univariate case

$$\mathcal{N}(\underline{x}\ ;\ \underline{\mu}, \Sigma) = \frac{1}{(2\pi)^{d/2}} |\Sigma|^{-1/2} \exp\left\{ -\frac{1}{2}(\underline{x} - \underline{\mu})\Sigma^{-1}(\underline{x} - \underline{\mu})^T \right\}$$

μ = length-d row vector
Σ = d x d matrix

$|\Sigma|$ = matrix determinant



Maximum likelihood estimate:

$$\hat{\mu} = \frac{1}{m} \sum_j \underline{x}^{(j)}$$

$$\hat{\Sigma} = \frac{1}{m} \sum_j (\underline{x}^{(j)} - \hat{\underline{\mu}})^T (\underline{x}^{(j)} - \hat{\underline{\mu}})$$

(average of dxd matrices)

Independent Gaussian models

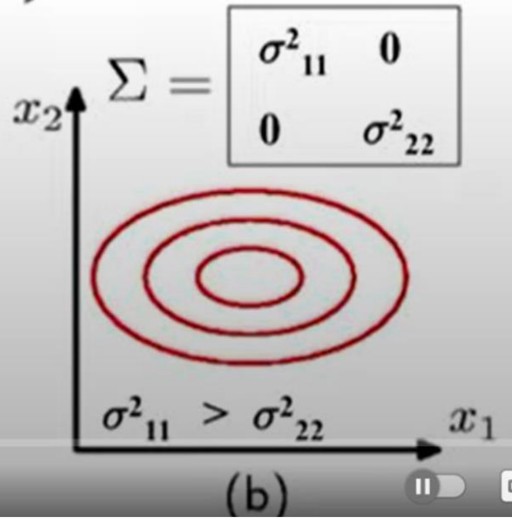
$$p(x_1) = \frac{1}{Z} \exp\left\{-\frac{1}{2\sigma_1^2}(x_1 - \mu_1)^2\right\} \qquad p(x_2) = \frac{1}{Z_2} \exp\left\{-\frac{1}{2\sigma_2^2}(x_2 - \mu_2)^2\right\}$$

$$\underline{x} = [x_1 \; x_2]$$

$$p(x_1)p(x_2) = \frac{1}{Z_1 Z_2} \exp\left\{-\frac{1}{2}(\underline{x} - \underline{\mu})^T \Sigma^{-1} (\underline{x} - \underline{\mu})\right\}$$

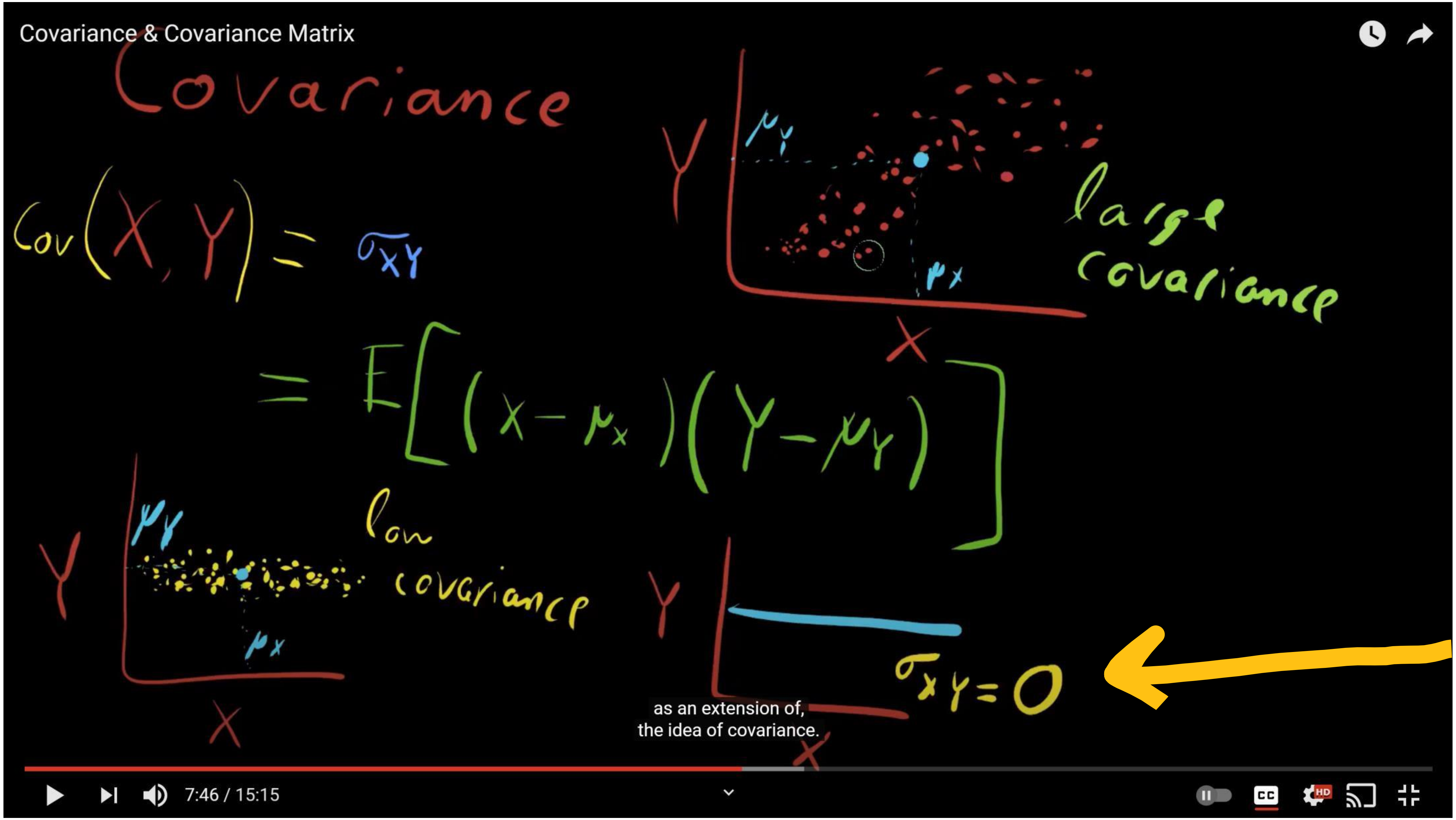
$$\underline{\mu} = [\mu_1 \; \mu_2]$$

$$\Sigma = \mathrm{diag}(\sigma_1^2 \; , \; \sigma_2^2)$$



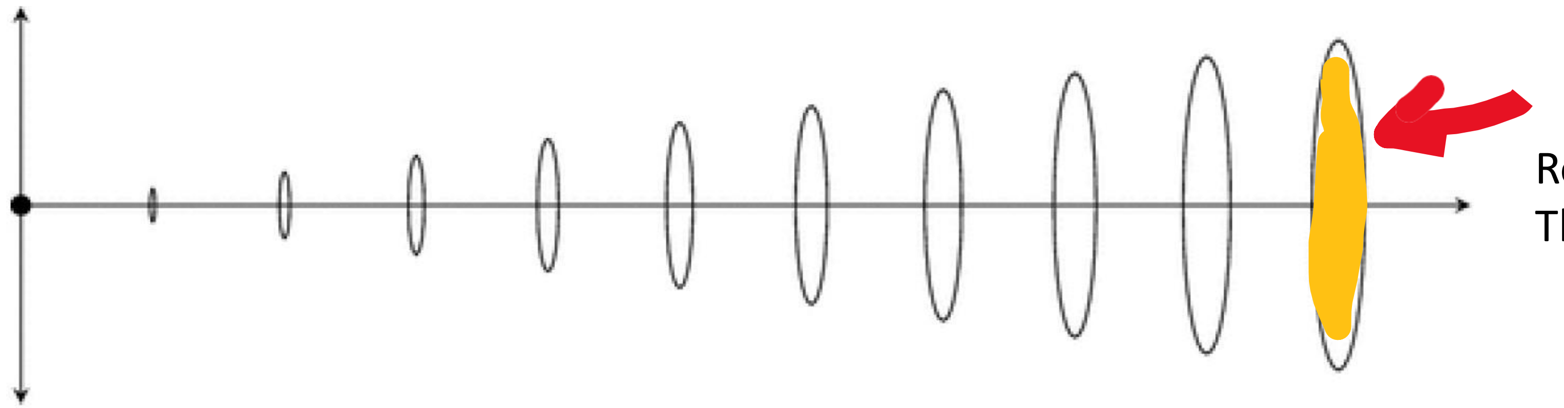
https://www.youtube.com/watch?v=D8p0KRdjmdU

Covariance & Covariance Matrix Building Intuition



Xand Y
Independent
-Diagonal Cov
Matrix

ERROR DEPENDS ON DISTANCE TRAVELED AND INITIAL HEADING ERROR.
ASSUME THE ERROR IS **RANDOM** AND CANNOT BE FIXED BY CALIBRATION.



Robot is somewhere in
This region

Fig. 5.6

Odometry errors

The accumulation of odometry errors as the distance moved gets longer is displayed in Fig. 5.6. The initial position of the robot is denoted by the dot at the origin. Assuming an error of at most $\pm 4\%$ in both the linear direction and the heading, the possible positions of the robot after moving $d = 1, 2, \ldots, 10$ m are displayed as ellipses. The minor radii of the error ellipses result from the linear errors:
$0.04s = 0.04, 0.08, \ldots, 0.4$ m,
while the major radii of the error ellipses result from the angular errors:
$d \sin(0.04 \cdot 360^\circ) = d \sin 14.4^\circ \approx 0.25, 0.50, \ldots 2.5$ m.
Clearly, the angular errors are much more significant than the linear errors.

Accurate Odometry and Error Modelling for a Mobile Robot

Kok Seng **CHONG**
kok.seng.chong@eng.monash.edu.au

Lindsay **KLEEMAN**
lindsay.kleeman@eng.monash.edu.au

Intelligent Robotics Research Centre (IRRC)
Department of Electrical and Computer Systems Engineering
Monash University, Clayton Victoria 3168, Australia
http://calvin.eng.monash.edu.au/IRRC/

Our work takes the assumption one step further. For a short unit of travel, the error is assumed to be zero mean, and white, that is, uncorrelated with the previous or next unit of travel. The variance of the cumulative error is then the sum of the variance of each statistically independent unit. This leads to a reasonable assumption that the variance of each unit of travel is proportional to the distance travelled

$$\sigma_L^2 = k_L^2 |d_L| \qquad \sigma_R^2 = k_R^2 |d_R| \tag{4}$$

$\sigma^2 \propto d$

where d_L and d_R are the distances travelled by each wheel, and k_L^2 and k_R^2 are constants with unit $m^{1/2}$.

SOLUTION FOR MORE ACCURACY;

1. BUY SUPER ACCURATE WHEEL ENCODERS* and TACHOMETER
2. IMPROVE ACCURACY BY COMBINING RESULTS WITH ANOTHER SENSOR MEASUREMENT
3. BE REALLY CAREFUL IN MAKING ROBOT MEASUREMENTS
4. BE PRECISE ABOUT INITIAL ORIENTATION

SADLY, THE ERROR CAN NEVER BE REDUCED TO ZERO!

* ACCURACY STILL SUFFERS FROM WHEEL SLIPPAGE AND OTHER RANDOM ERRORS.

Suppose that a robot moves with a constant *velocity* of 10 cm/s for a period of *time* of 5 s.[1] The *distance* it moves is 50 cm. In general, if a robot moves at a constant velocity v for a period of time t, the distance it moves is $s = vt$. When power is applied to the motors it causes the wheels to rotate, which in turn causes the robot to move at some velocity. However, we cannot specify that a certain power causes a certain velocity:

- No two electrical or mechanical components are ever precisely identical. A motor is composed of magnets and electrical wiring whose interaction causes a mechanical shaft to rotate. Small differences in the properties of the magnet and wire, as well as small differences in the size and weight of the shaft, can cause the shafts of two motors to rotate at slightly different speeds for the same amount of power.

- The environment affects the velocity of a robot. Too little friction (ice) or too much friction (mud) can cause a robot to move slower in comparison with its movement on a dry paved surface.

- External forces can affect the velocity of a robot. It needs more power to sustain a specific velocity when moving uphill and less power when moving downhill, because the force of gravity decreases and increases the velocity. Riding a bicycle at a constant velocity into the wind demands more effort than riding with the wind, and a cross-wind makes the relation between power and velocity even more complicated.

Good feedback as in a Cruise Control can help with the last point.

REMEMBER: ELECTROMAGNETIC INTERFERENCE and DATA ACQUISITION ERRORS MAY LEAD TO LESS ACCURACY.

DO YOU REALLY WANT TO KNOW ABOUT ODOMETRY ERROR AND PROPAGATION?

Nikolaus Correll 673 subscribers

The error propagation law explained in one and multiple dimensions using odometry as the running example. Matlab code is available on github.

https://github.com/correll/Introducti...

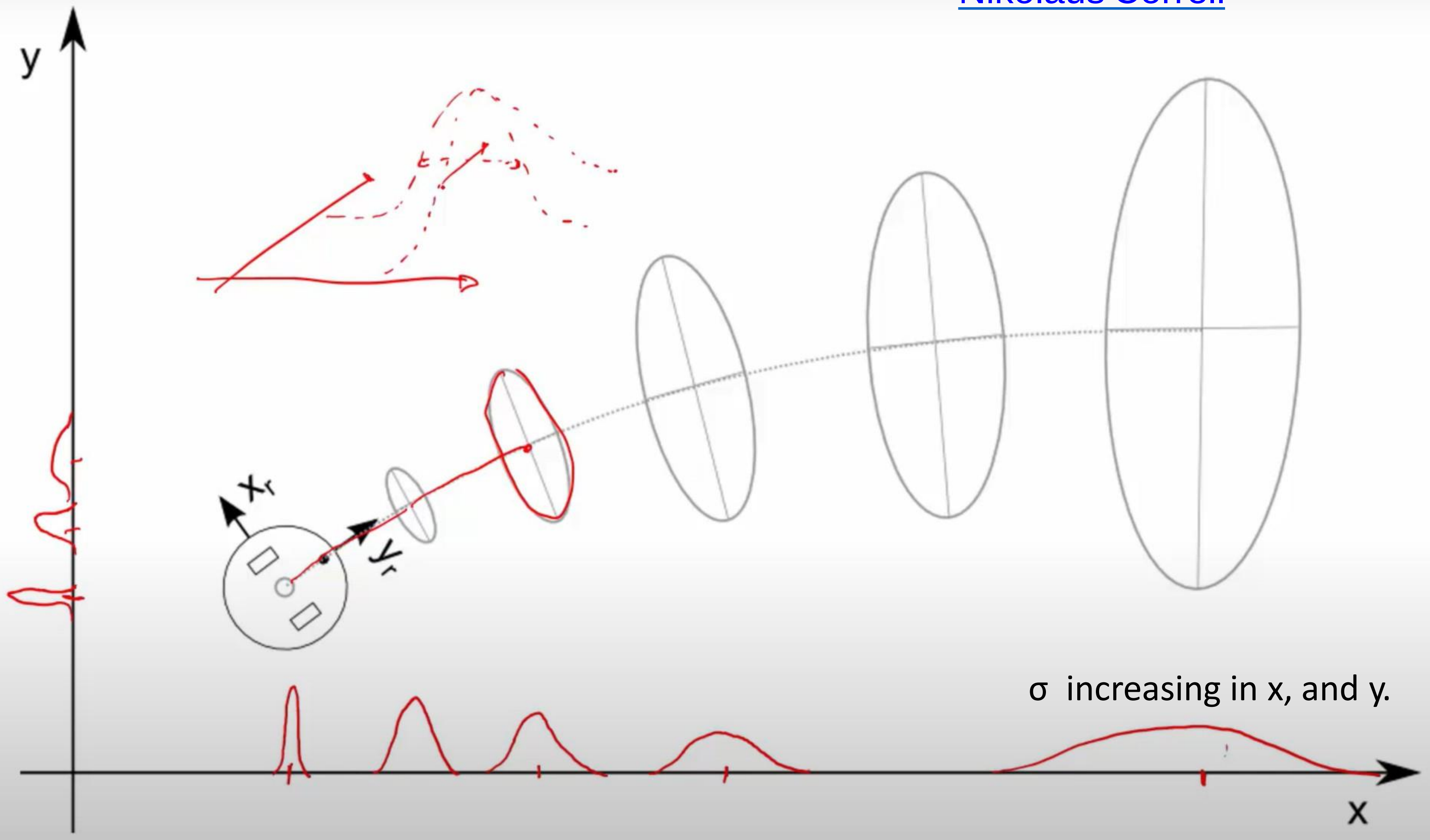
https://www.youtube.com/watch?v=ubg_AAM7Zd8 29:20

https://github.com/Introduction-to-Autonomous-Robots/Introduction-to-Autonomous-Robots/tree/master/matlab

At 23:11 Correll presents the MATLAB code.

Error propagation law: Odometry example

[Nikolaus Correll]



Nikolaus Correll

Example: Odometry

$$\Delta x = \Delta s cos(\theta + \Delta\theta/2)$$
$$\Delta y = \Delta s sin(\theta + \Delta\theta/2)$$
$$\Delta\theta = \frac{\Delta s_r - \Delta s_l}{2} \qquad \Delta s = \frac{\Delta s_r + \Delta s_l}{2}$$

$$f(x, y, \theta, \Delta s_r, \Delta s_l) = [x, y, \theta]^T + [\Delta x \quad \Delta y \quad \Delta\theta]^T$$

$$\Sigma_{p'} = \nabla_p f \Sigma_p \nabla_p f^t + \nabla_{\Delta_{r,l}} f \Sigma_\Delta \nabla_{\Delta_{r,l}} f^T$$

Error propagation law

Motion Component

Wheelslip component

$$\nabla_p f = \left[\frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y} \quad \frac{\partial f}{\partial \theta}\right] = \begin{array}{c} \begin{array}{ccc} \Delta x & \Delta y & \Delta\theta \end{array} \\ \begin{bmatrix} 1 & 0 & -\Delta s sin(\theta + \Delta\theta/2) \\ 0 & 1 & \Delta s cos(\theta + \Delta\theta/2) \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$

$$\nabla_{\Delta_{r,l}} f = \begin{array}{c} \begin{array}{cc} \Delta s_r & \Delta s_L \end{array} \\ \begin{bmatrix} \frac{1}{2}\cos(\theta + \Delta\theta/2) & \frac{1}{2}\cos(\theta + \Delta\theta/2) \\ \frac{1}{2}\sin(\theta + \Delta\theta/2) & \frac{1}{2}\cos(\theta + \Delta\theta/2) \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \end{array}$$

$$\Sigma_p = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad \Sigma_\Delta = \begin{pmatrix} k|\Delta s_R| & 0 \\ 0 & k|\Delta s_L| \end{pmatrix}$$

TEST TO FIND THE k'S

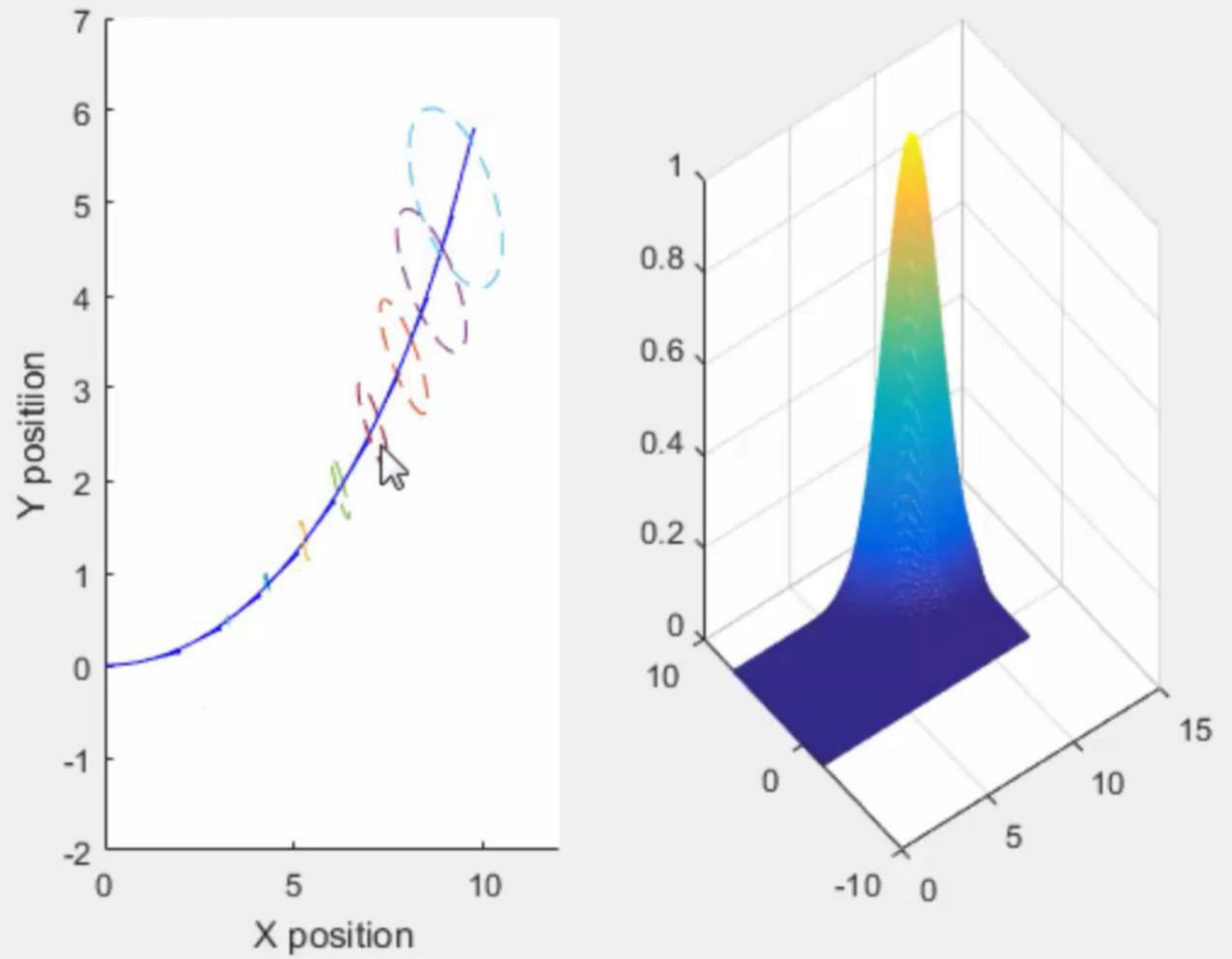
KNOWN INITIAL POSITION

Editor - C:\Users\Administrator\Deskt

plot_trajectory.m

```
for I=1:length(dsl),
  oldx=x;
  oldy=y;
  [x,y,theta,Cp]=pos_

  % Plotting
  subplot(1,2,1);
  plot([oldx x],[oldy
  if(mod(I,10)==0)
   quiver(x,y,cos(the

   Ellipse_x=Cp(1,1).
   Ellipse_y=Cp(2,2).
   plot(x+Ellipse_x.*                                  eta),'--');

  end;
  subplot(1,2,2);
   [X,Y]=meshgrid(lin
   XR = cos(theta)*(X
   YR = -sin(theta)*(


   G=Gauss2D(XR,YR,x,y,Cp(1,1),Cp(2,2));
   mesh(linspace(0,12),linspace(-2,7),G);
   drawnow;
end;
```

Figure 1



[Nikolaus Correll]

Command Window

Aaron Becker UH Robotics

Prelude to Kalman Filter: Combining Uncertain Measurements

444 views •Premiered Dec 24, 2019   4:45

https://www.youtube.com/watch?v=-zOmHUENLJE

Leonardo da Vinci often made maps, both for military purposes and for canal construction. He therefore designed several distance-recording devices, including an odometer. This is one of several variations on an instrument described by the Roman architect and engineer, Vitruvius, whose works were rediscovered early in the Renaissance. It was geared to drop a pellet into a box for a given number of revolutions of a wheel, thus computing the distance traveled.