Sensor Fusion and Filtering

or

"Making sensors make sense"

Paul Pounds

19 March 2013

University of Queensland

But first…

Some house keeping

Calendar at a glance

You are here

OMG!

Week	Dates	Lecture	Reviews	Demos	Assessment submissions
1	25/2 – 1/3	Introduction			
2	4/3 – 8/3	Principles of Mechatronic Systems design			
3	11/3 – 15/3	Principles of Sailing			Design brief
4	18/3 – 22/3	Sensor Fusion and Filtering	Progress review 1		
5	25/3 -29/3	???			
Break	1/4 – 5/4				
6	8/4 – 12/4	By request	Progress seminar		
7	15/4 – 19/4	By request		25% demo	
8	22/4 – 26/4				
9	29/4 – 3/5		Progress review	50% demo	
10	6/5 – 10/5				
11	13/5 – 17/5			75% demo	Preliminary report
12	20/5 – 24/5				
13	27/5 – 31/5	Closing lecture		Final testing	Final report and addendum

FAQ Roundup

- **Do rudders count towards the hull dimensions?**
  - No – they can extend beyond the 150 mm x 75 mm bounding box (but then will be invalid for scoring).

Next week's lecture

- Nobody nominated anything.
  - Seriously? Why would you not do that?
- Ok, ok – don't panic. We can fix this.
- I propose to instead run a best-practices soldering tutorial during the lecture time
  - Because your soldering is terrible (probably).

Progress Review

- Show you have been doing stuff!
  - You will have 3-5 minutes to demonstrate your contribution to the team
- Bring evidence!
  - Sketches, notes, prototypes, analysis, work breakdowns, etc. are all good.
- Pass/fail assessment
  - It should be difficult to fail this if you have actually done something useful

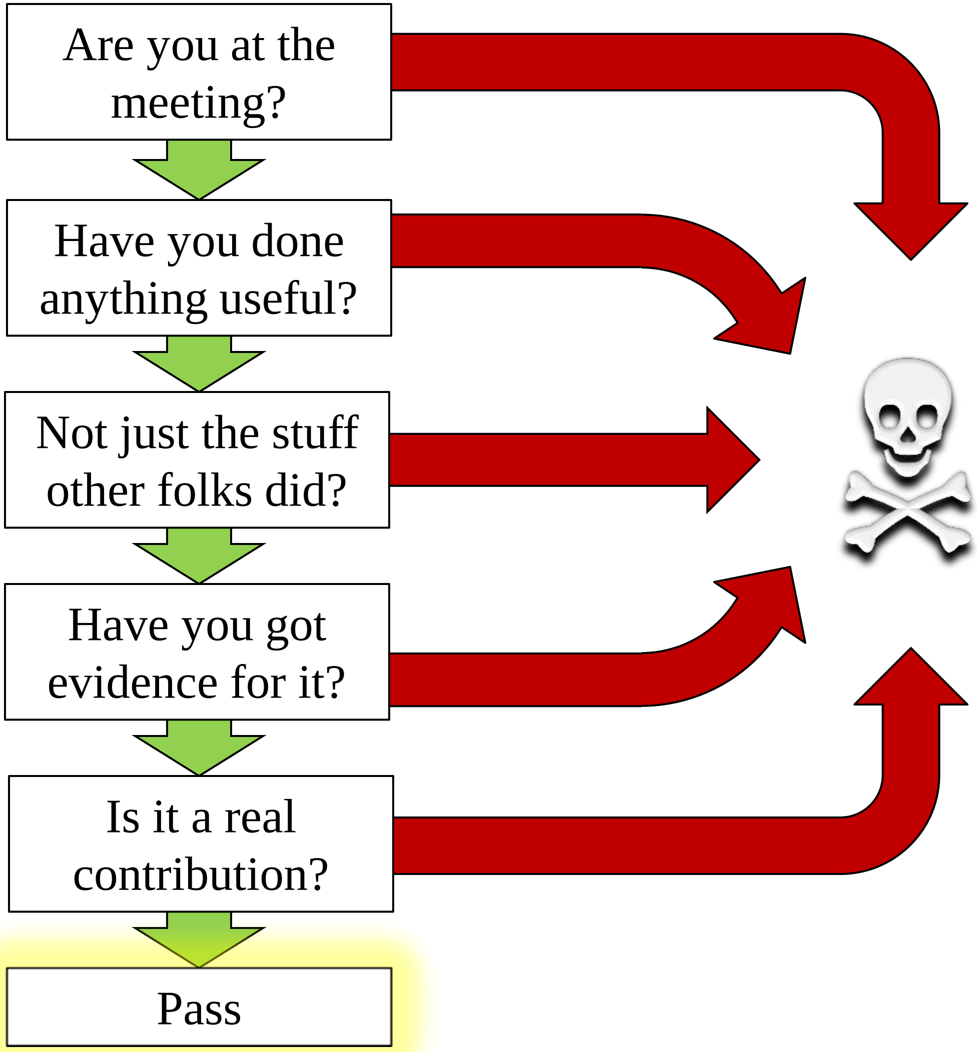
Progress Review sessions

Group times:

Wed 20		Thursday 21	
~~Group ?~~	~~9:00-9:30~~	Group 13	9:00-9:30
Group 15	9:30-10:00	Group 11	9:30-10:00
Group 8	10:00-10:30	Group 10	10:00-10:30
Group 14	10:30-11:00	Group 5	10:30-11:00
Group 4	14:00-14:30 A	Group 3	13:00-13:30
Group 2	14:00-14:30 B	Group 6	13:30-14:00
Group 1	14:30-15:00 A	Group 9	14:00-14:30
Group 7	14:30-15:00 B	Group 12	14:30-15:00

Group B sessions are held in Axon 211, all other sessions are in GPS 310

Progress Review flow chart



Are you at the meeting?

Have you done anything useful?

Not just the stuff other folks did?

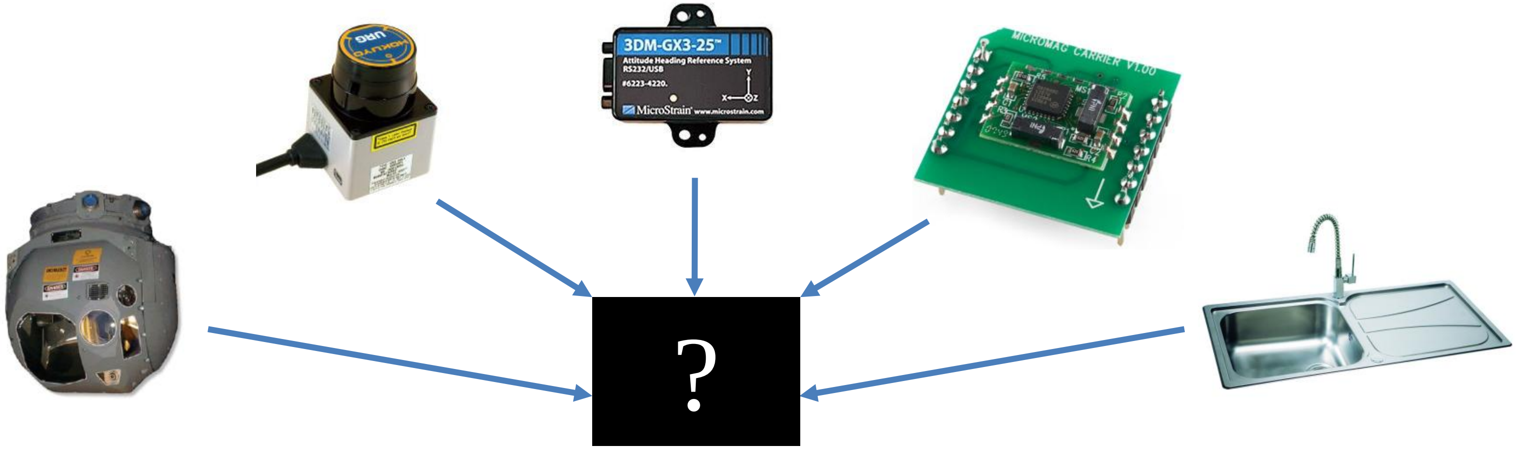
Have you got evidence for it?

Is it a real contribution?

Pass

Onwards to sensor fusion

To fusion and beyond!!



What is this sensing stuff?

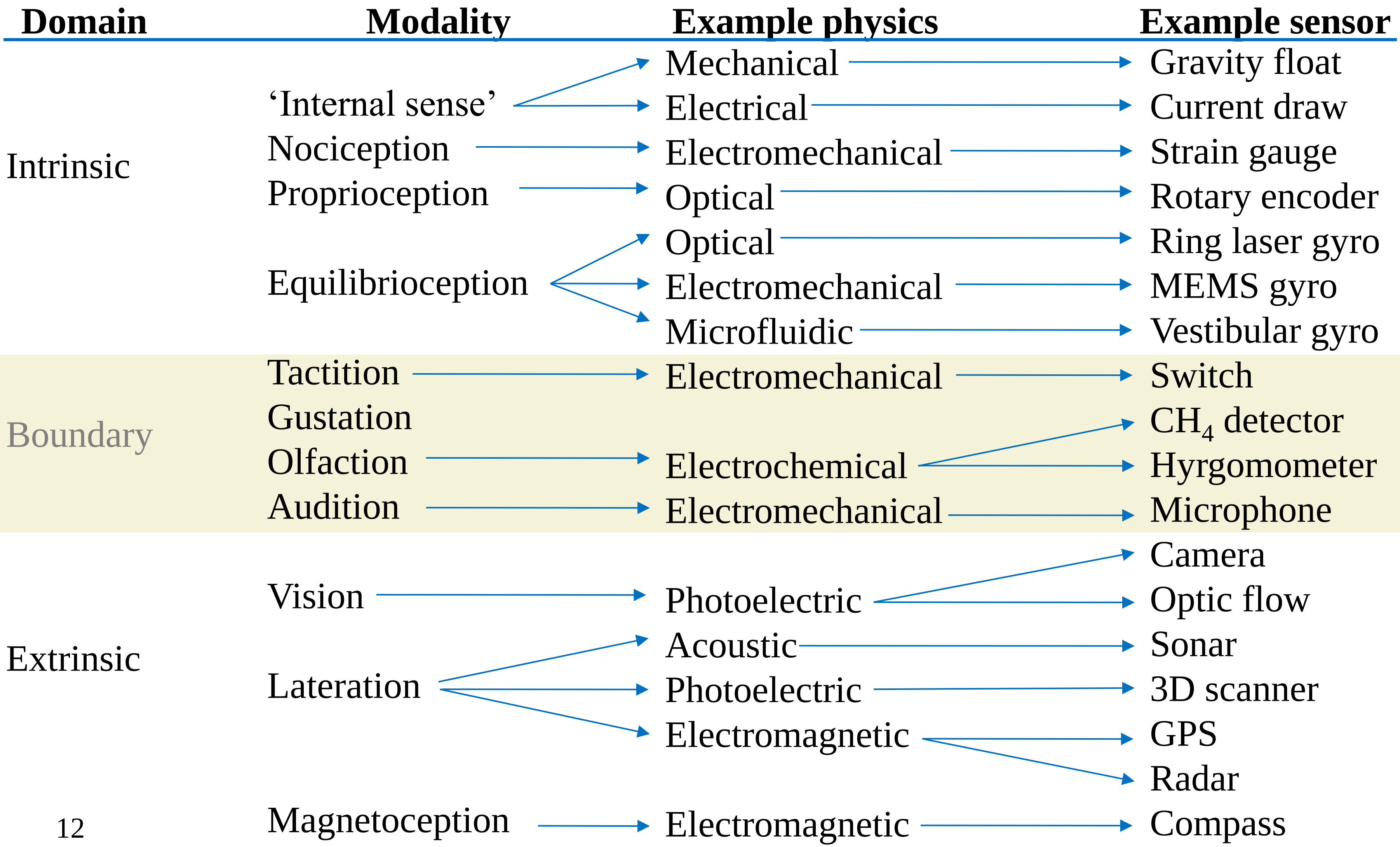
- How systems find out about the world
  - And themselves
- Sensing is the measurement of some physical property of the environment
  - Physical property is analogous to the measurand, related to the state value of interest
  - Physical signal is typically transduced into an electrical signal (and often digitised)

Snuh?

- Sensors use a physical property to produce a signal related to the thing being measured.

Right.

An incomplete sensor taxonomy



Domain	Modality	Example physics	Example sensor
Intrinsic	‘Internal sense’	Mechanical	Gravity float
		Electrical	Current draw
	Nociception	Electromechanical	Strain gauge
	Proprioception	Optical	Rotary encoder
	Equilibrioception	Optical	Ring laser gyro
		Electromechanical	MEMS gyro
		Microfluidic	Vestibular gyro
Boundary	Tactition	Electromechanical	Switch
	Gustation		CH_4 detector
	Olfaction	Electrochemical	Hyrgomometer
	Audition	Electromechanical	Microphone
Extrinsic	Vision	Photoelectric	Camera
			Optic flow
	Lateration	Acoustic	Sonar
		Photoelectric	3D scanner
		Electromagnetic	GPS
			Radar
	Magnetoception	Electromagnetic	Compass

More deeply

- Measurement is an attempt to find the true value of some real state parameter
  - Measurements and true states generally differ

- For practical, entropic, budgetary and philosophical reasons, no sensor is perfect.
  - Some are merely ‘adequate’.

Quick terminology

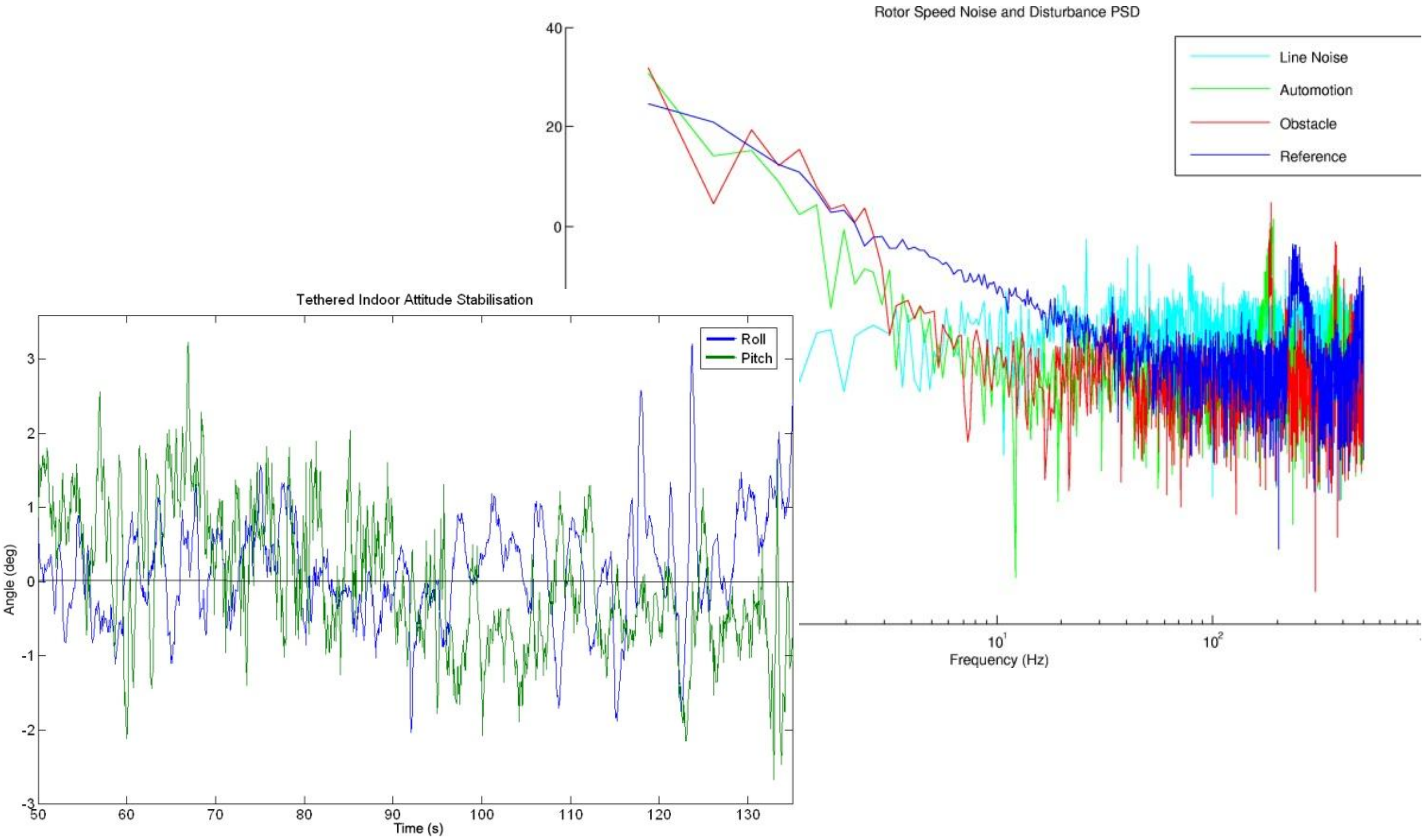
- State: numeric parameterisation of a thing
- Signal: time-varying state function that conveys information
- True state: real state being sought
- Measurement: signal from a sensor
- Estimate: inferred guess of the true state

Signals imperfect

- Measurement signals can be thought of as containing information about the true value

- Confounding effects obscure information
  - Entropic noise (eg. thermal noise)
  - Coupled noise/cross-talk
  - Bias
  - Nonlinearity

Postcards from the front



Filtering

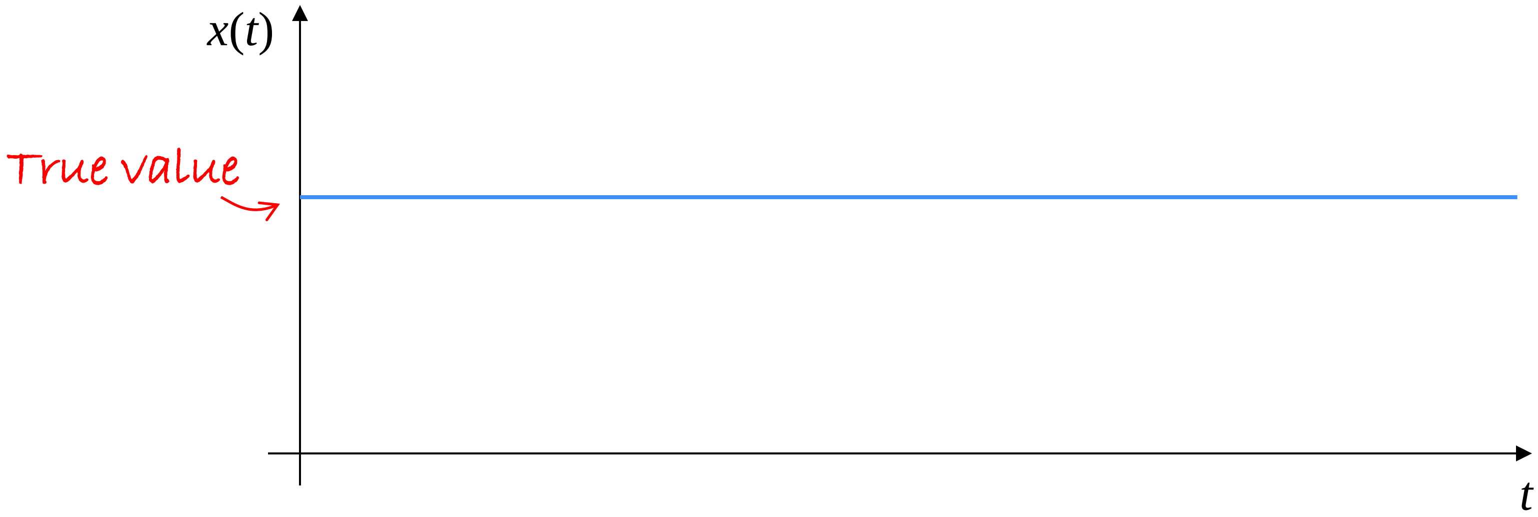
- Filters act on signals to remove confounding effects and extract information

- Many common examples:
  - – Low pass filter: remove high frequency noise
  - – High pass filter: remove low frequency bias
  - – Common-mode filter: remove line coupling

Use multiple measurements of a signal in time to estimate the true state – temporal diversity

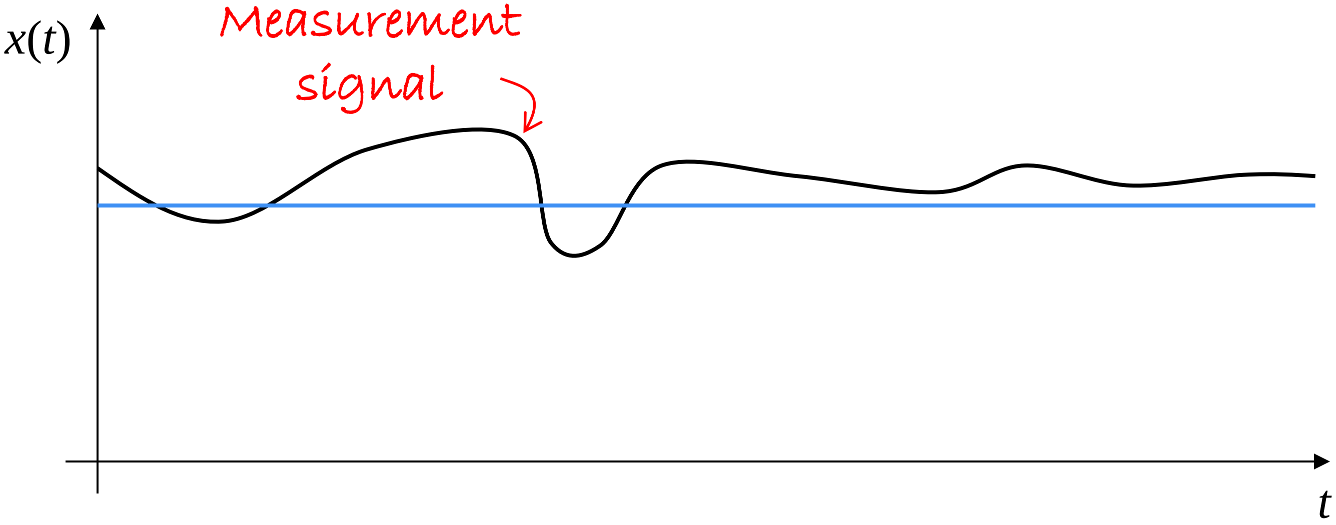
The intuitive idea

- Given time-history measurements of a state, what is its ‘most likely’ true value?



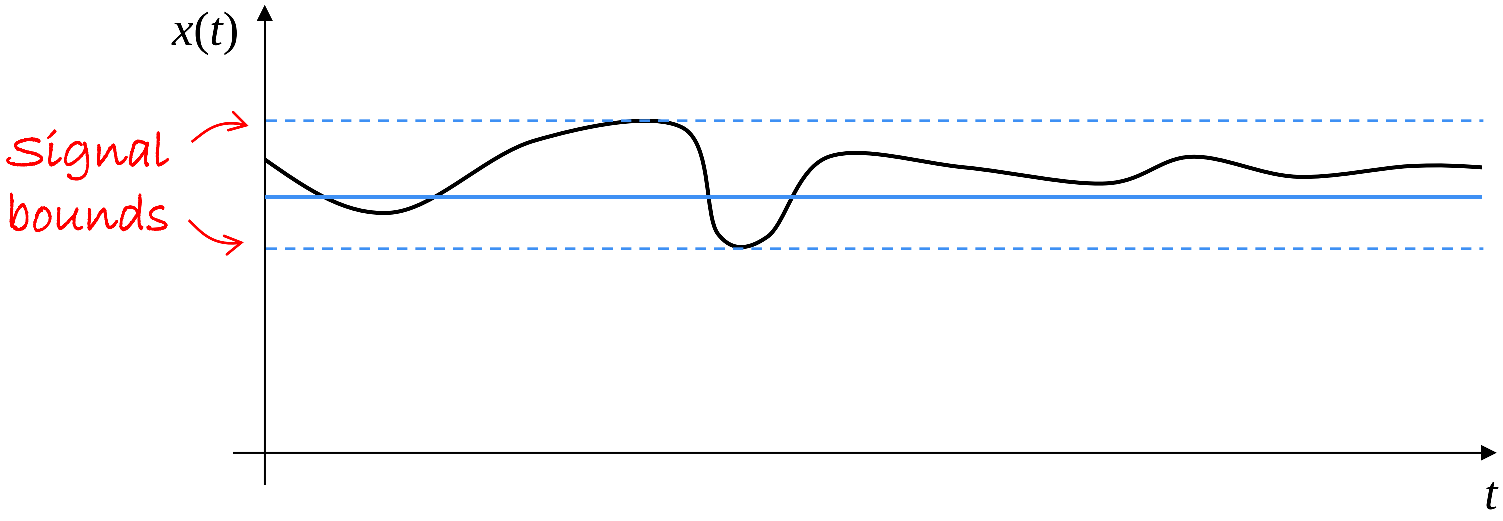
The intuitive idea

- The time-history of the measurement signal conveys information of the true value



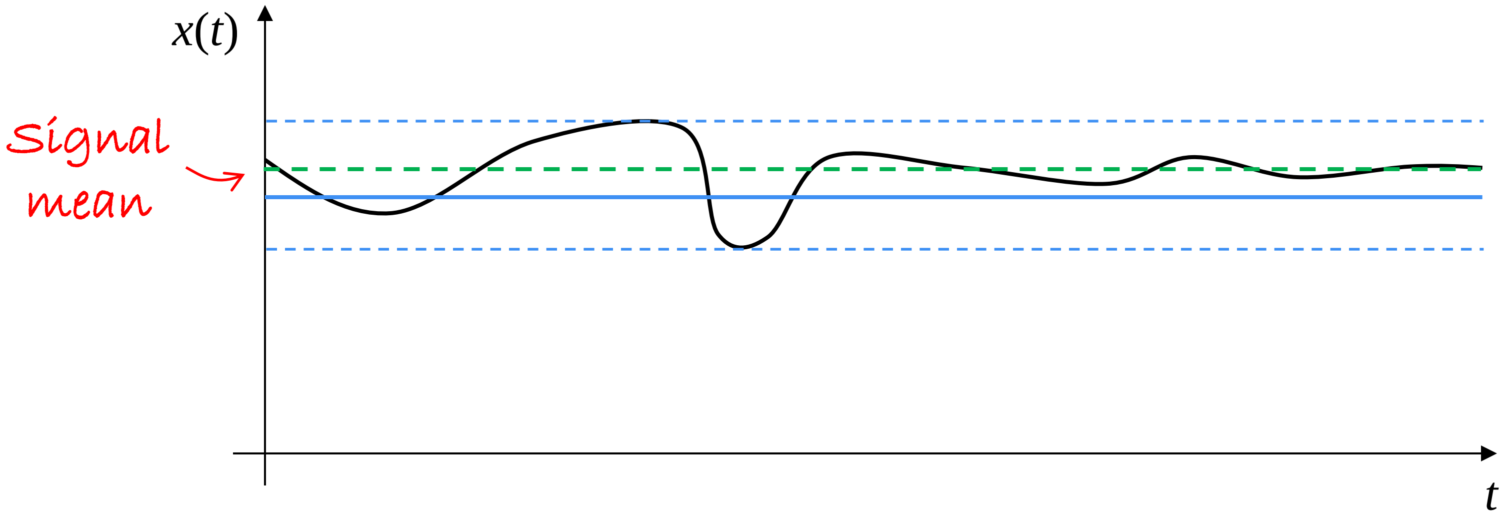
The intuitive idea

- Examining temporal properties of the signal allow us to infer the true value



The intuitive idea

- Examining temporal properties of the signal allow us to infer the true value



Depressing truth

A simple RC or ‘exponential’ 1st order filter will solve 80% of your practical noise elimination or smoothing problems*

- Quick, easy, and a snap to code/build!

*The other 20% requires an engineering degree or two

Sensor fusion

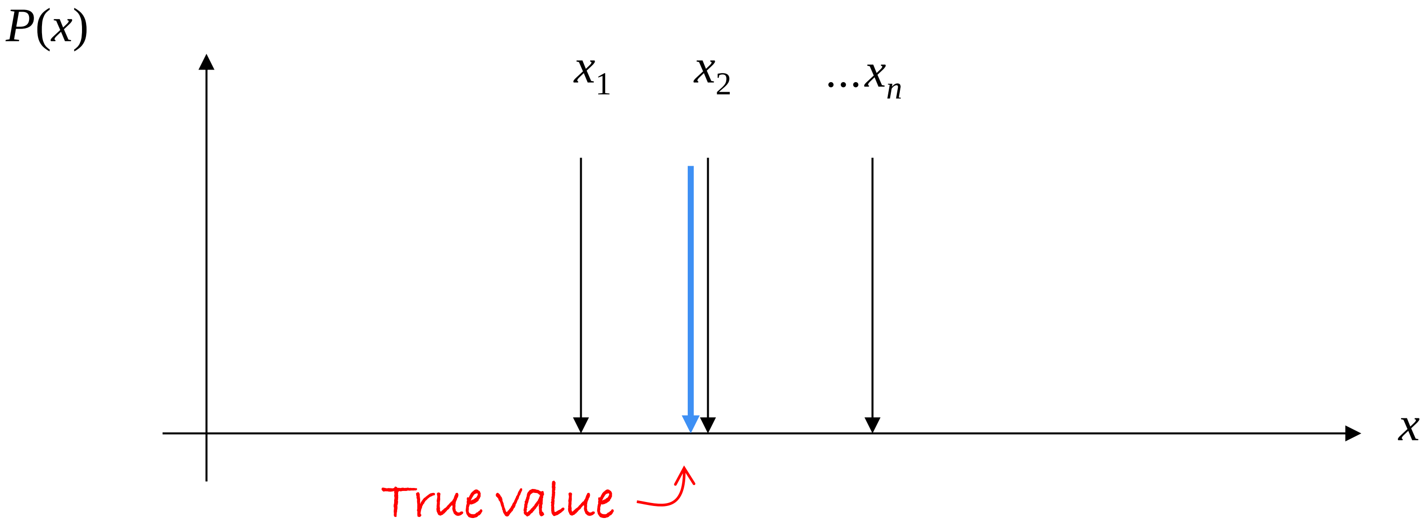
- Combination of multiple different sensors
  - Each sensor adds information (even poor ones), allowing for more accurate estimates
  - Sensors must measure the same states, or states relatable through a system model

eg. compass and heading gyro to estimate bearing

Use multiple sensor modalities to estimate the true state – measurement diversity

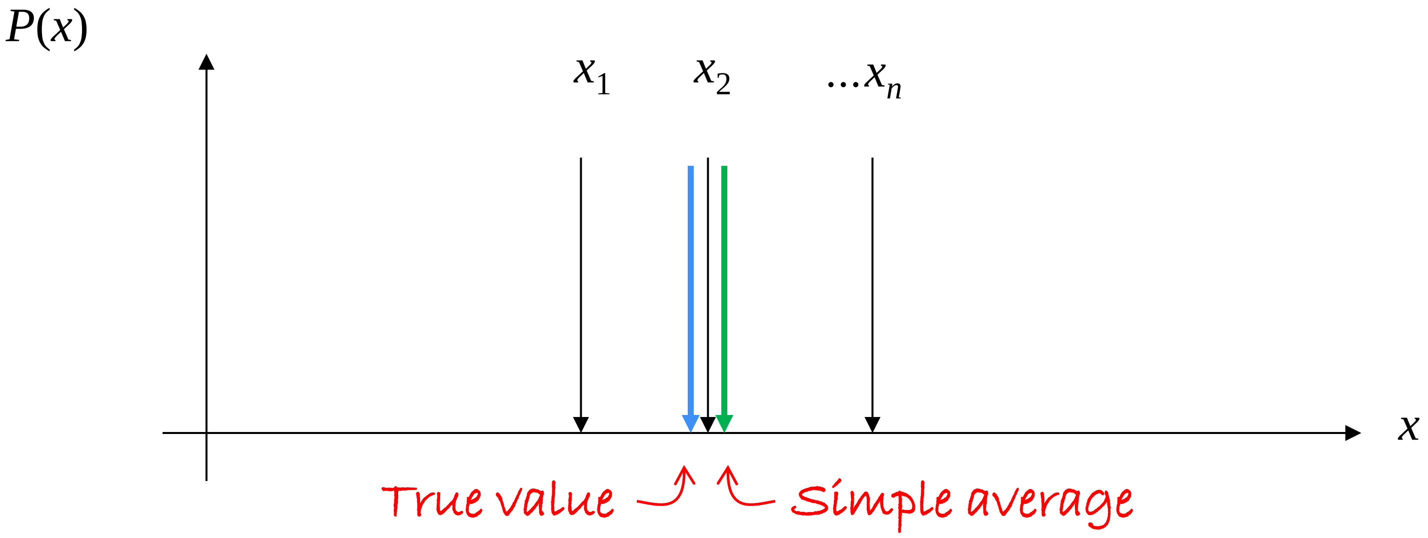
The intuitive idea

- *n* sensors can produce more accurate estimates together than possible individually



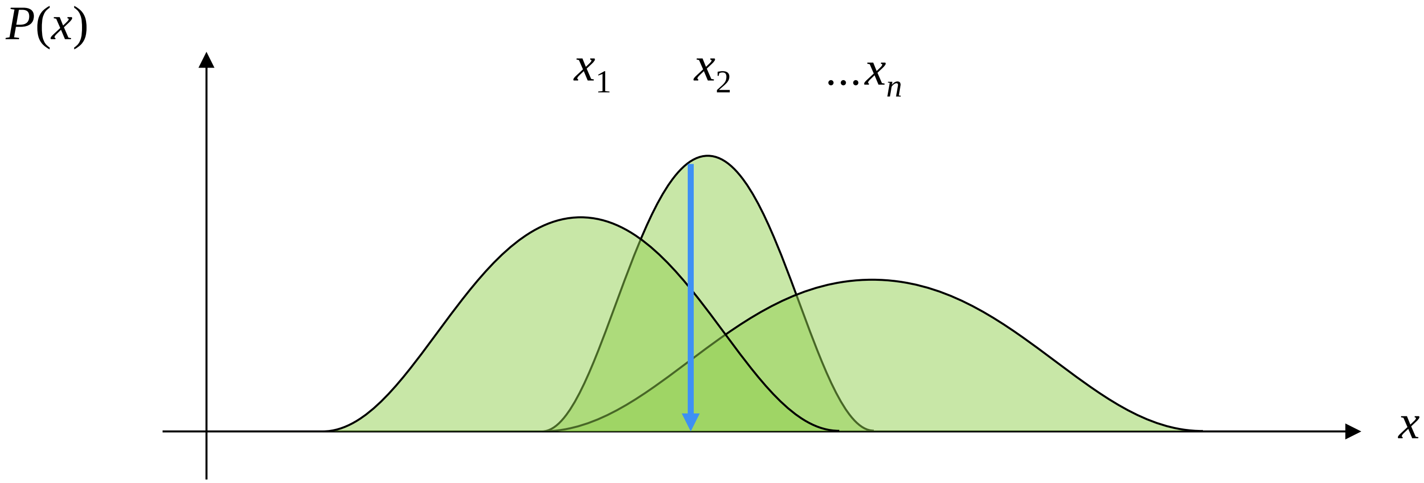
The intuitive idea

- *n* sensors can produce more accurate estimates together than possible individually



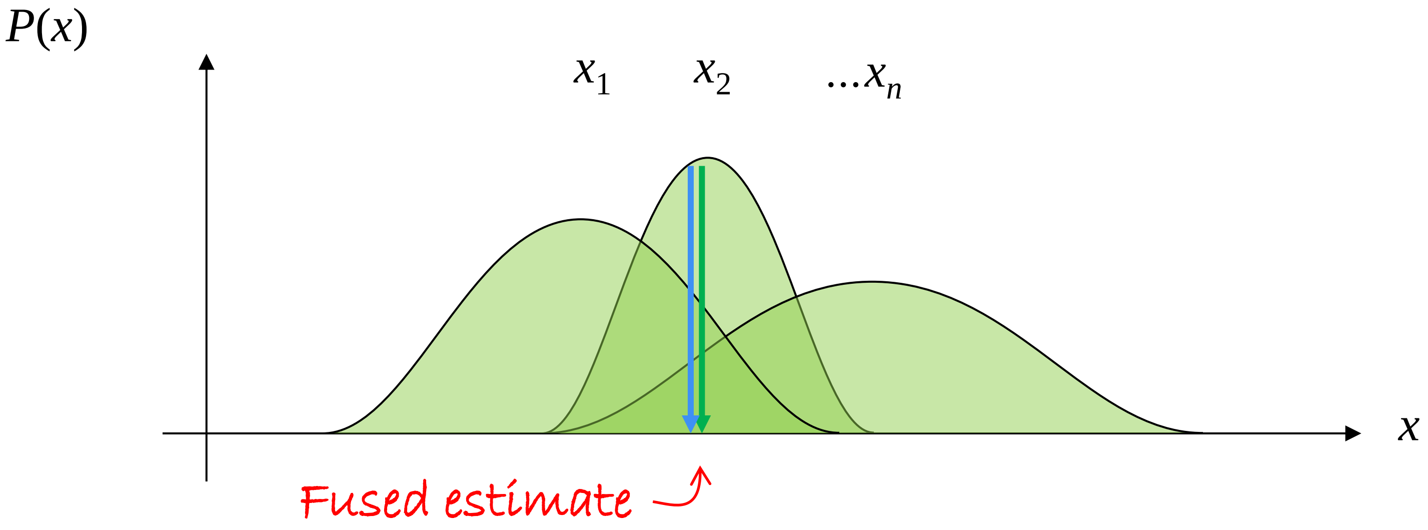
The intuitive idea

- Sensor mean and variance encode more information than an instantaneous sample



The intuitive idea

- Can incorporate stochastic signal behaviour



Fundamental requirement

- Filtering/fusion tacitly assume we know something about the signal being sought:
  - Frequency band of true value/noise
  - Relative amplitude/power
  - Waveform shape or encoding

- Use knowledge to isolate known properties of the signal and suppress spurious effects

(Naturally, you can also filter fused estimates)

Ergodic principle

- Over long enough time scales, a constant signal is equal to its mean
  - In ergodic dynamical systems of constant energy, all states are visited equally often
- Holds in the case of normal (Gaussian) distributions
  - This is why averaging works – static 'window'
  - Not necessarily the case for 'coloured' noise

That’s great… but?

- How do we get that fused estimate?
- Several ways:
  - Weighted averaging
  - Least squares
  - Kalman filter
  - Particle filter
  - Complementary filter
  - Linear observer
  - Bayesian network
  - Dempster-Shafer

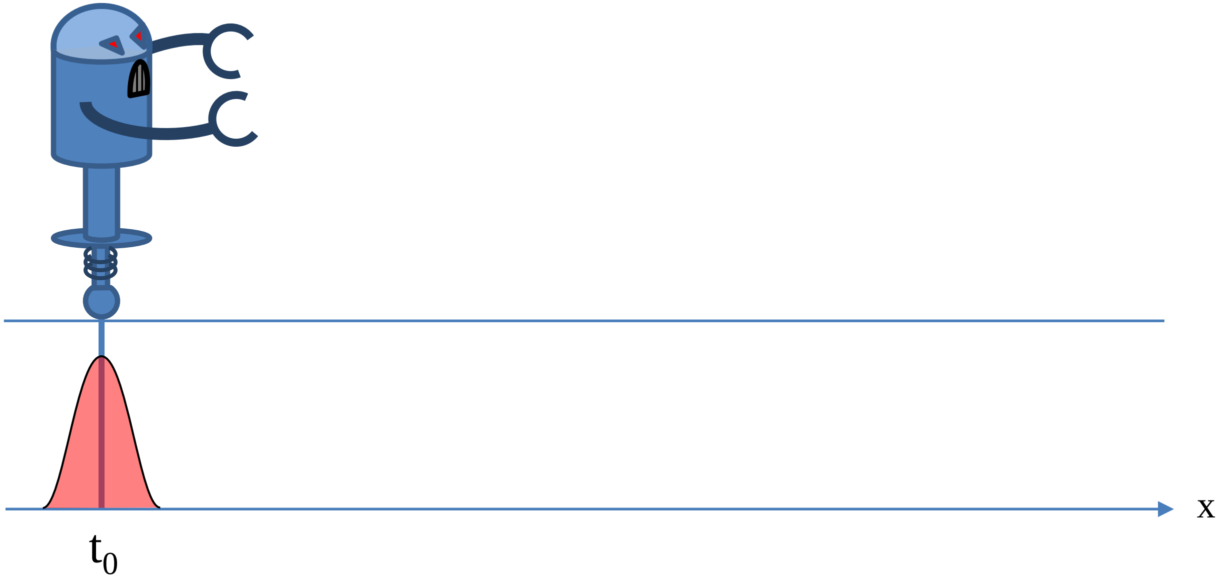
Key point: subsequent measurements cause estimates (and their variance) to converge

The Kalman filter

- Suppose we have a noisy measurement of our current state, with some estimate of variance
- If we know the system dynamics we can guess what the next state will be (with variance)
  - Compare where we think we should be to where our sensors tell us we are
  - Take a weighted average of the two, based on their covariance, as the new state

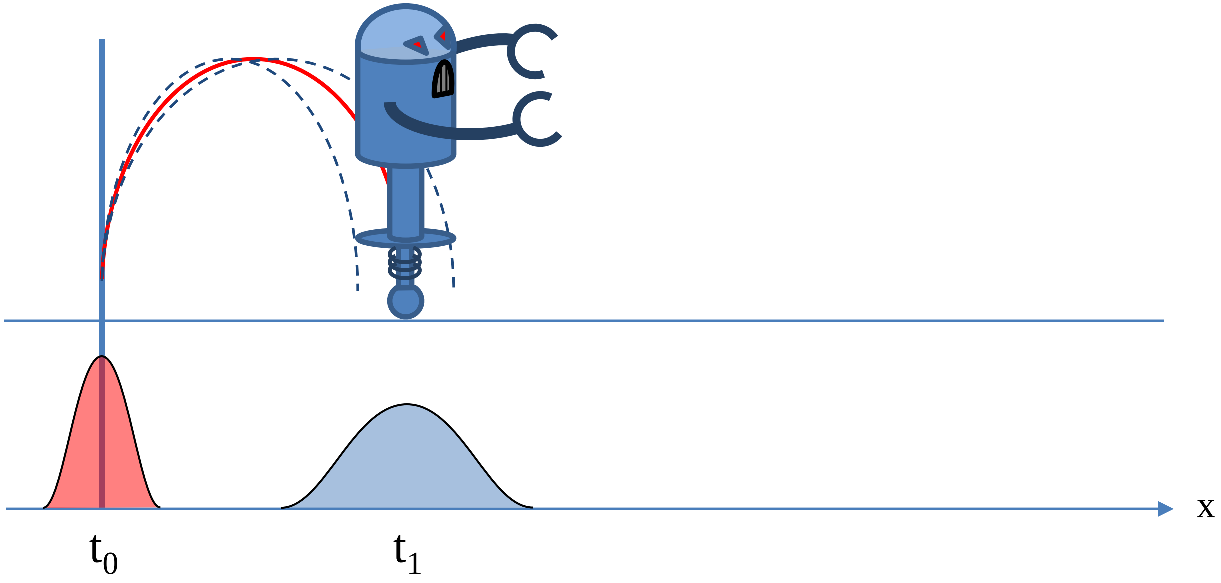
Eg. Pogo stick robot

- We know its approximate position and velocity



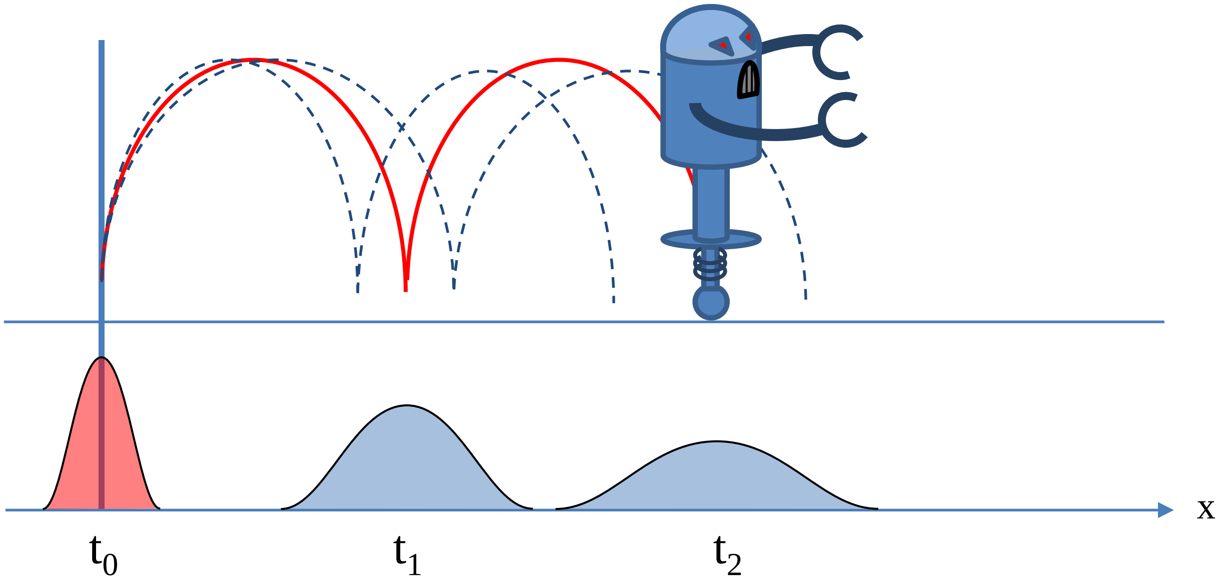
Eg. Pogo stick robot

- We know its approximate position and velocity
  - Using ballistics we can guess where it will land after each jump



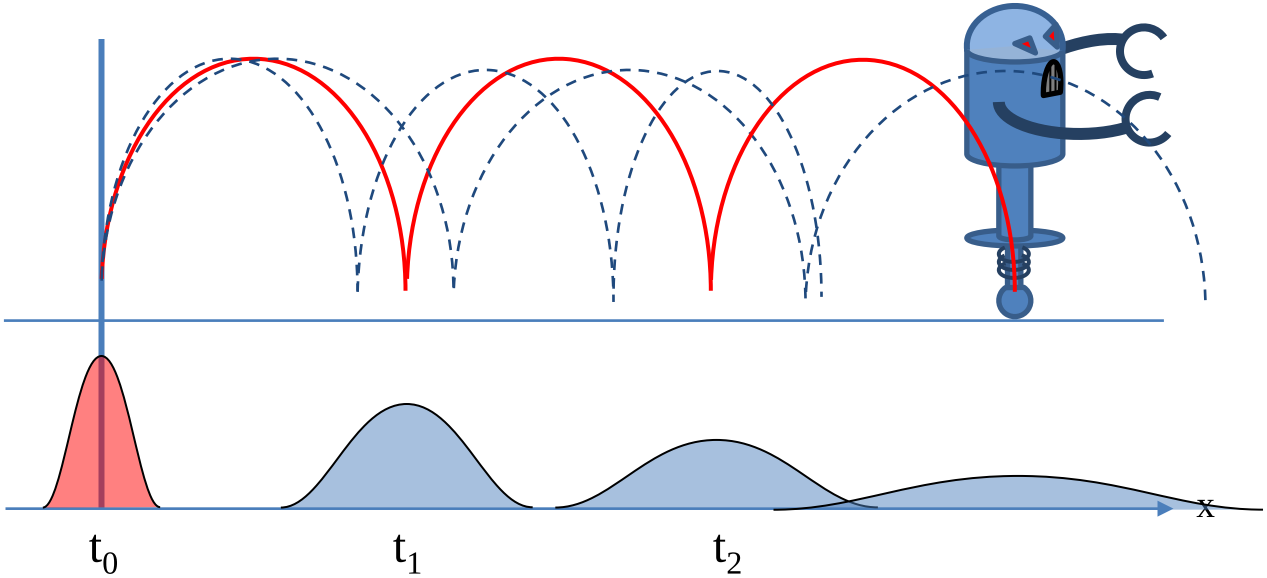
Eg. Pogo stick robot

- We know its approximate position and velocity
  - Using ballistics we can guess where it will land after each jump… becoming less certain with time



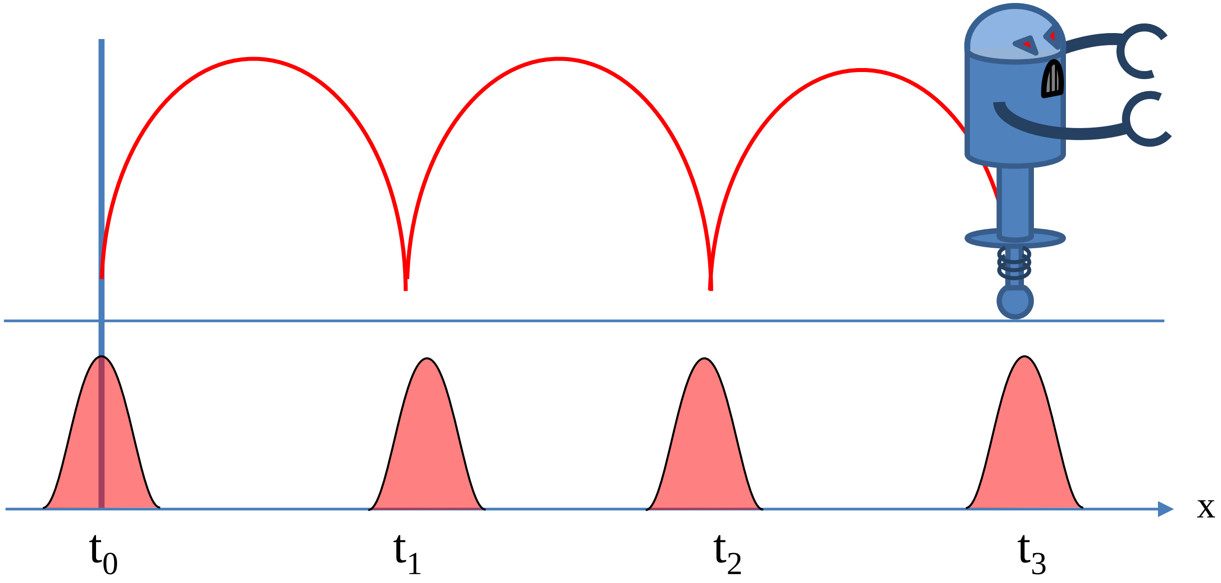
Eg. Pogo stick robot

- We know its approximate position and velocity
  - Using ballistics we can guess where it will land after each jump… becoming less certain with time



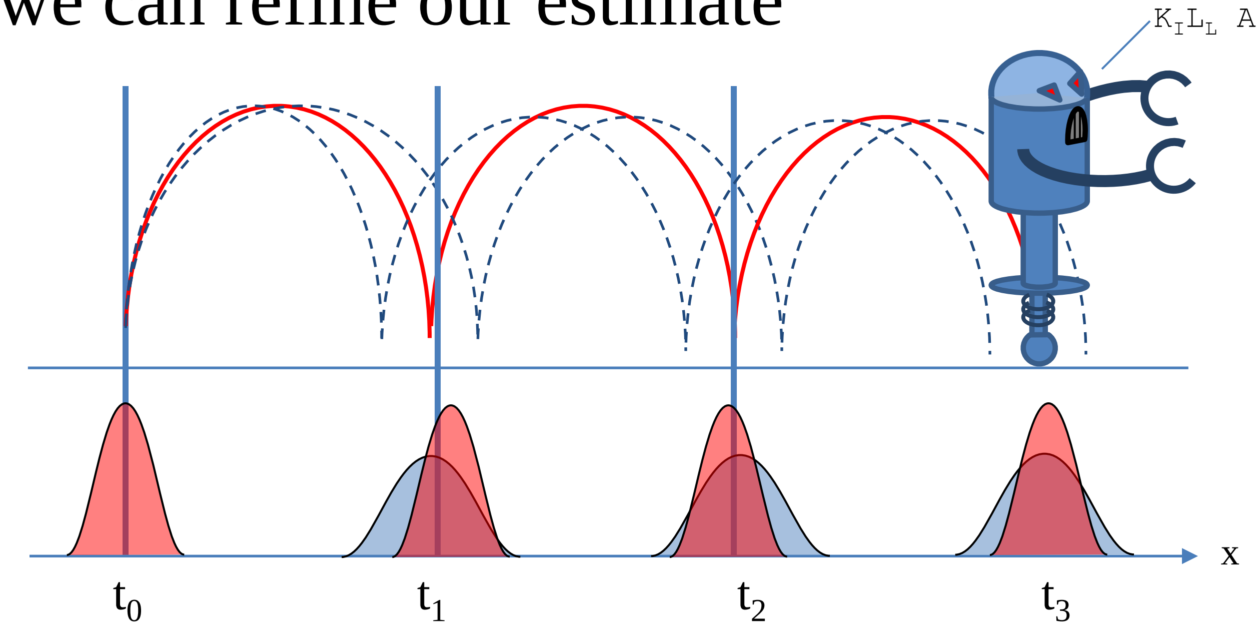
Eg. Pogo stick robot

- We know its approximate position and velocity
  - Using ballistics we can guess where it will land after each jump… becoming less certain with time
  - After each jump, we get a new measurement and we can refine our estimate



Eg. Pogo stick robot

- We know its approximate position and velocity
  - Using ballistics we can guess where it will land after each jump… becoming less certain with time
  - After each jump, we get a new measurement and we can refine our estimate



Other approaches

- By now you should be completely sick of hearing about the Kalman filter
  - If not, go here:

    digi.physic.ut.ee/mw/images/d/d5/Poormankalman.pdf

- Let’s also look at the particle filter, the complementary filter and linear observer
  - Surya Singh also has a nice primer on estimators:

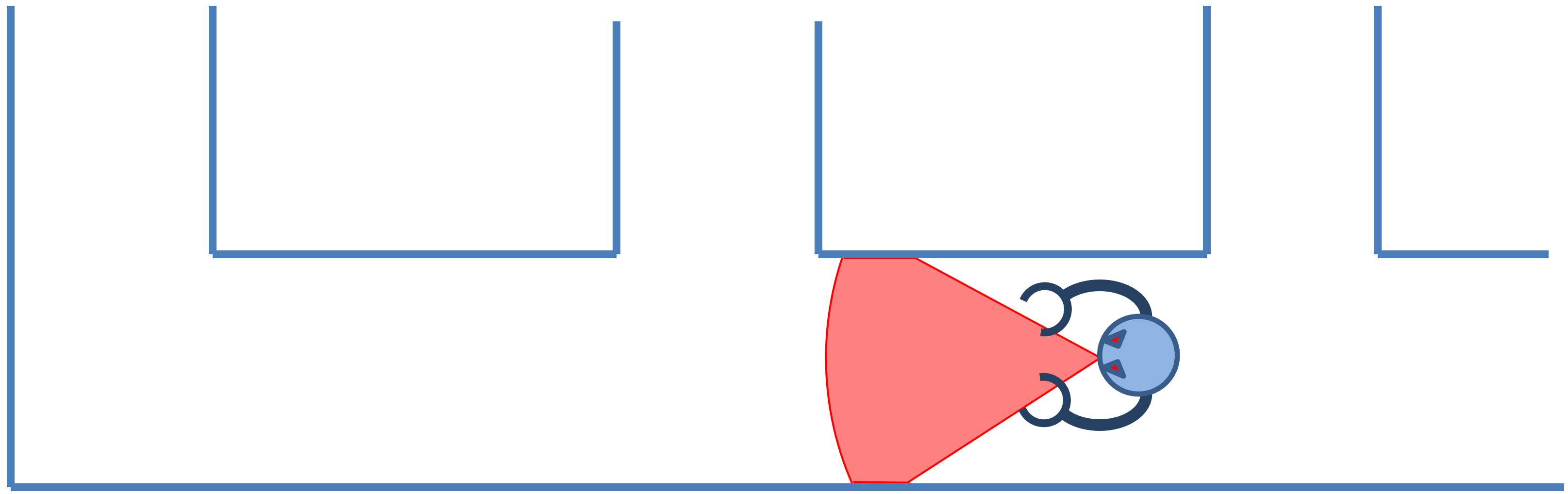
robotics.itee.uq.edu.au/~metr3800/doc/ClassNotes_METR3800.pdf

The particle filter

- Similar concept, but uses discretely sampled estimates of the space of possible states
  - Simulate each step in time - track the particles
  - Find out how much each subsequent measurement agrees with each particle
  - Use the ‘best’ particle as the estimate
  - Occasionally resample around the most reliable particle

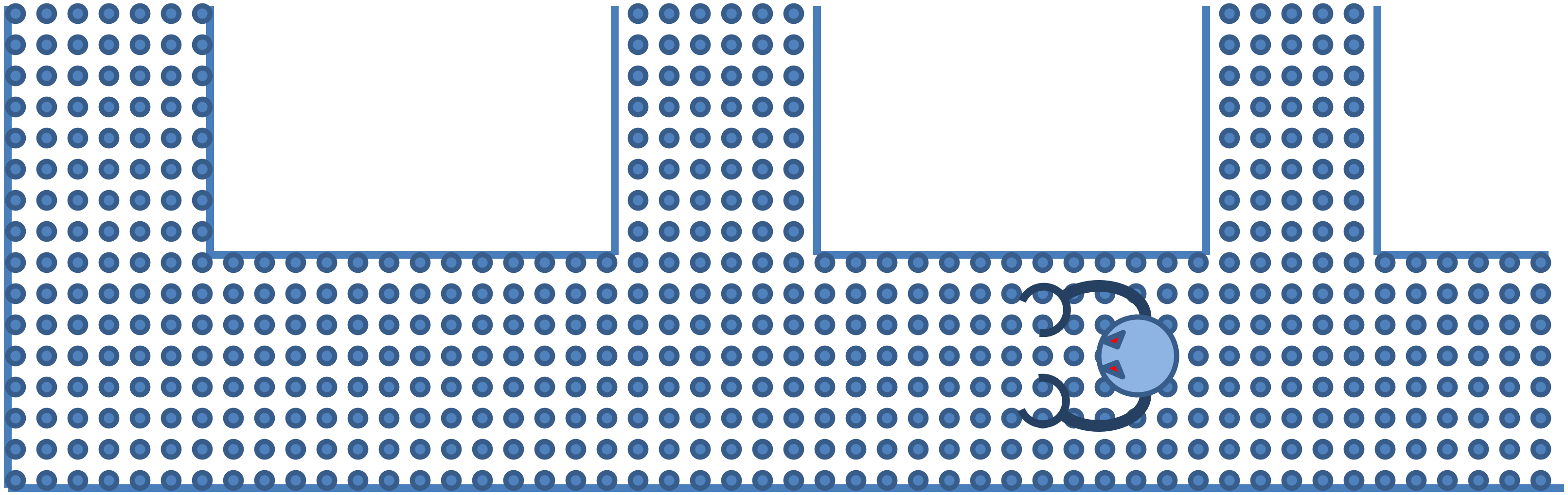
Eg. Kidnapped Robot Problem

- Robot moved to an unknown location



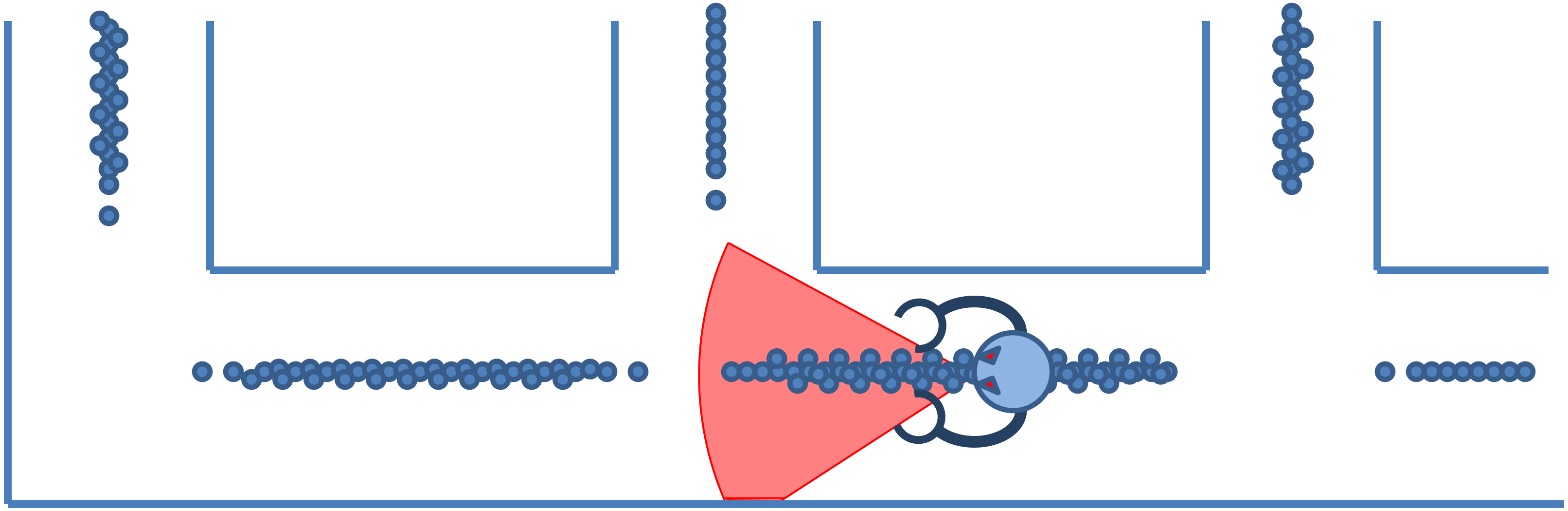
Eg. Kidnapped Robot Problem

- Robot moved to an unknown location
- Estimate we could be anywhere



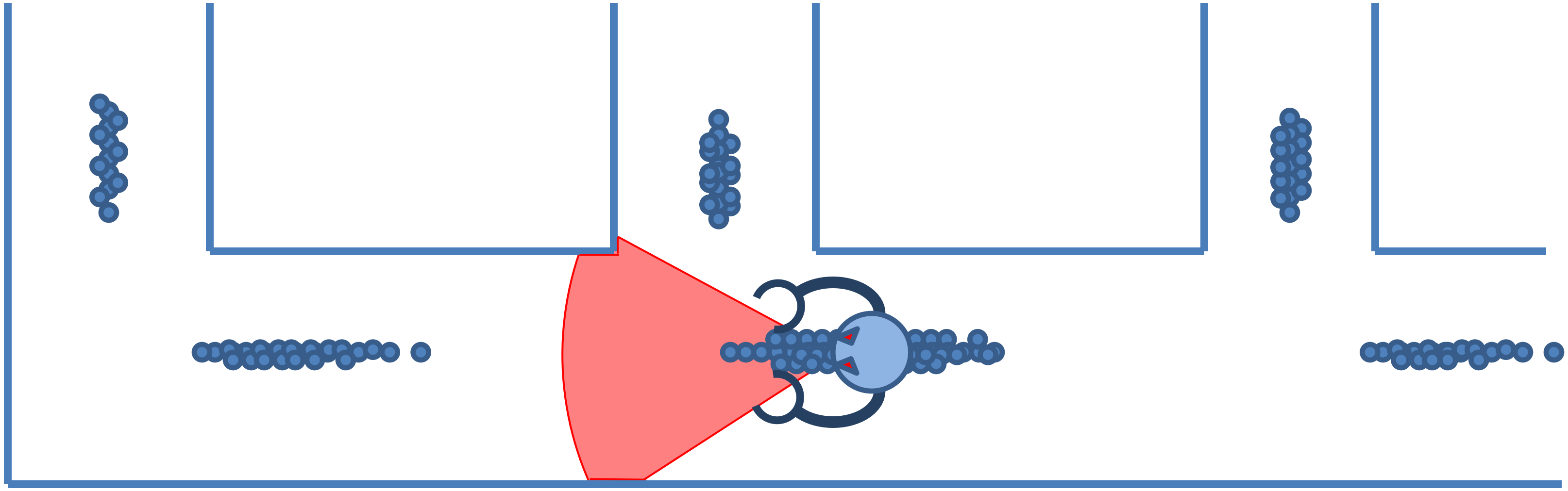
Eg. Kidnapped Robot Problem

- Robot moved to an unknown location
- Estimate we could be anywhere – subsequent observations reduce likely candidate positions



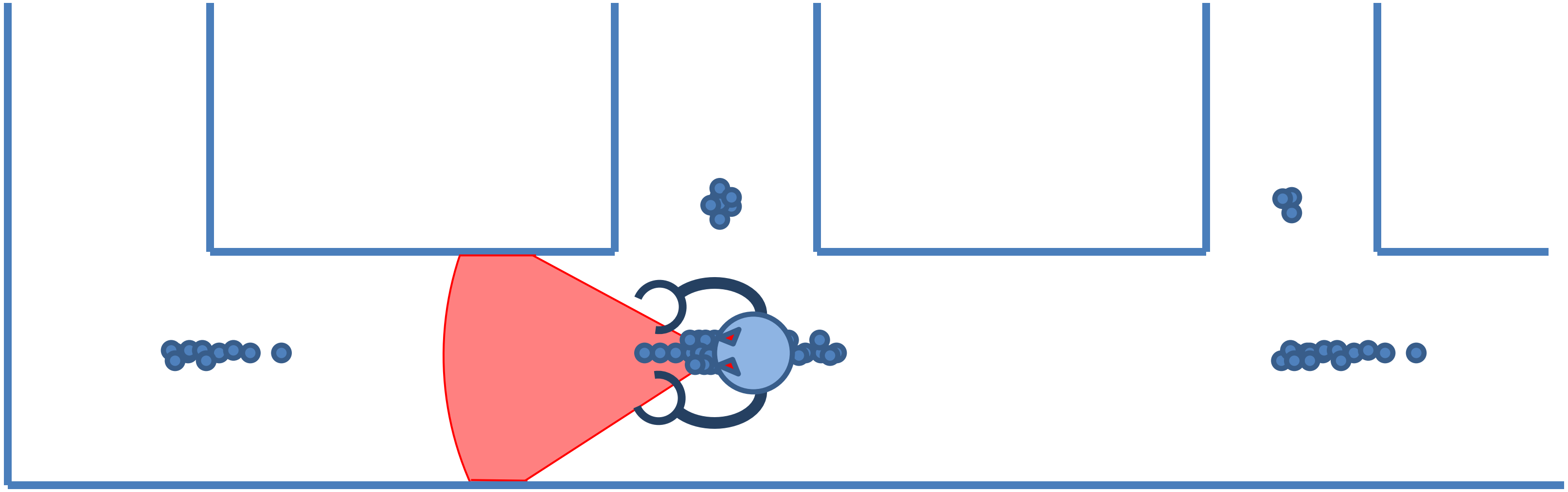
Eg. Kidnapped Robot Problem

- Robot moved to an unknown location
- Estimate we could be anywhere – subsequent observations reduce likely candidate positions



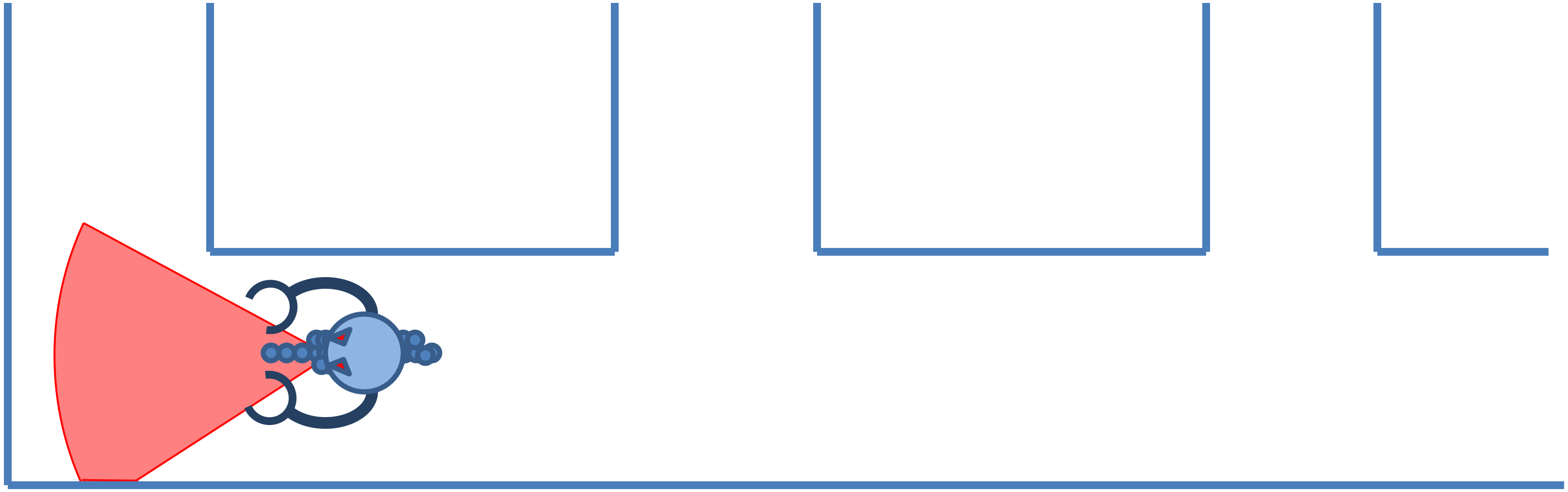
Eg. Kidnapped Robot Problem

- Robot moved to an unknown location
- Estimate we could be anywhere – subsequent observations reduce likely candidate positions



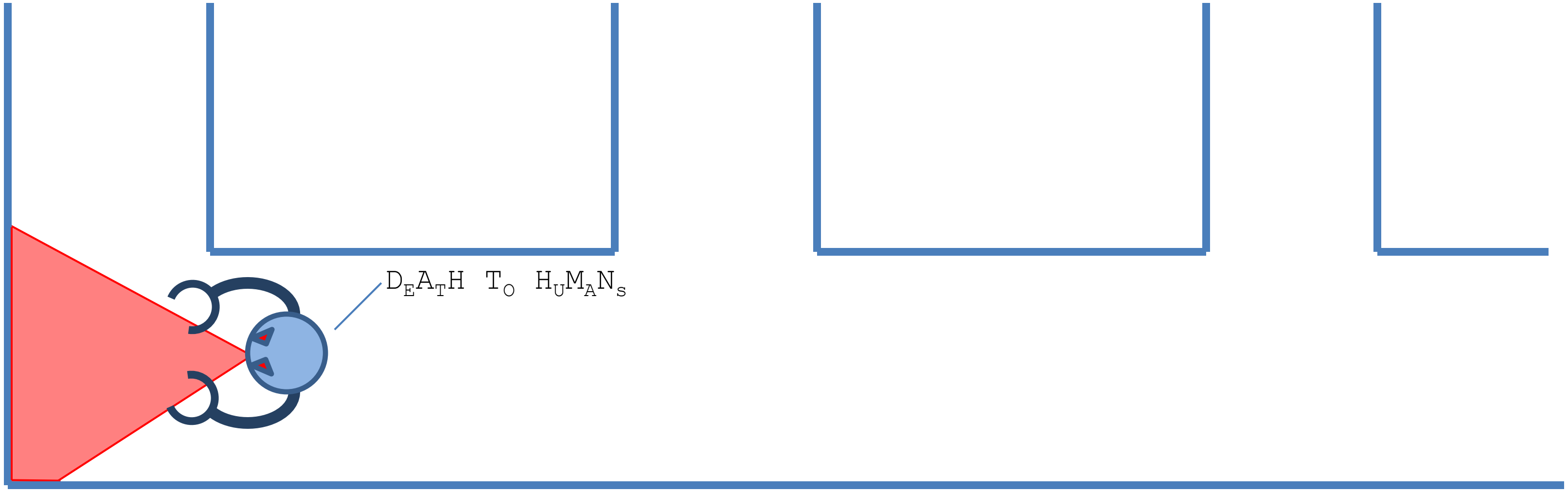
Eg. Kidnapped Robot Problem

- Robot moved to an unknown location
- Estimate we could be anywhere – subsequent observations reduce likely candidate positions



Eg. Kidnapped Robot Problem

- Robot moved to an unknown location
- Estimate we could be anywhere – subsequent observations reduce likely candidate positions



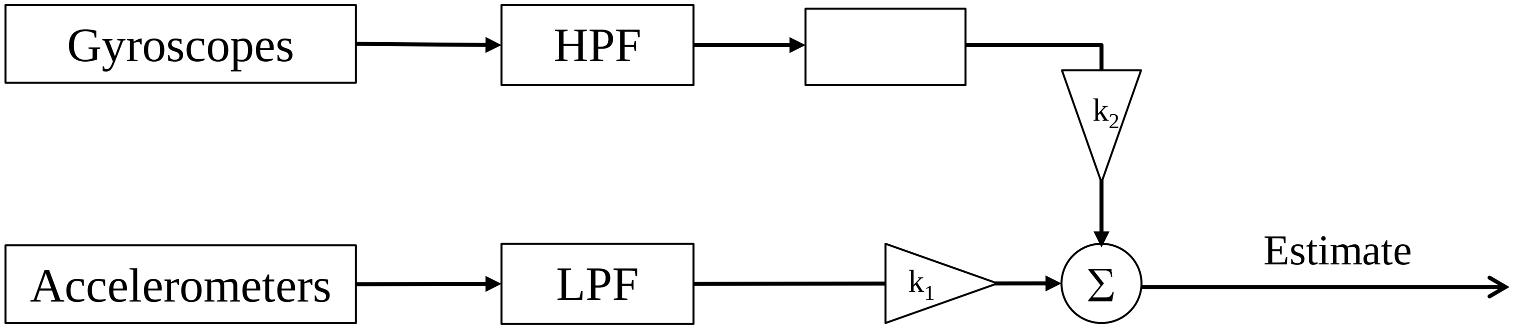
Complementary filter

- Exploits heterogeneous sensor performance to overcome individual shortcomings

- Motivating application: MEMS IMU
  - Accelerometers: unbiased but very noisy
  - Gyroscopes: only kinda noisy but biased

Why not just use the accelerometers to correct the low-frequency bias of the gyros?

Complementary filter



- Exploit signal bandwidth properties:
  - – Low pass filter accelerometer angle estimates
  - – High pass filter gyros and integrate
  - – Output is a weighted mix of estimates

Gratuitous name drop: ANU’s Prof. Rob Mahony, Paul’s PhD supervisor, wrote the complementary filter commonly used in UAV avionics stacks

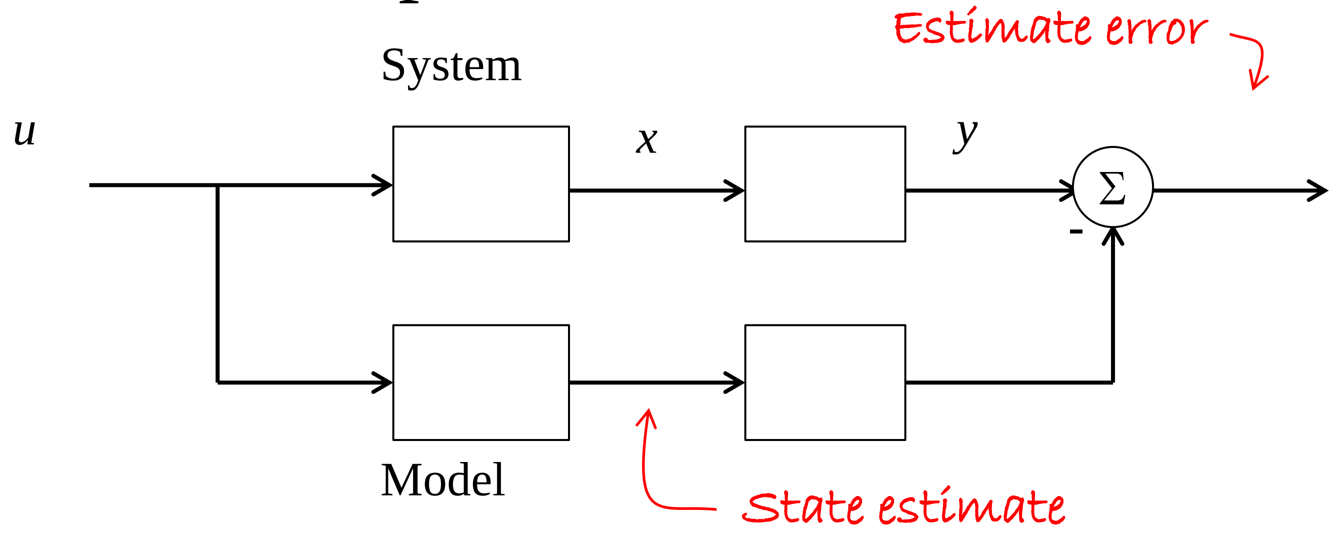
Linear observers

Quick blast from the past semester:
METR4202 observers

Note: if you haven’t done METR4202, don’t worry –
this won’t be on the exam… and also there *is* no exam.

Linear observers

- Observers (aka “estimators”) are used to infer the hidden states of a system from measured outputs.



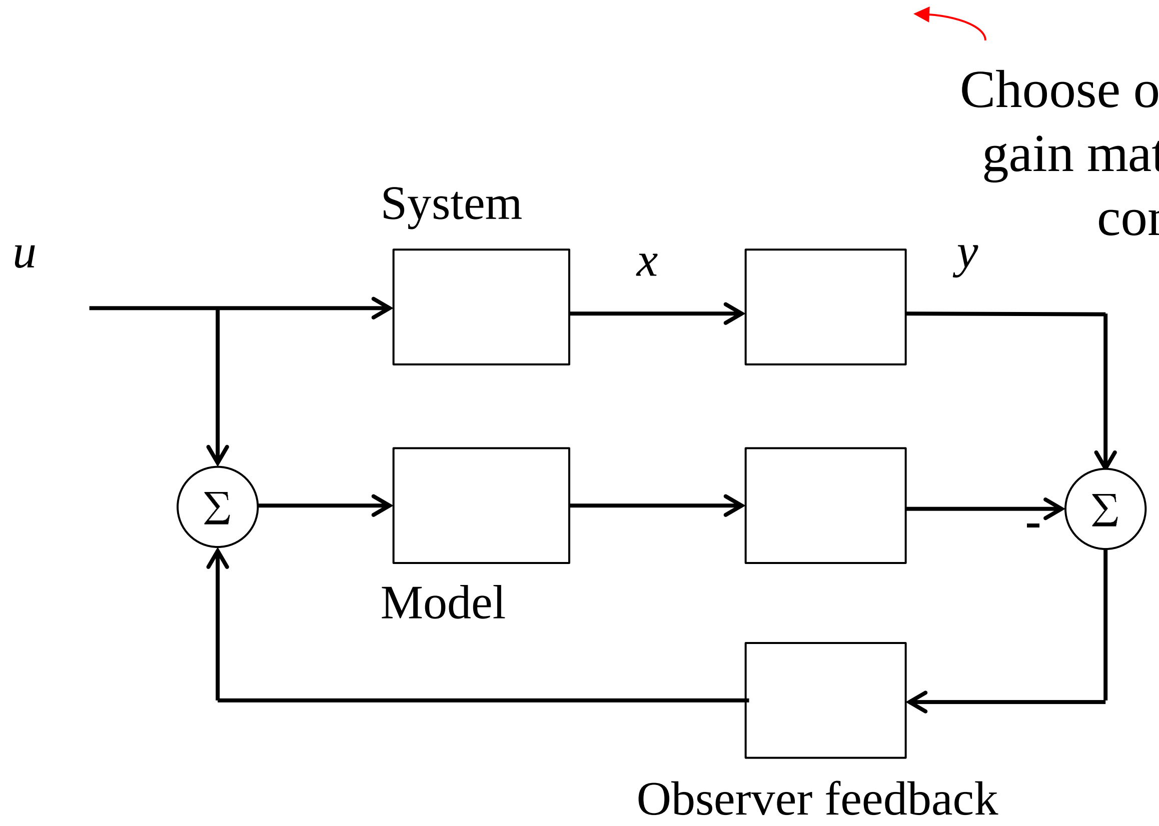
A controller is designed using estimates in lieu of full measurements

Linear observers

- The state estimate can be treated like a control system itself
  - Dynamics to update the estimate:
  - Using an ‘error signal’, , the difference between the real output measurement and the output estimate – the state estimate can be driven by a feedback term.

Linear observers

- Just like you might expect:



Choose observer feedback gain matrix **L** to make converge to 0

Cross-contextualisation

- State-space observers are a sensor-fusion method that infers states from signals
  - But if observers are control functions…
  - And observers are filtering functions…

Profound realisation:

- Fundamentally, filtering is really control and control is really filtering!
  - Oh boy!

Some practical advice

- Some things engineers *never* try to build if they can buy, copy or otherwise avoid it:
  - Power supplies
  - Motor drivers
  - Analog amplifiers
  - Inertial Measurement Units
  - **Sensor fusion and estimation algorithms**

There are many good pre-canned S-F algorithms out there – try using them before writing your own!

Some practical advice

When combining sensors:

- Align measurements spatially and temporally
  - Calibrated kinematic transformation matrix?
  - Time-stamps, common interrupts lines?
- Use sensors to correct other sensors
  - Compensate motion of camera with IMU?
  - Augment dead-reckoning with optical flow?
- Reduce inter-sensor vibration/flex – rigidity!

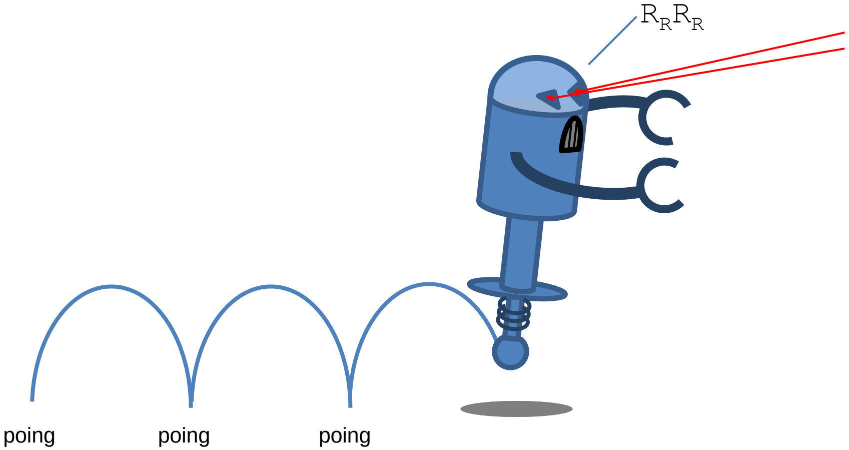
Some practical advice

- Directly sense the state of interest, if possible
  - Avoid numerical integration or differentiation

- Hardware filters use fewer processor cycles
- Software filters take up less board (usually)

- Cheap sensor is cheap; better sensor is better
  - Fused cheap sensors might be *almost* as good
  - The easy solution is often a better sensors

Questions?



Tune-in next time for…

Your soldering is terrible

or

“How I learned to stop worrying and love flux”

Fun fact: One of the first practical applications of the Kalman filter was attitude estimation of the Apollo spacecraft.