HCI and Control theory

Computational Interaction Summer School, Zürich 2017

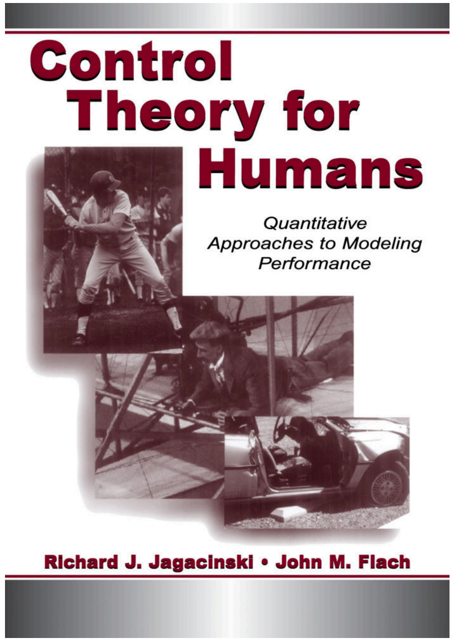
Roderick Murray-Smith

School of Computing Science,
University of Glasgow
Roderick.Murray-Smith@glasgow.ac.uk
Http://www.dcs.gla.ac.uk/~rod

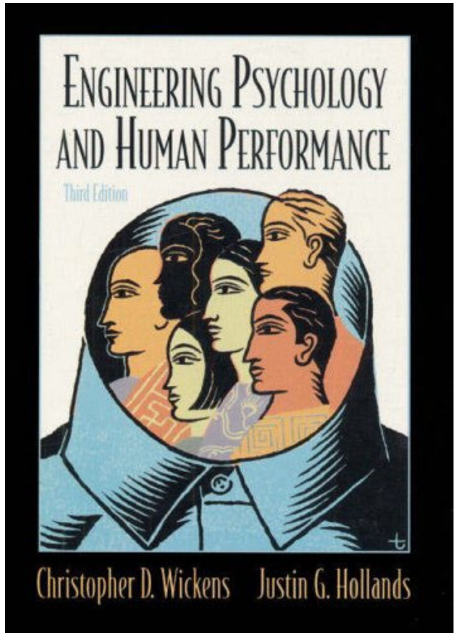
Outline

9:00-10:00	Lecture 1 – Intro to Control and HCI
10:00-10:30	Practical 1 – Dynamics of interface objects
	Coffee break
10:30-11	Lecture 2 – Examples of control in HCI
11:00-12:00	Deep networks and representing hands poses (Daniel Buschek)

Recommended text: Jagacinski & Flach, *Control Theory for Humans*, 2003. (Many figures in this talk taken from this textbook)



Or Ch.10 of Wickens & Hollands



WHAT SHOULD EVERYONE IN HCI KNOW ABOUT CONTROL THEORY?

How will an understanding of control change your approach to HCI?

- Control theory can provide new analytic tools for partitioning and understanding human behaviour, but it can also provide a new way to look at HCI

Jagacinski, R. J. *A qualitative look at feedback control theory as a style of describing behavior.* Human Factors, 19.4 (1977): 331-347.

Background & History

- Control theory applied to technical systems
- Manual Control theory
  - developed from the 40's, peaking in 60's, typically from dynamic systems researchers
  - focusing on the human's ability (typically a well-trained operator) to close the loop as the driver/pilot/gunner
  - Driven by engineering motivations, and need to eliminate error
- Skills development
  - Focused on undisturbed environments, learning and skill acquisition

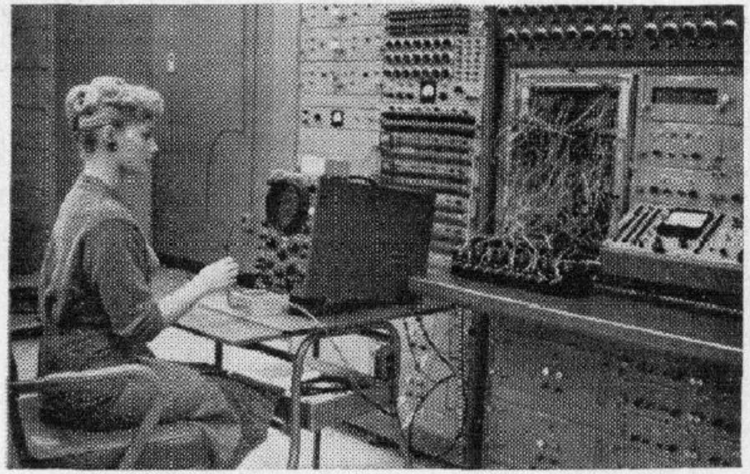


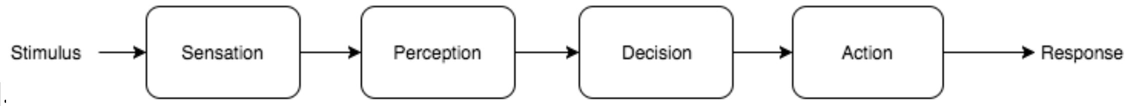
Fig. 11—Experimental arrangement.

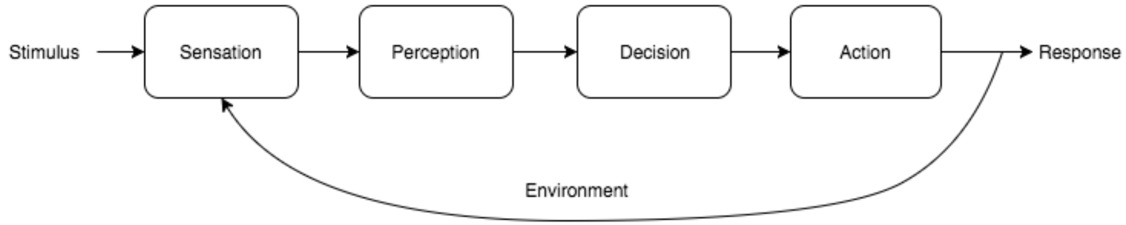
In order to communicate we need to control…

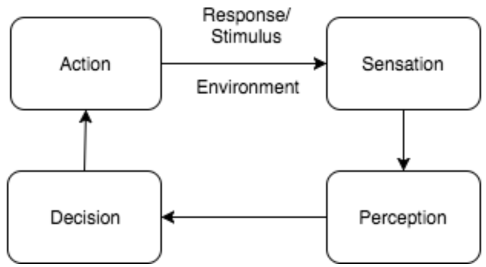
- HCI often presented as communication between human and computer
- In order to communicate we need to control our bodies…
- User's intent intertwines with the properties of the control loops used to generate the information – the physiology of the body and the physical dynamics and transducing properties of the input device, and in computational interaction with the complex interface dynamics.
- Often the purpose of communication with the computer is to control a variable in the real world, e.g. what are we controlling in a music system?

Causality & closed loops

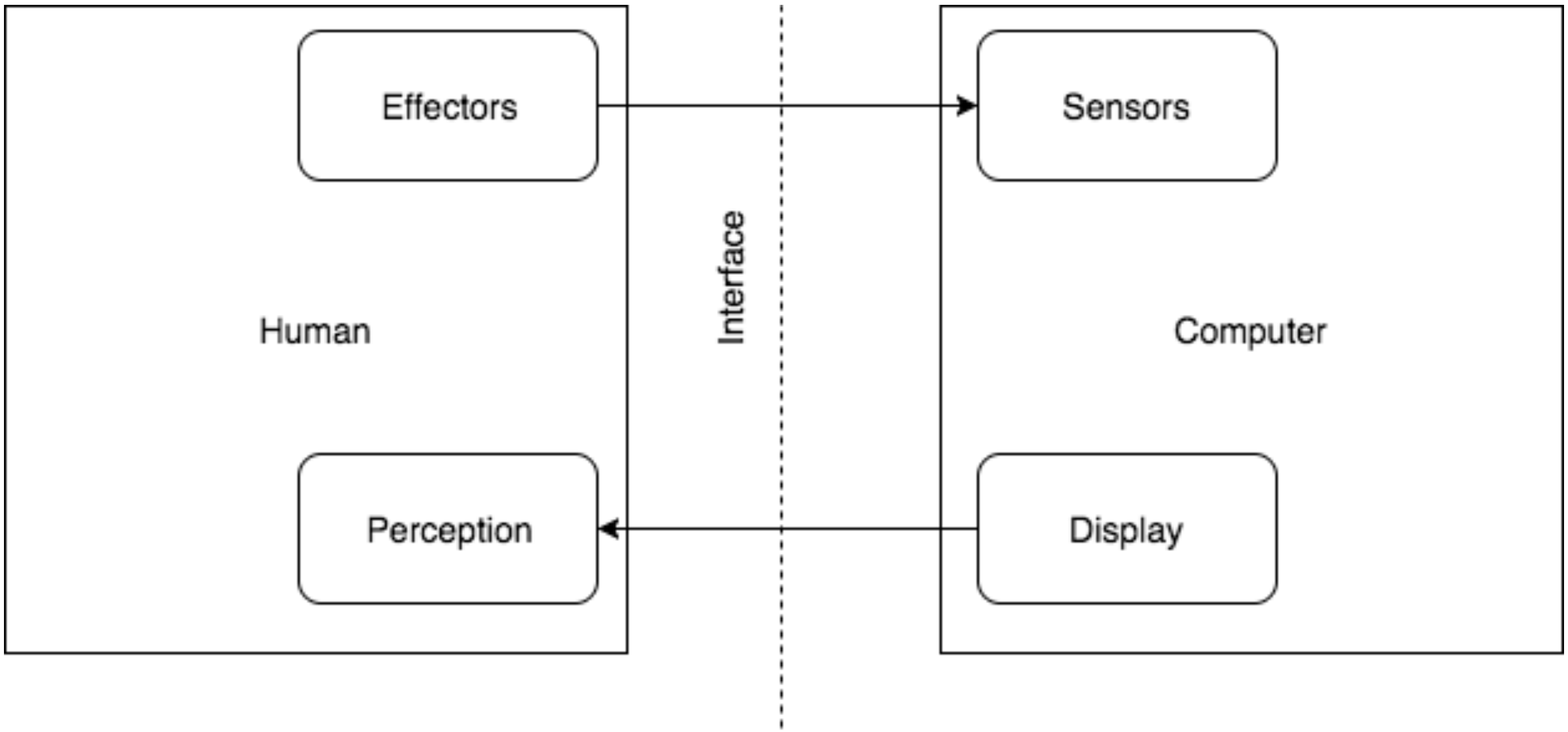
- Can we get away from the notion that behaviour is a simple response to a stimulus?
- How much work in Computational Interaction models a subsystems in closed-loop contexts? Analysing the components out of the closed-loop context gives a false sense of rigour.
- Humans tend to be good at prediction and anticipation and will adapt to the overall loop dynamics (e.g. crossover models in manual control where pilots adapt their behaviour such that the close—loop behaviour stays fairly constant, despite changes in aircraft dynamics).
- This means that many modular representations in control engineering do not translate easily to a human-controlled loop.







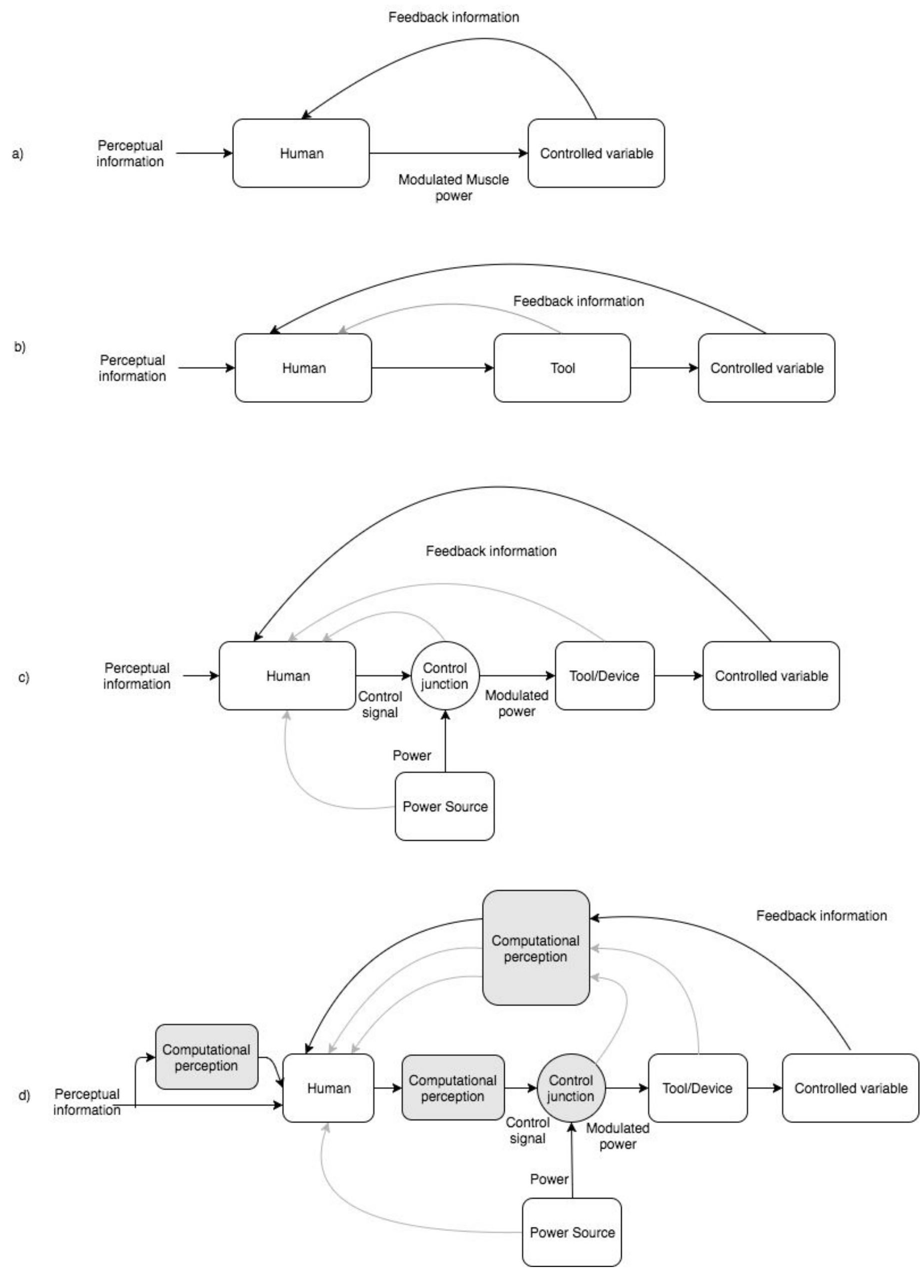
The human-computer interaction/control loop



Evolution of manual control

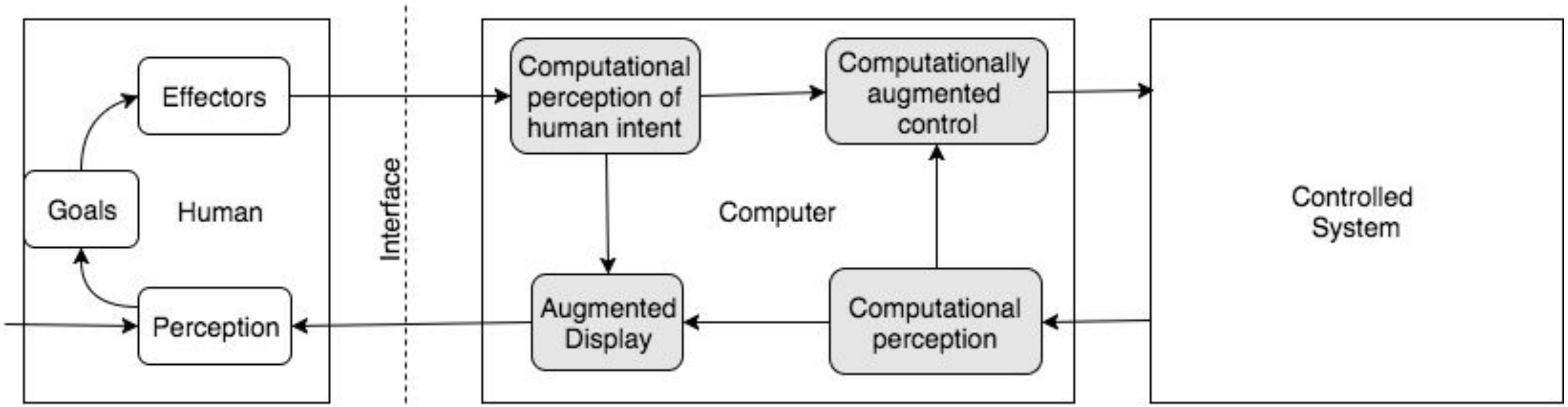
a) Manual engagement with our bare hands
b) Use of tools
c) Use of external power, and then modulate and control ourselves, allows us to control a system with less physical effort.. but still requires skilled muscular control and significant cognitive and perceptual demands.

What next?



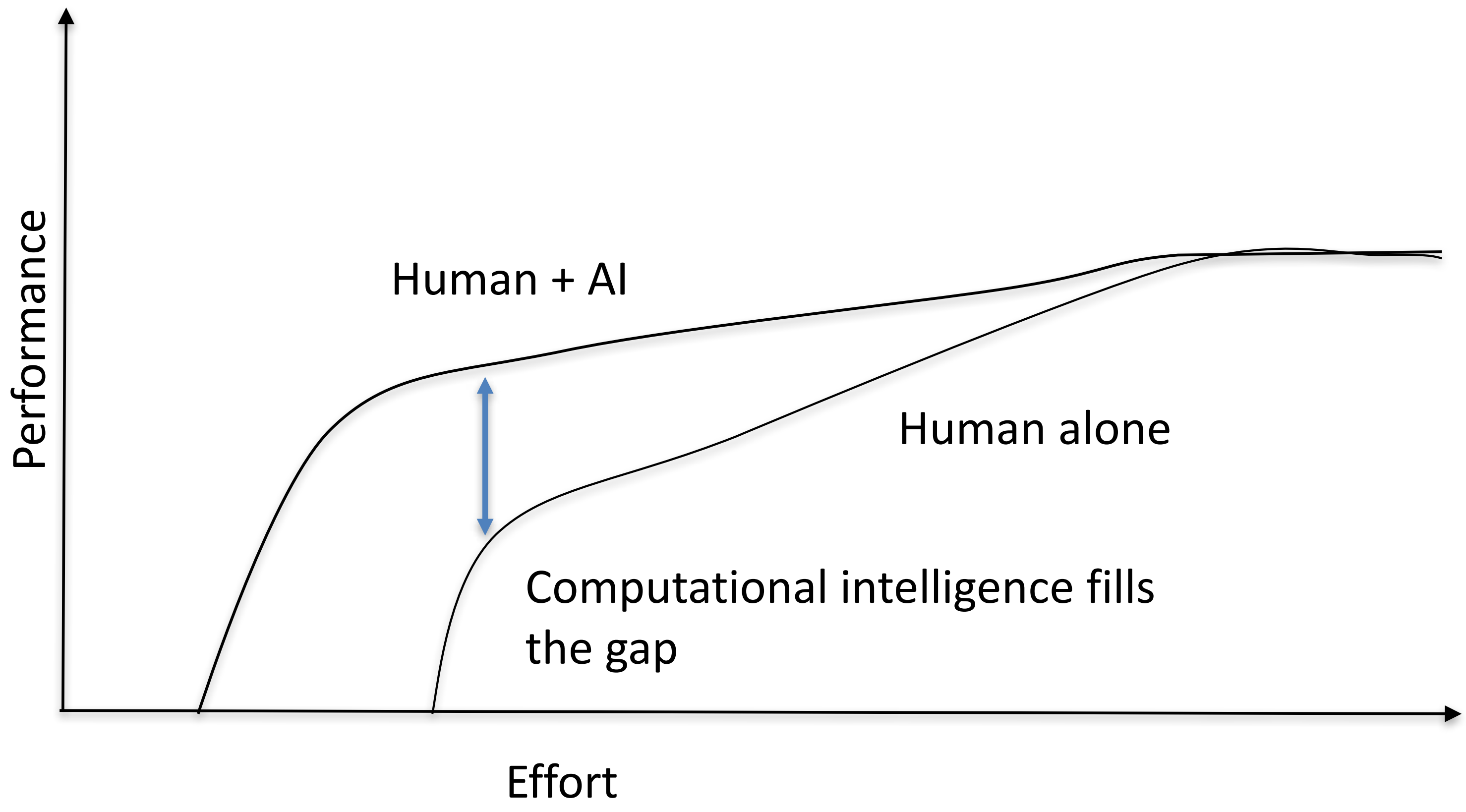
- Btw Do read Charles Kelley's book *Manual Control theory and applications*, *1964*.

Insertion of external *computational* power

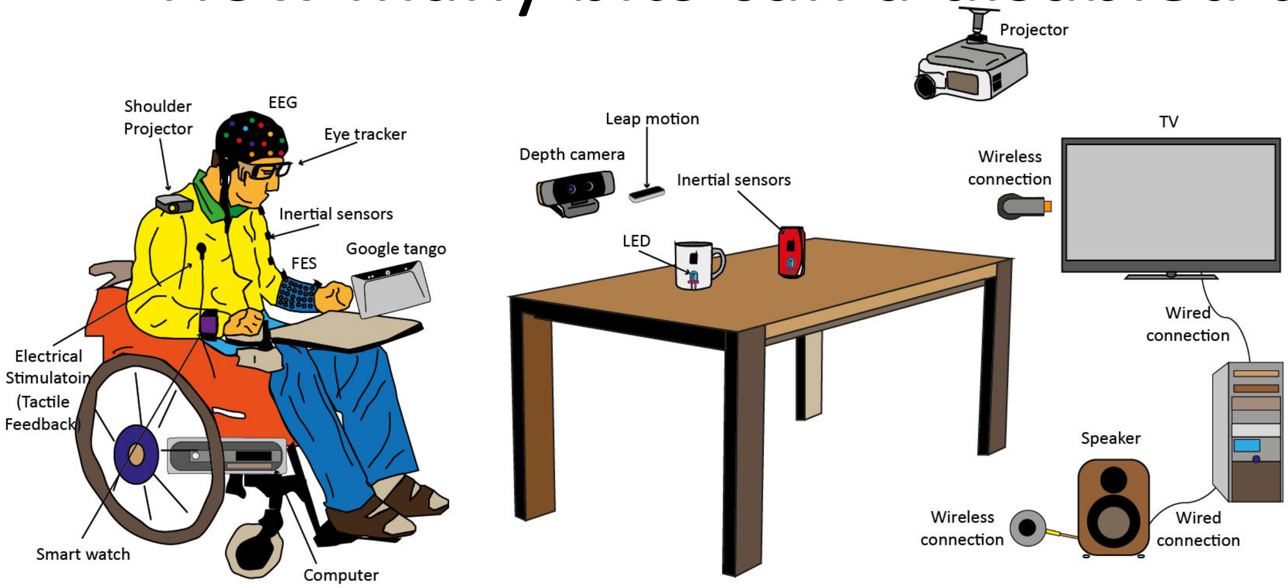


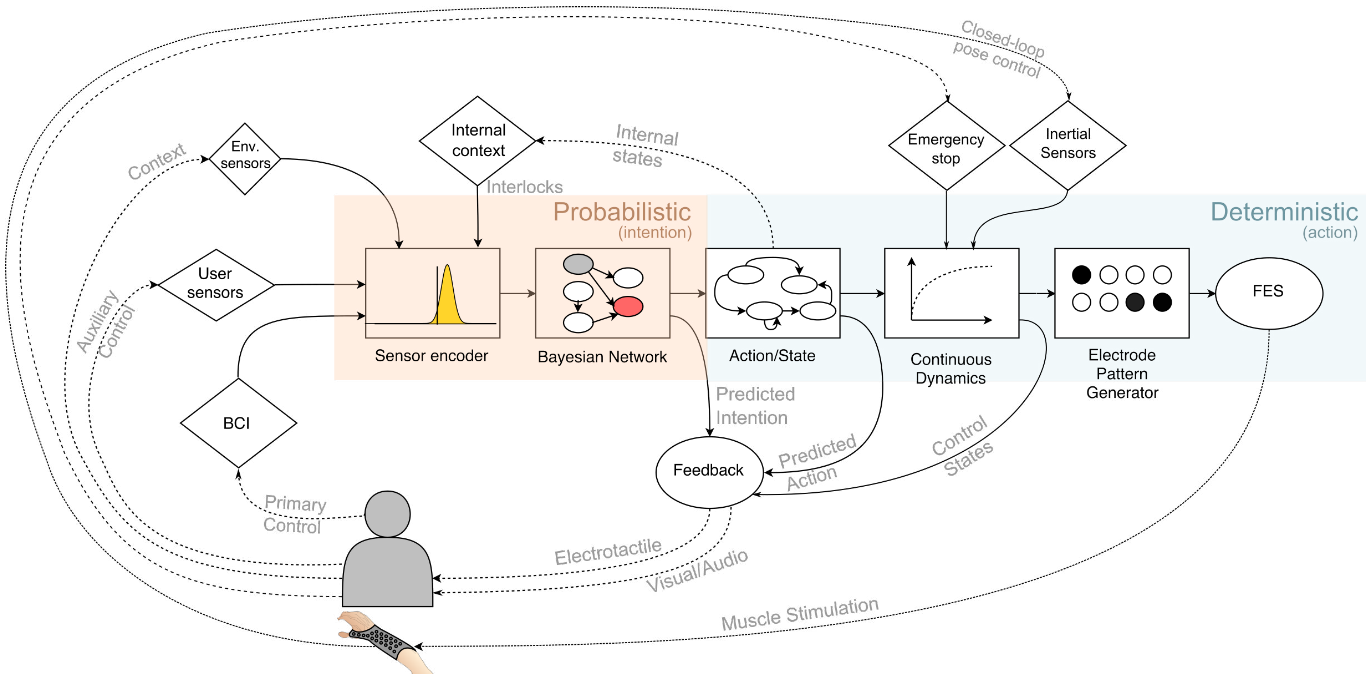
So now instead of power, we add *computational* power
For less *effort* we achieve the same or more *performance*

Performance/Effort/Speed trade-offs



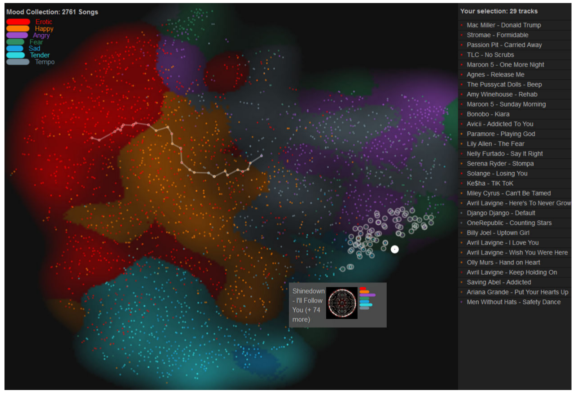
How many bits can a disabled user generate?

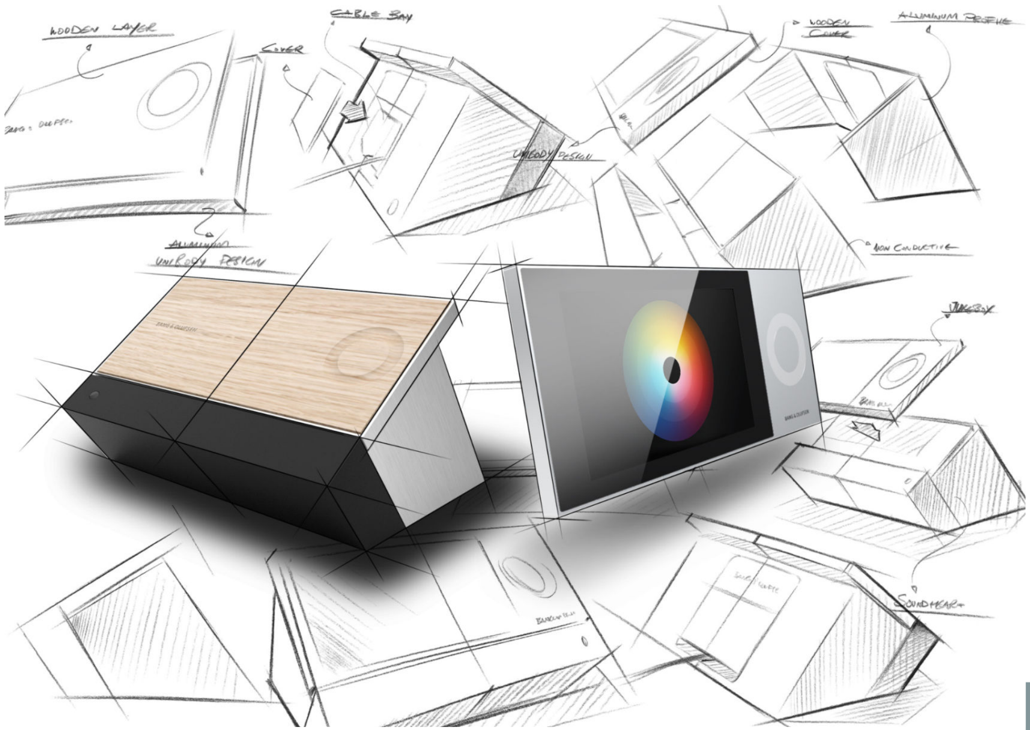




Designing for a range of effort levels

BEO Moment
(launched Jan. 2015)

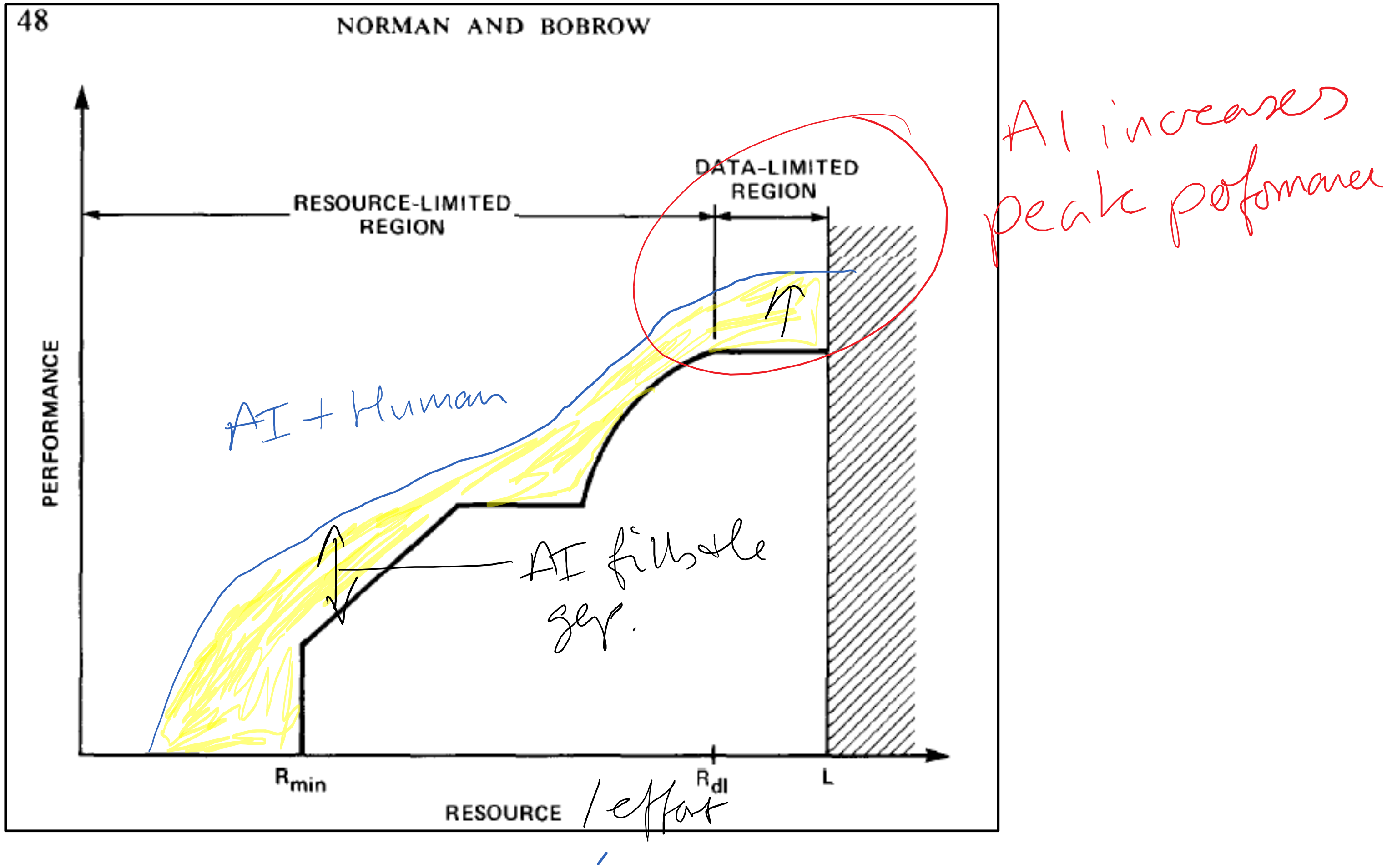




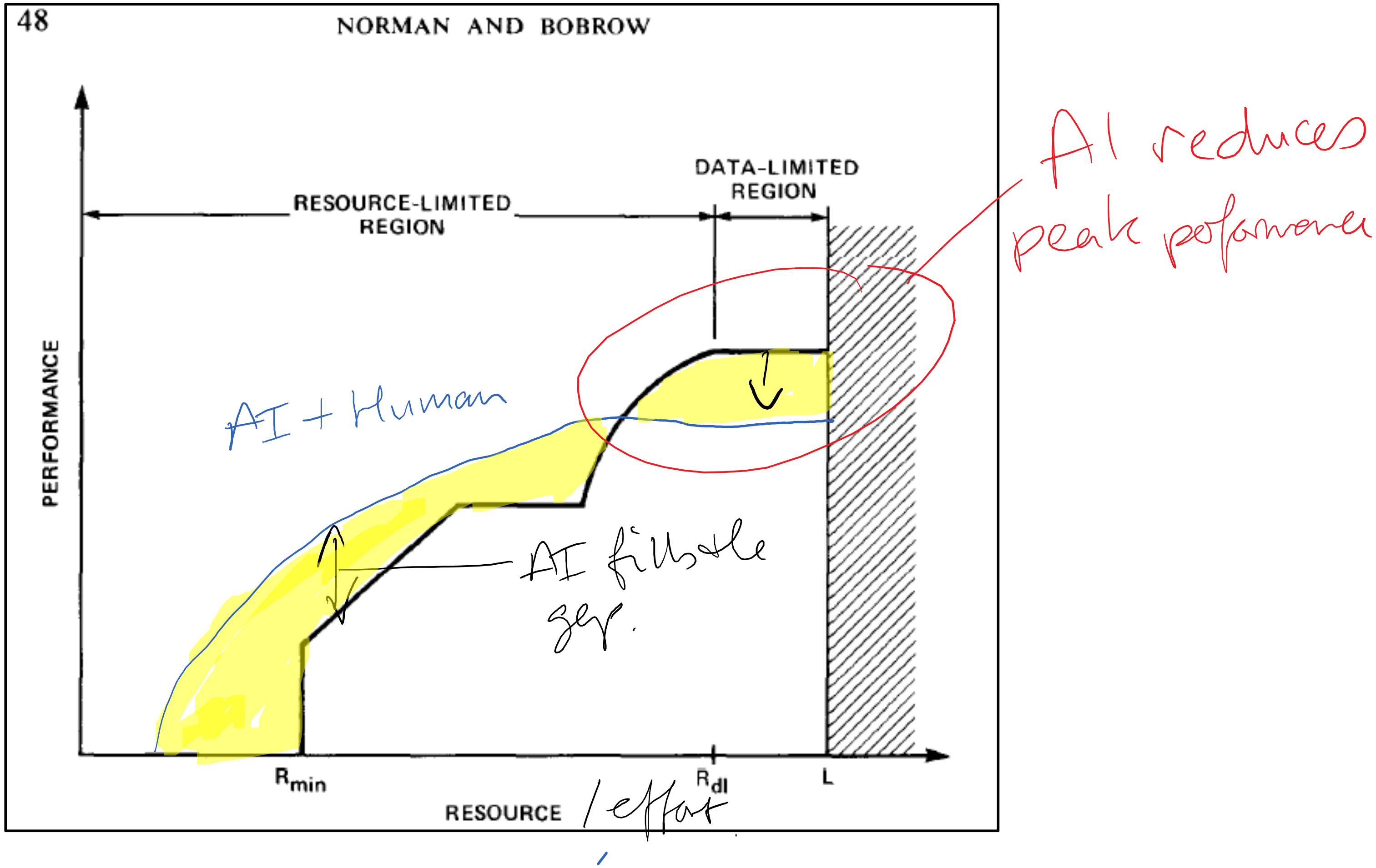
Boland, Daniel (2015) *Engaging with music retrieval.*
PhD thesis, University of Glasgow
http://theses.gla.ac.uk/6727/

BANG & OLUFSEN

Effort – *Performance Resource Functions*



Effort – *Performance Resource Functions*



Classes of continuous control in HCI

- Can you give any examples?

Classes of continuous control in HCI

- Hitting a fixed spatial target with a pointer (e.g. hit an icon)
- Tracking a moving target with a pointer
- Driving a pointer through a spatial continuum of constraints (e.g. steering law)
- Gestures
  - view these as time-based, rather than spatial
- Panning, Scrolling, Zooming dynamics
- Homeostasis- & tracking-based interaction
  - From pointing-without-a-pointer to eye tracking interaction

Going beyond static GUIs

- The modern GUI is primarily spatial in nature, using targeting as the means of communication.
  - This simplifies the problem of providing feedback (simple position displays will do)
  - *However,* loses much of the potential richness of communication.

Trajectory-based interactions

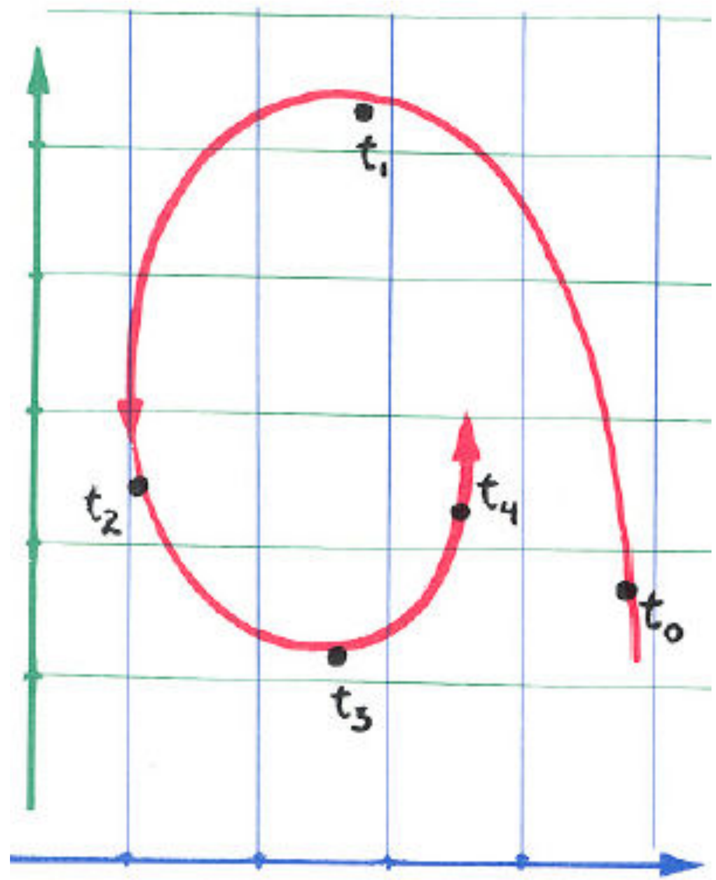
- Navigating through nested-menus, drawing curves, swiping screens and moving devices in the real world, are becoming common tasks in modern computer interfaces.
- Fitts' law, as it has been applied to pointing tasks is not enough to model users' performances in these tasks

Canonical representations of interface dynamics

- potential to give a cleaner underlying structure in design - a canonical representation of the dynamics as a differential equation which is independent of how it was implemented in code.
- makes the comparison of performance between different design decisions easier to document and analyse.

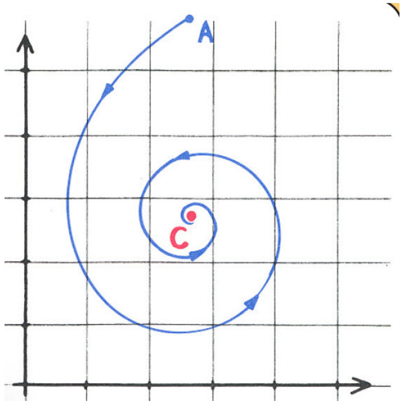
Dynamic systems

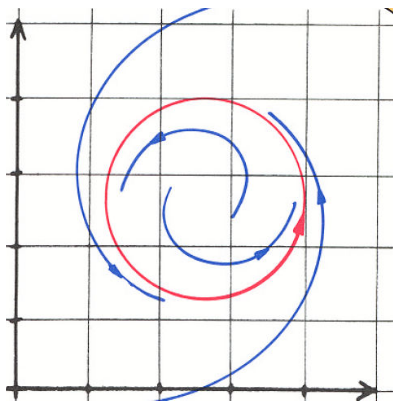
- How a system responds in time.
- Investigation of the behaviour of a controlled system requires us to observe its change of state.
  - In physical systems this requires transfers of energy or mass. An instantaneous change of state in such a system would require an infinitely large flow of energy or mass. In real systems we have a transition which takes place over time, and we call such systems *dynamic systems*.
- Dynamic systems have a *state* represented by a real vector in a *state space*.



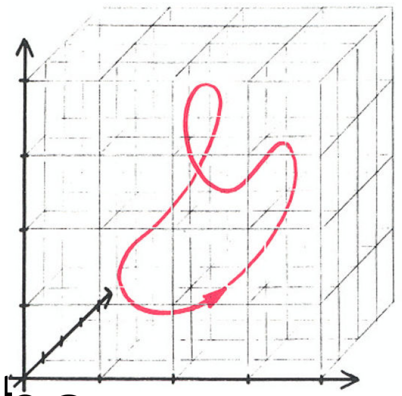
Dynamic systems

- They have three classes of behaviour:
  1. *equilibrium* (state does not change over time),
  2. *transition* (from one state along a trajectory in state space to another condition, either equilibrium or periodic)
  3. *periodic behaviour* -the system passes through the same states at constant intervals of time.

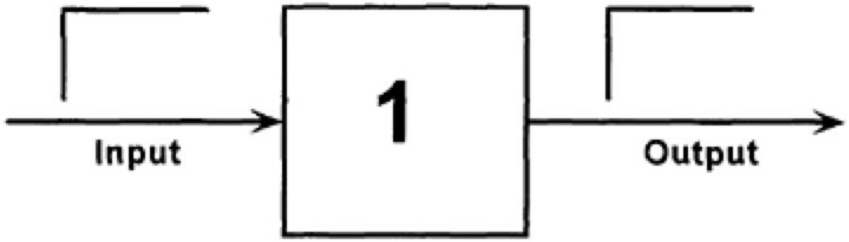


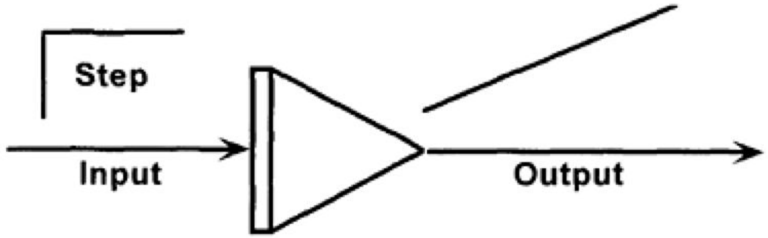


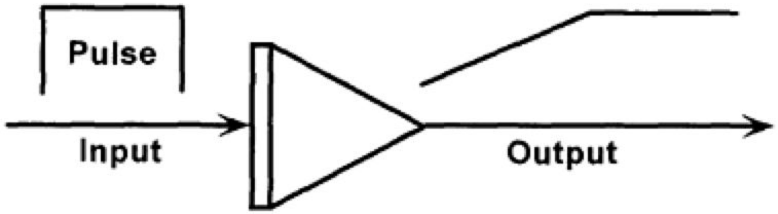
System order

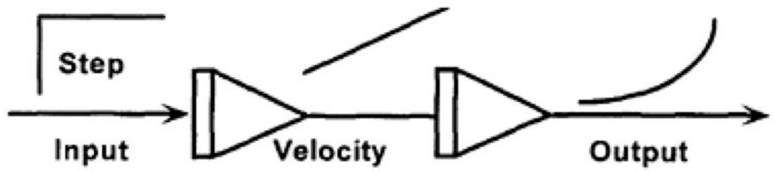


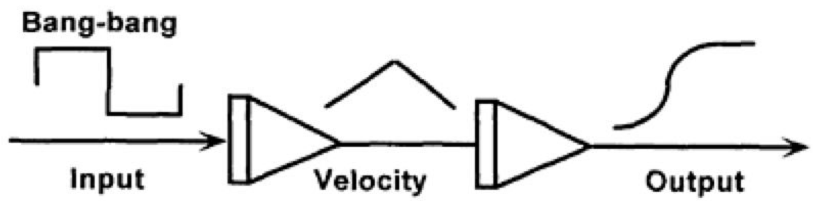
- Can refer to the highest derivative, or the number of coupled 1st order differential equations used in the model
- Control order refers to number of integrations between control input to a plant and output of a plant
- We will think of a dynamic relation between displacement of a control device and behaviour of system being controlled


1
Input
Output


Step
Input
Output

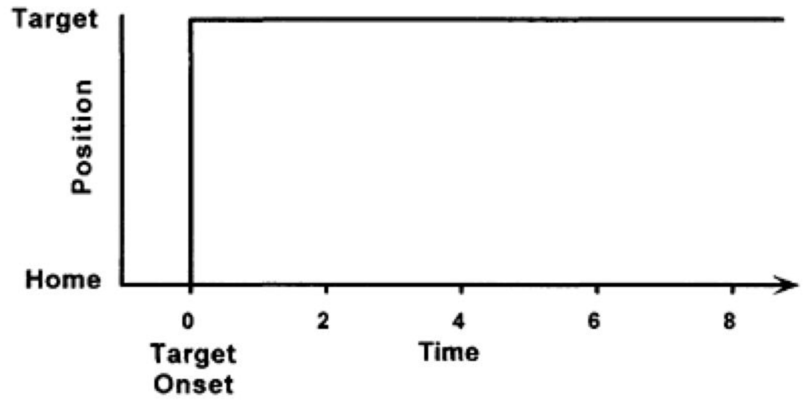

Pulse
Input
Output

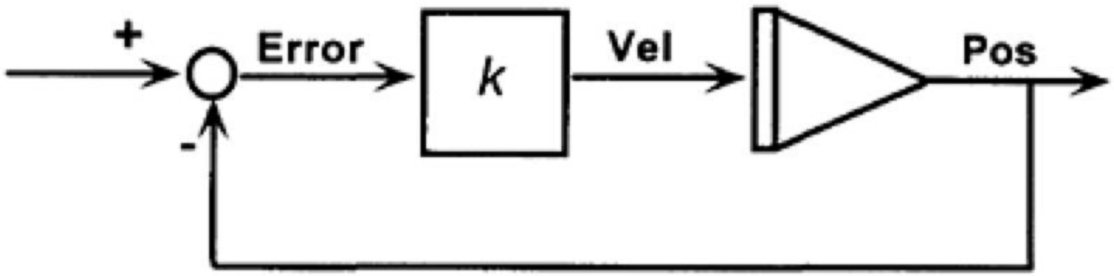

Step
Input
Velocity
Output

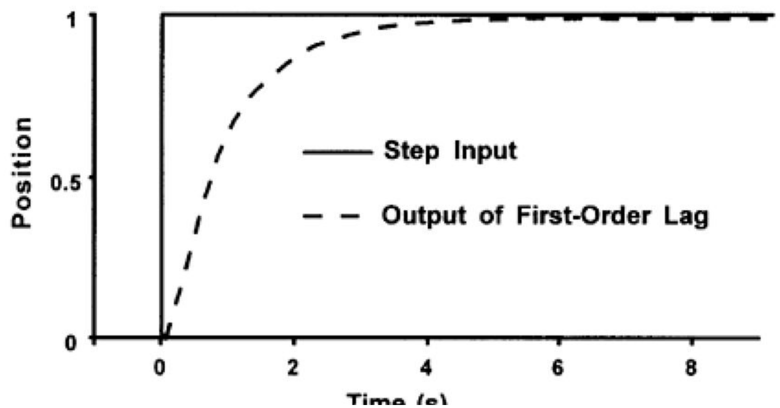

Bang-bang
Input
Velocity
Output

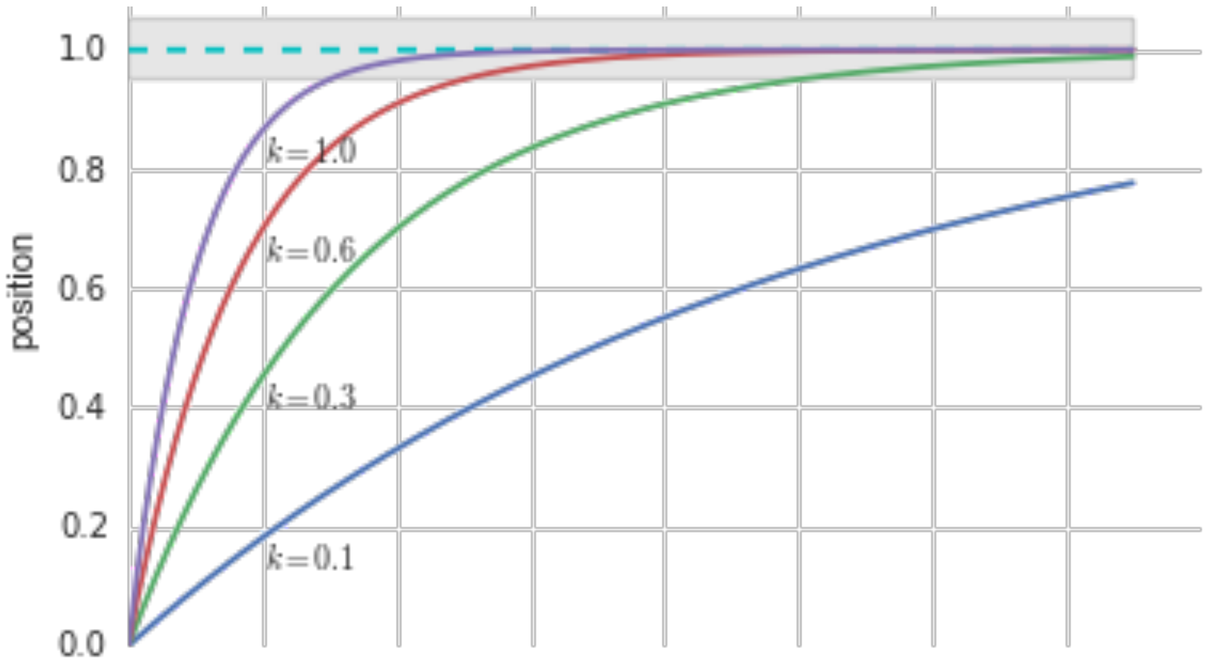
First order responses

- Output is proportional to integral of error

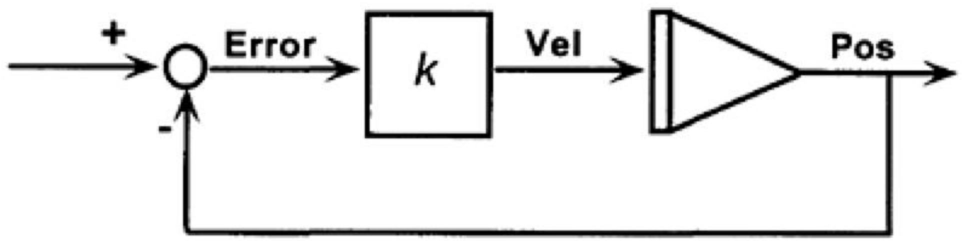








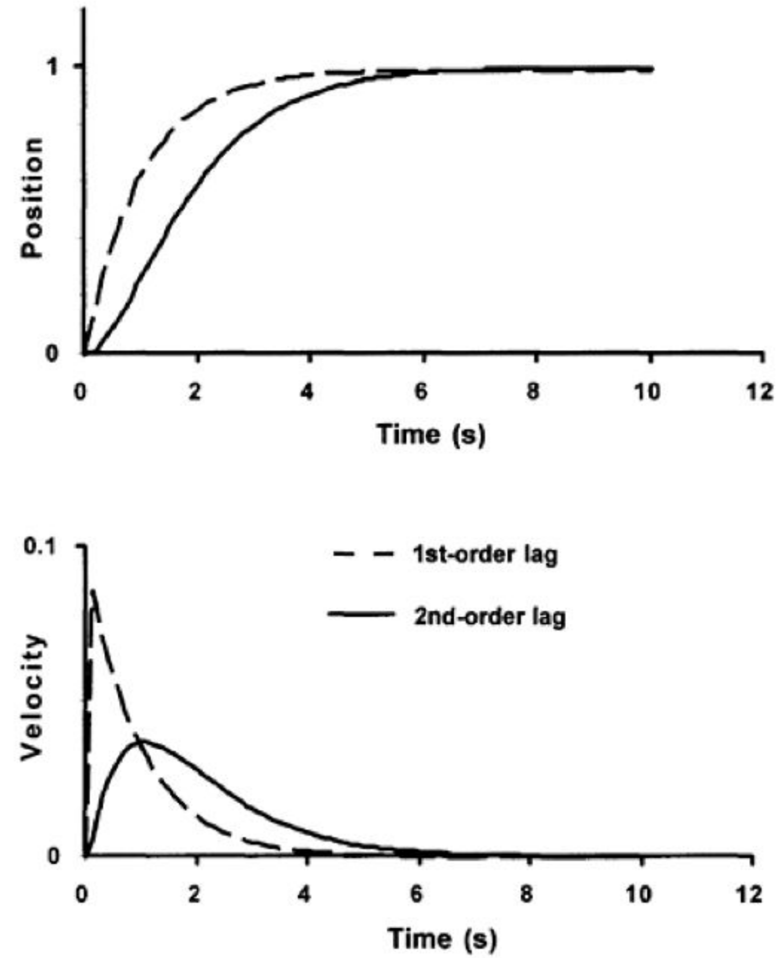
1.0
0.8
0.6
0.4
0.2
0.0
position
$k=1.0$
$k=0.6$
$k=0.3$
$k=0.1$

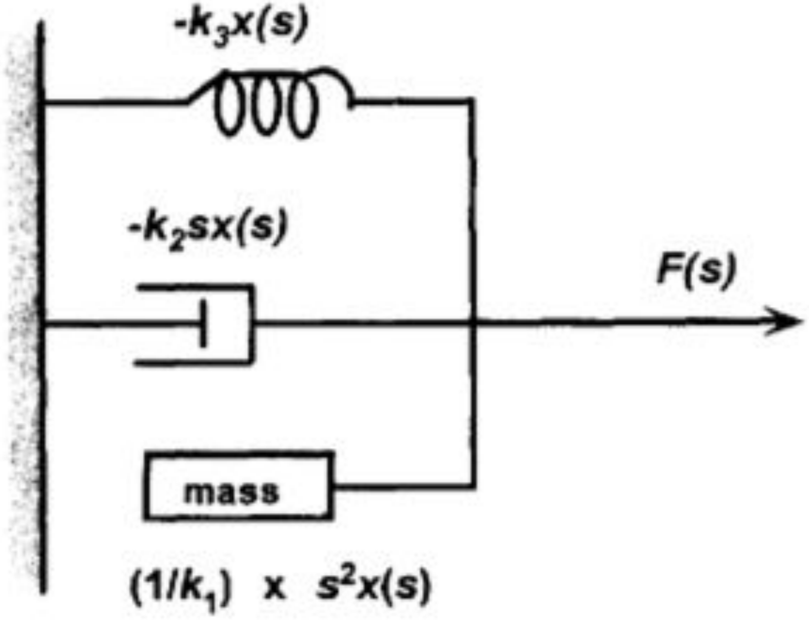


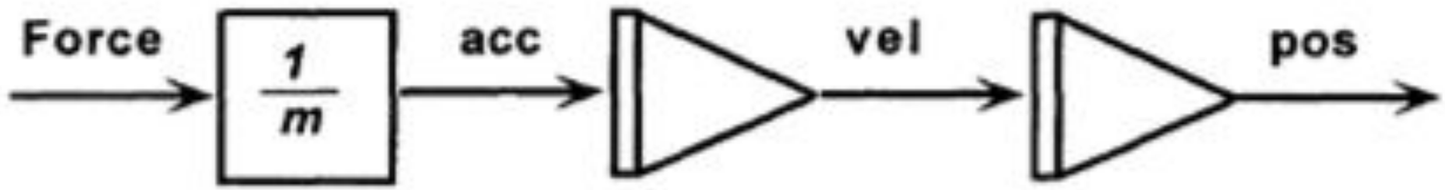
+
-
Error
k
Vel
Pos

2^{nd} order systems

"The linear second-order system may be the most common and most intuitive model of physical systems" Bahill (1981)







- E.g. in pointing, an arm has mass
  - systems with mass cannot instantaneously achieve high velocity.
  - The rate at which velocity builds up depends on the force applied to the limb, and its mass.
  - a = F/m

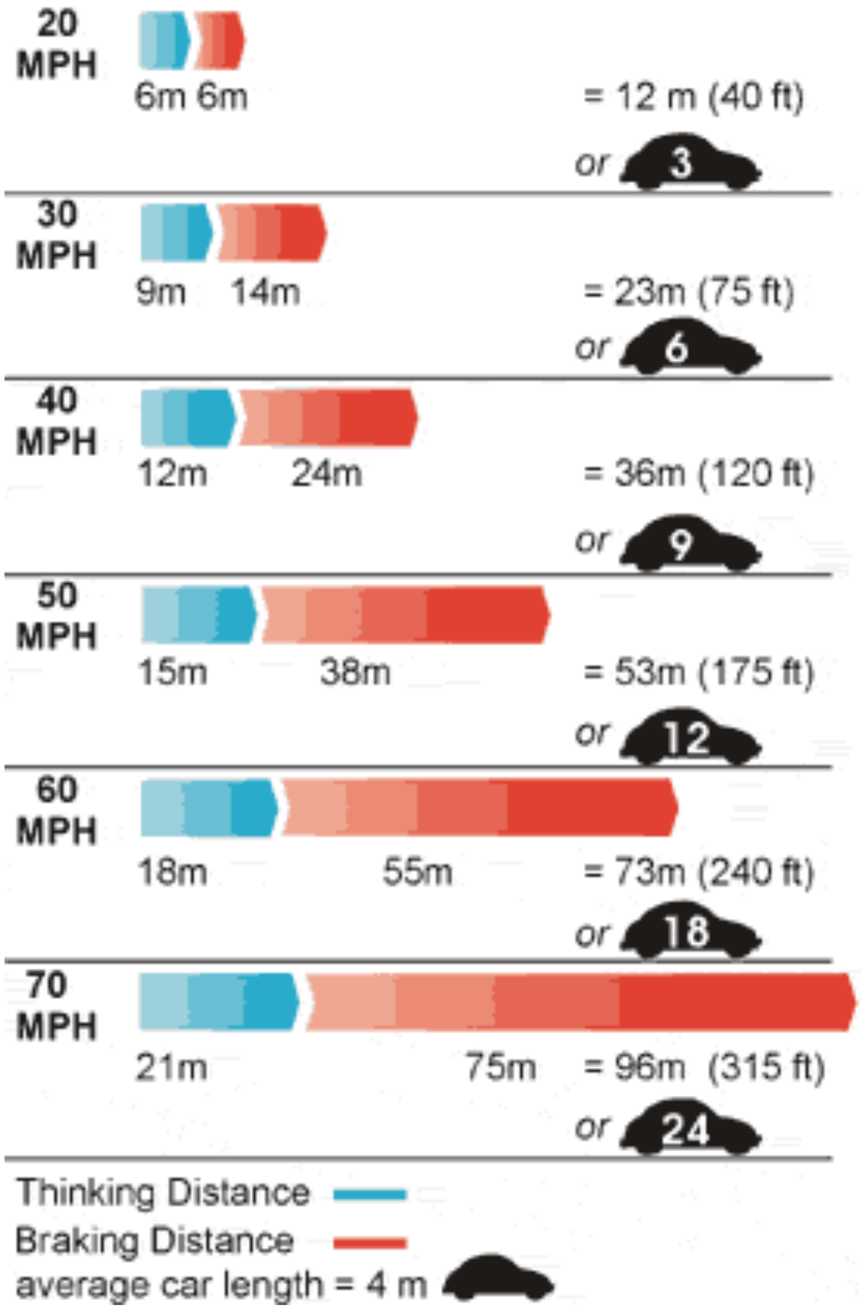
Controlling a car

- At what distance from a junction should a car driver hit the brake?

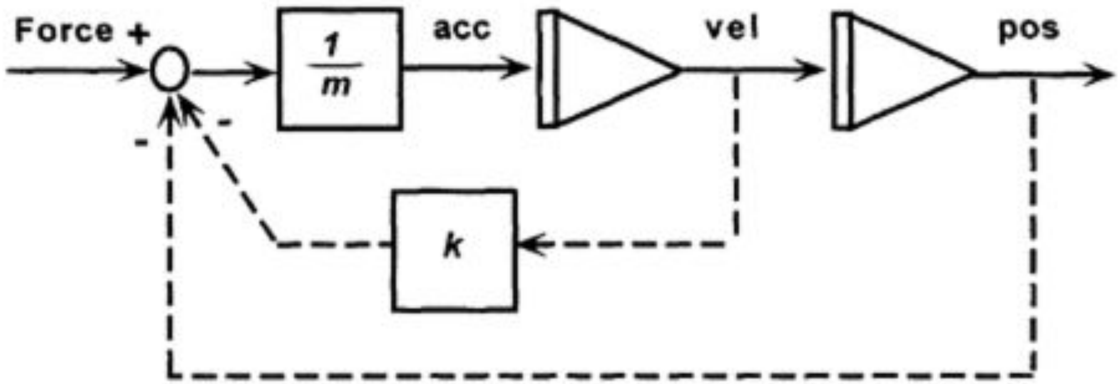
Depends on the speed!

Driver must know the distance (position) and speed (velocity)

Same is true for any system behaving according to 2nd law of motion…

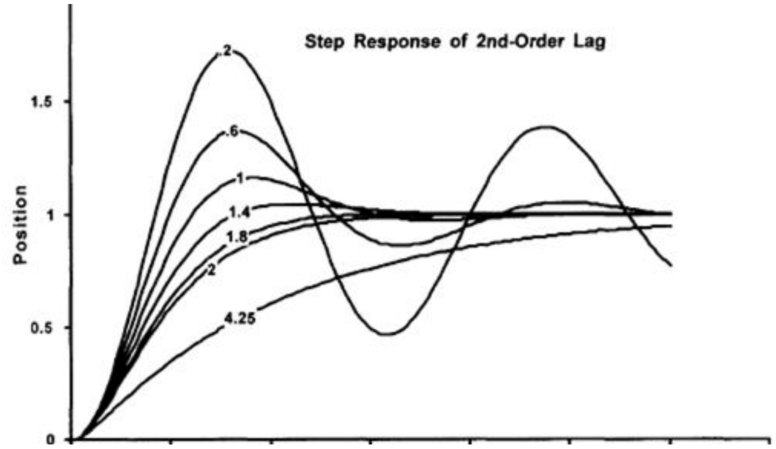


Controlling a 2^{nd} order system

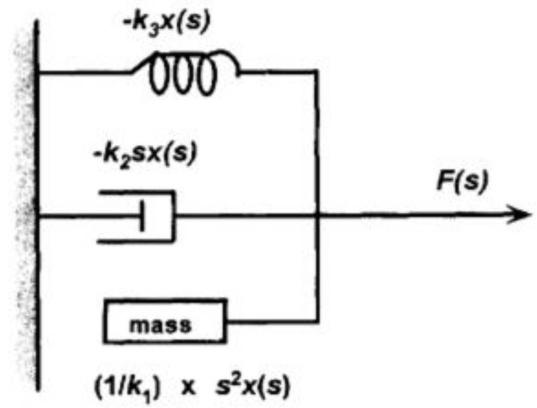


– Here we have a parameter k weighting the velocity feedback relative to position feedback

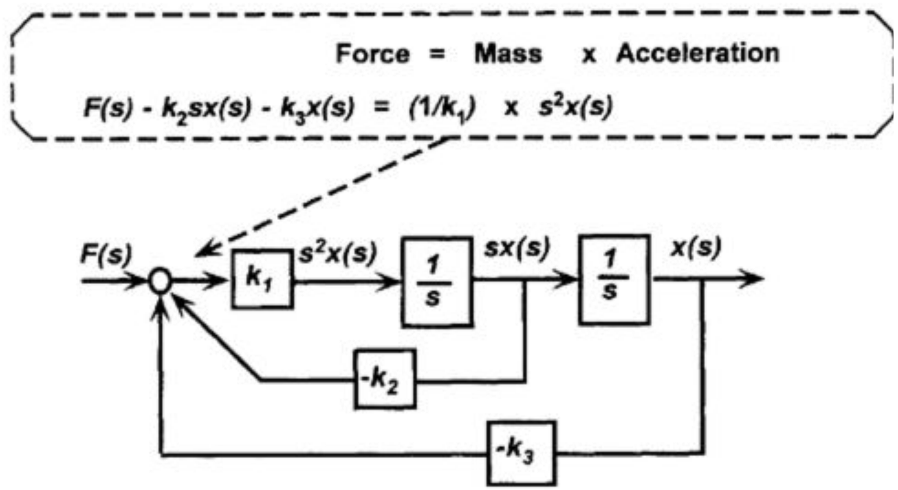
- For small values of k we see large overshoot and oscillations (underdamped)
  - Ignoring velocity leads to overshoot
- For k >2 we have no overshoot (overdamped)
  - Too much velocity weighting leads to sluggish behaviour



- The spring, mass, damper system is often used as a prototype of a 2nd order system



$$F(t) - k_2 \dot{x}(t) - k_3 x(s) = \frac{1}{k_1} \ddot{x}(t)$$

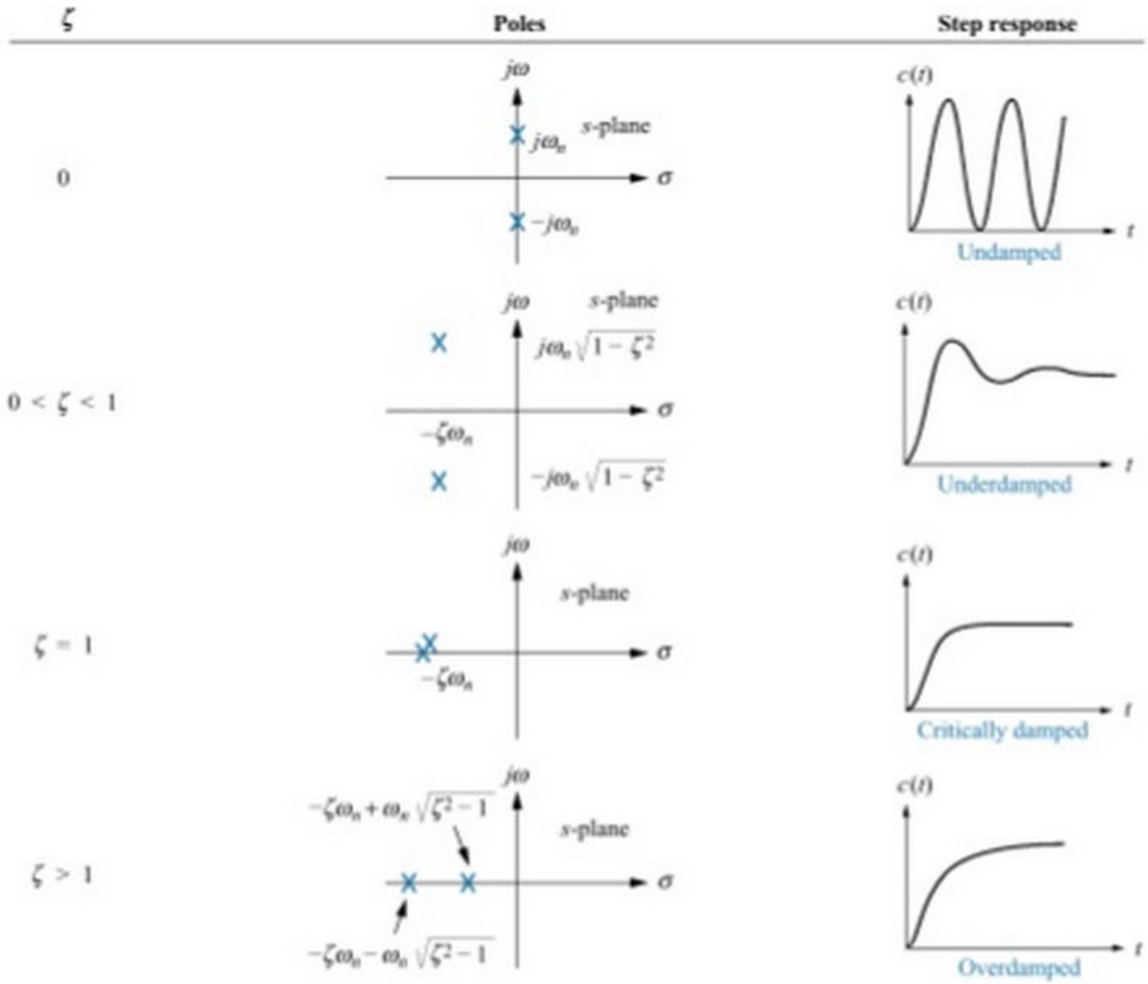


$$F(s) = \int_0^\infty e^{-st} f(t)\, dt.$$

$$\zeta = \frac{k_2}{2}\sqrt{\frac{k_1}{k_3}}$$

$$\omega_n = \sqrt{k_1 k_3}$$

- ζ Is the damping ratio.
  - $\zeta=0$ the system is undamped, and would oscillate continuously
  - $0<\zeta<1$ underdamped
  - $\zeta>1$ overdamped
- ω_n is the undamped natural frequency
  - Frequency at which the system would oscillate if damping were zero.
- $1/(\zeta\,\omega_n)$ is the dominant time constant for the underdamped case



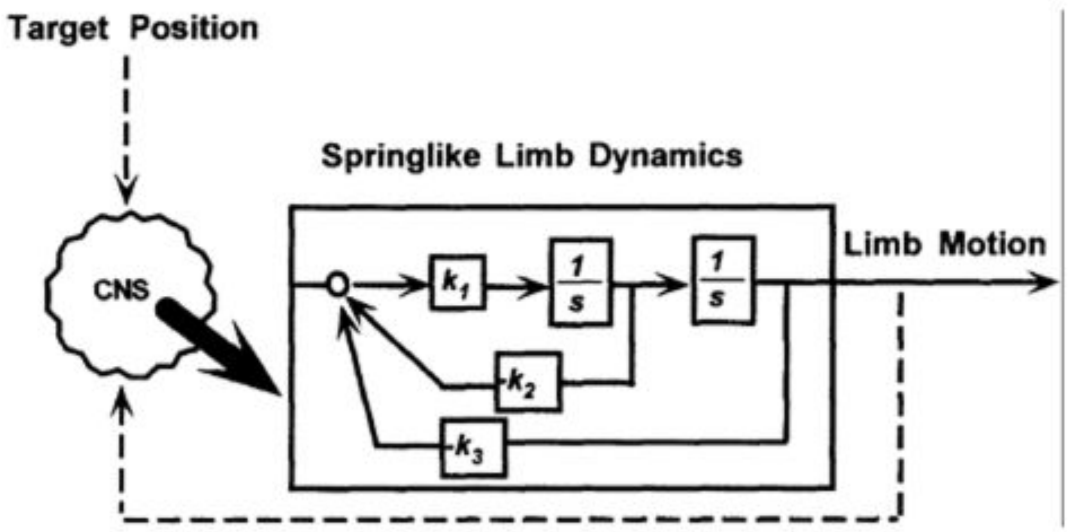
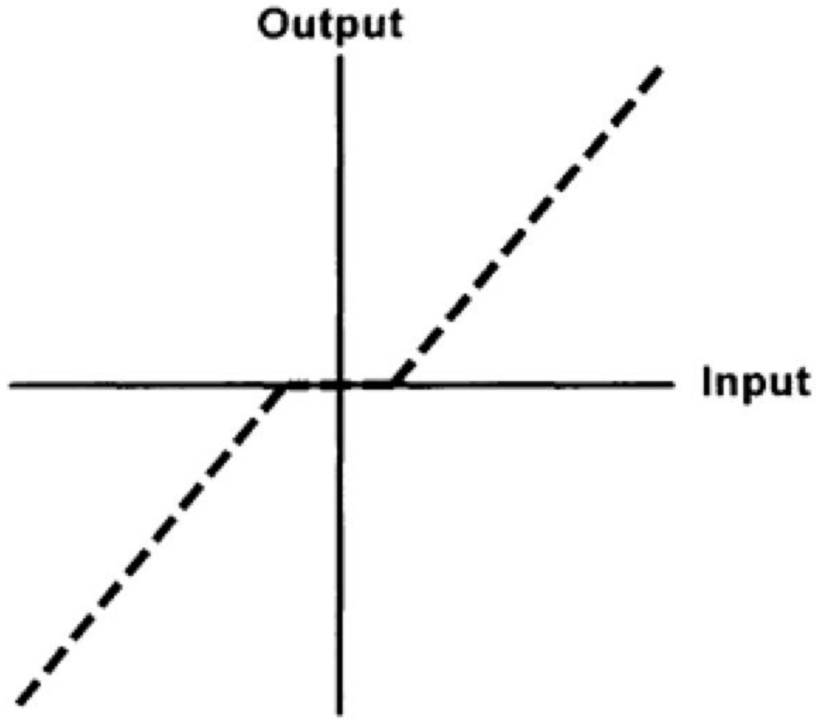


FIG. 6.5. A hierarchical control system. The outer loop perceives the need to move the arm to a new target position (e.g., based on visual feedback). The inner loop represents the mass-springlike dynamics of the limb.

- How is the control distributed between the central nervous system and the dynamics of the limb?

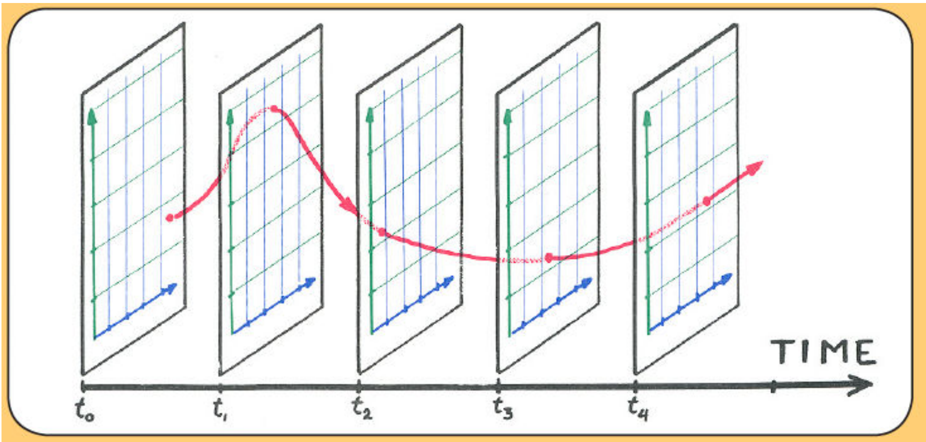
Control devices

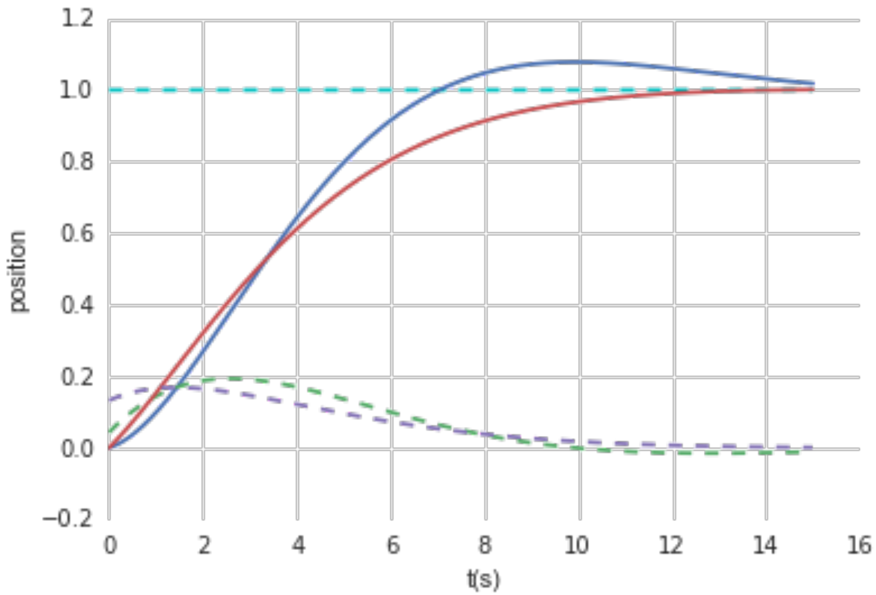
- Importance of null space (deadzone)
- Hysteresis
  - Moving deadband associated with reversal of direction
  - E.g. debouncing keys
- Be careful to compare same dynamics when comparing control device
  - E.g. don't compare position mouse with velocity joystick



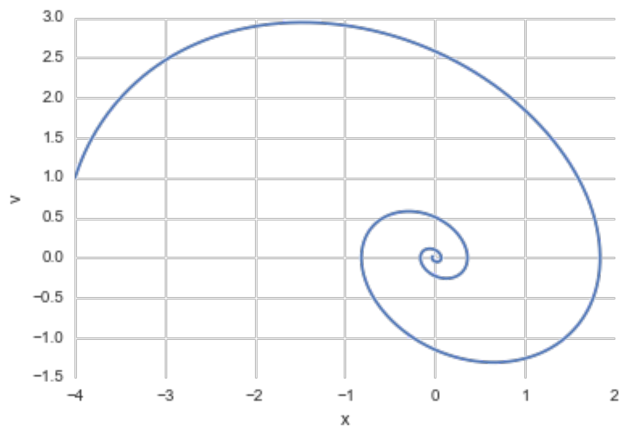
VISUALISING CONTROL SYSTEMS AND RESPONSES

Time-series responses

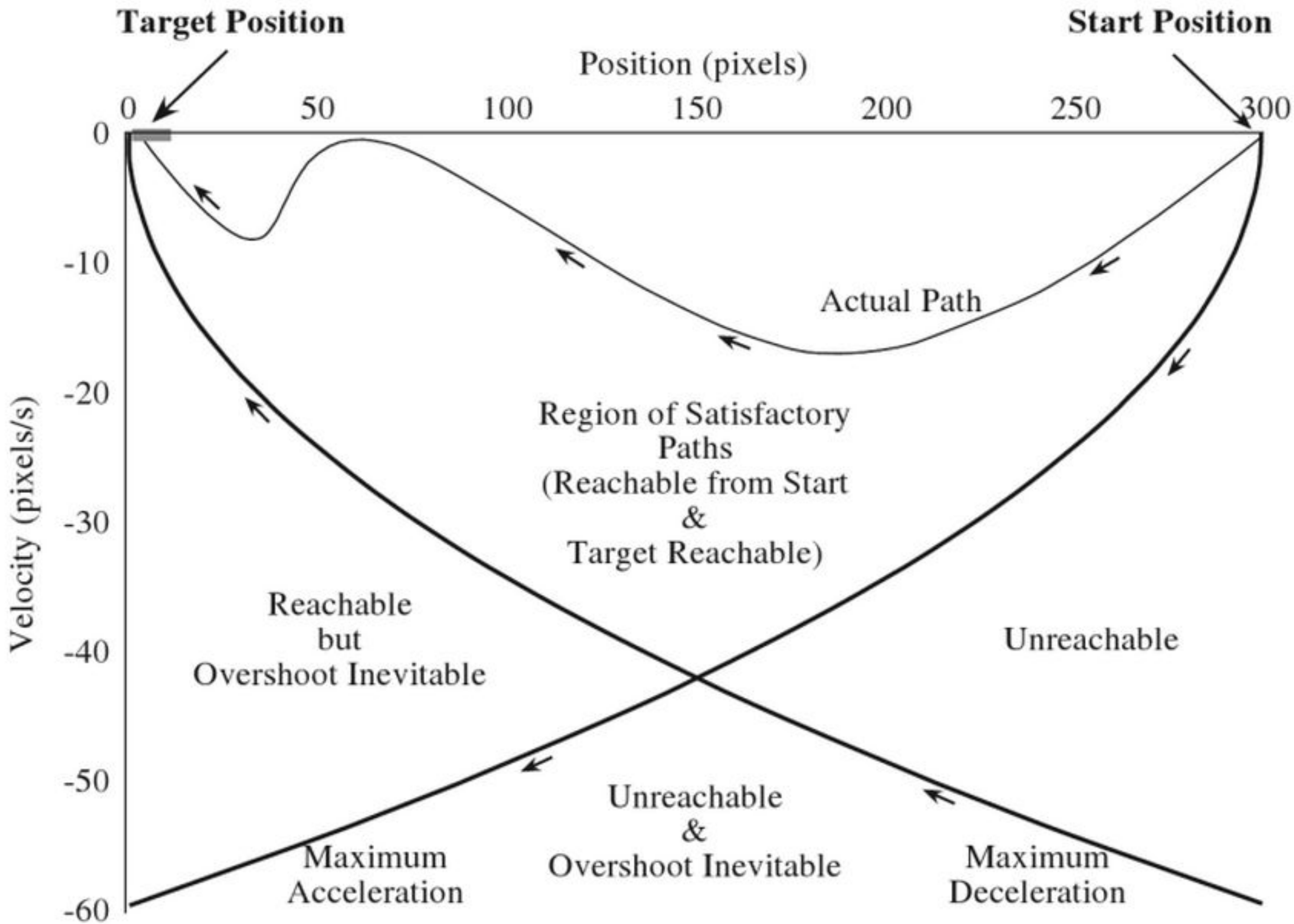




Phase plots

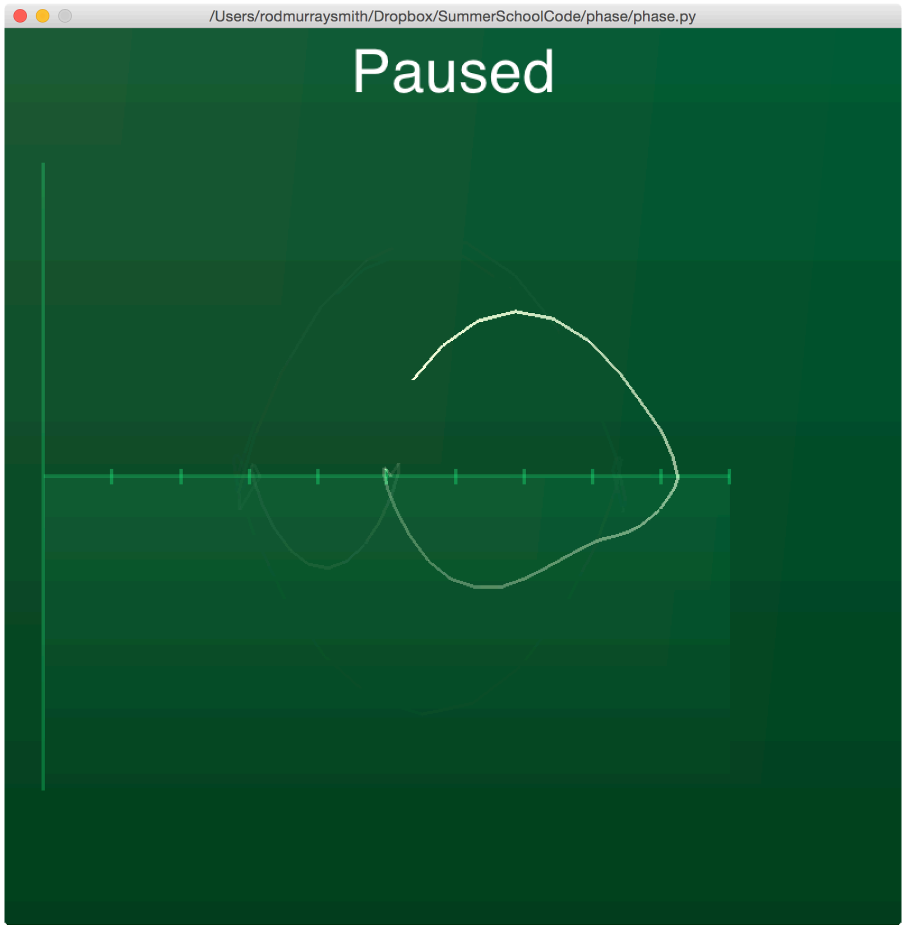


- Phase plots are graphical tools for studying second-order systems.
  - Present motion trajectories in the state-space of a second order system (the phase plane) corresponding to various initial conditions.
  - Provide insight into the qualitative nature of the system, without having to solve the systems analytically.



P48, Fig.3.3 in Bennett & Flach 2011,

Try phase.py



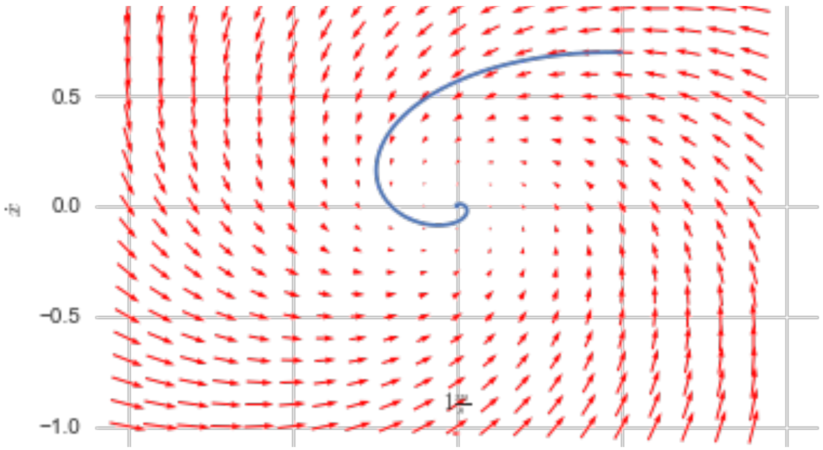
To create a classifier:

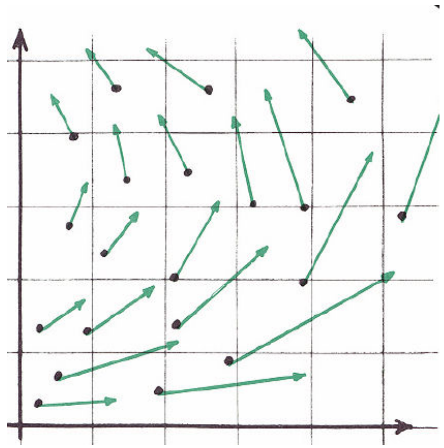
- Right click to begin
- Left click to place first path segment
- Left click again to extend path
- Optionally use <A> and <D> to change width
- Repeat until path complete
- Right click to save
- Press <U> to undo last point while drawing)

To preserve traces on screen:

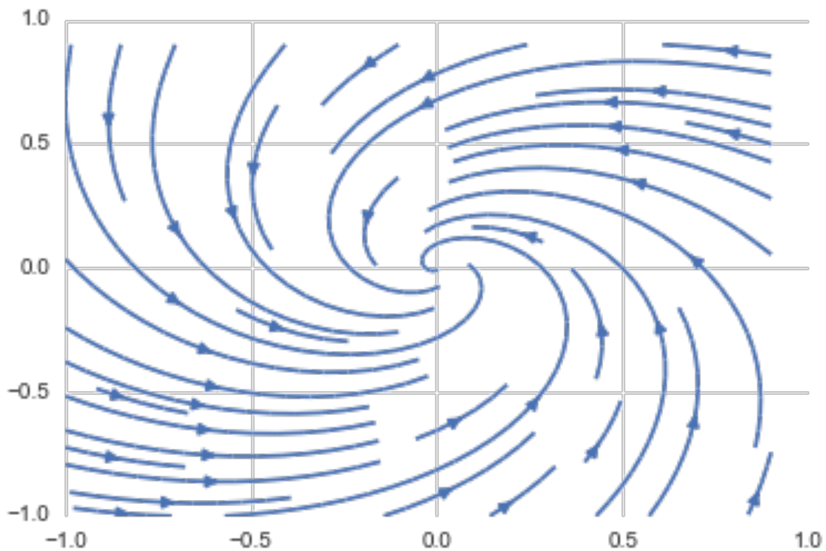
- Hold <Shift> and perform a movement
- Release <Shift> to stop recording
- Perform these steps as many times as needed
- Press <T> to toggle showing all stored recordings

Vector fields

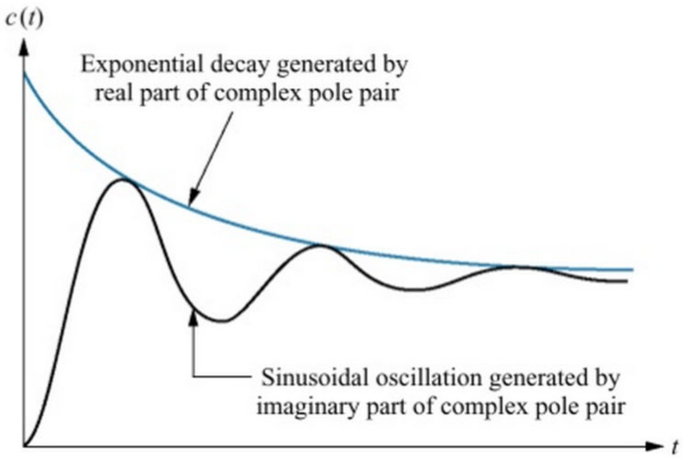


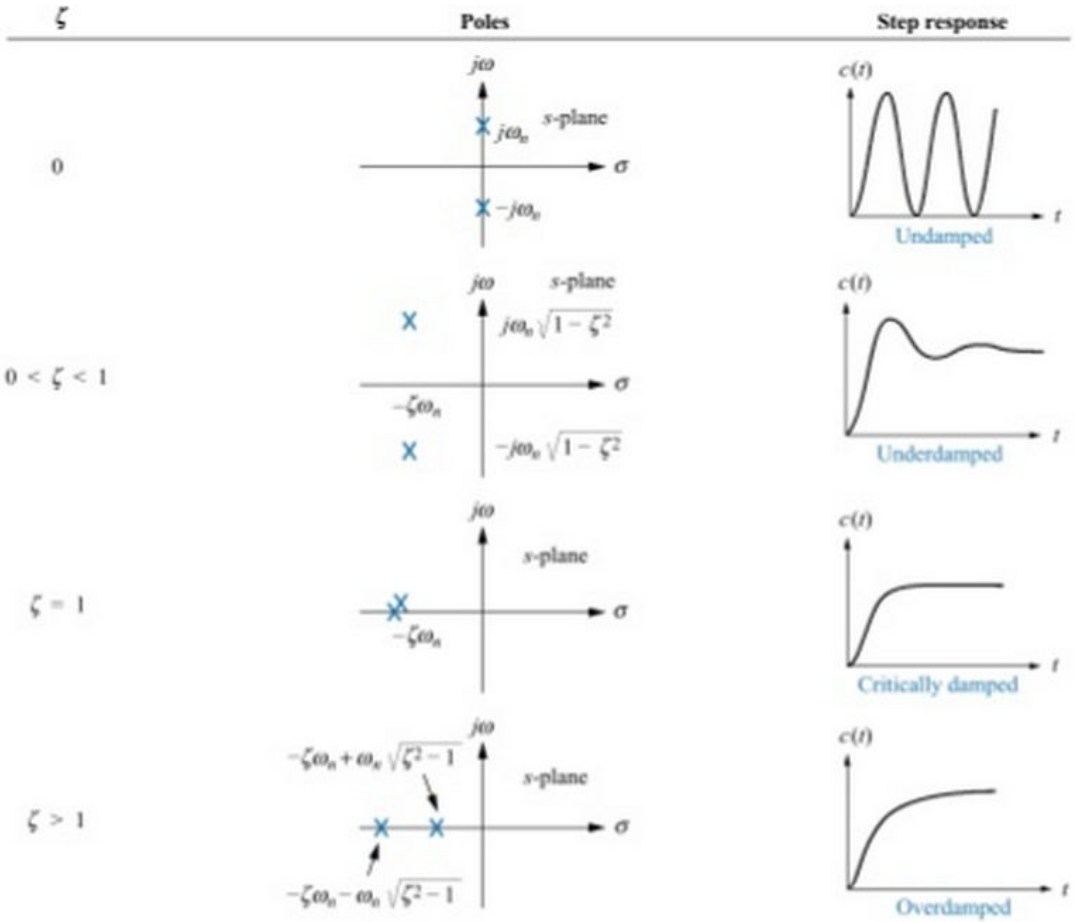


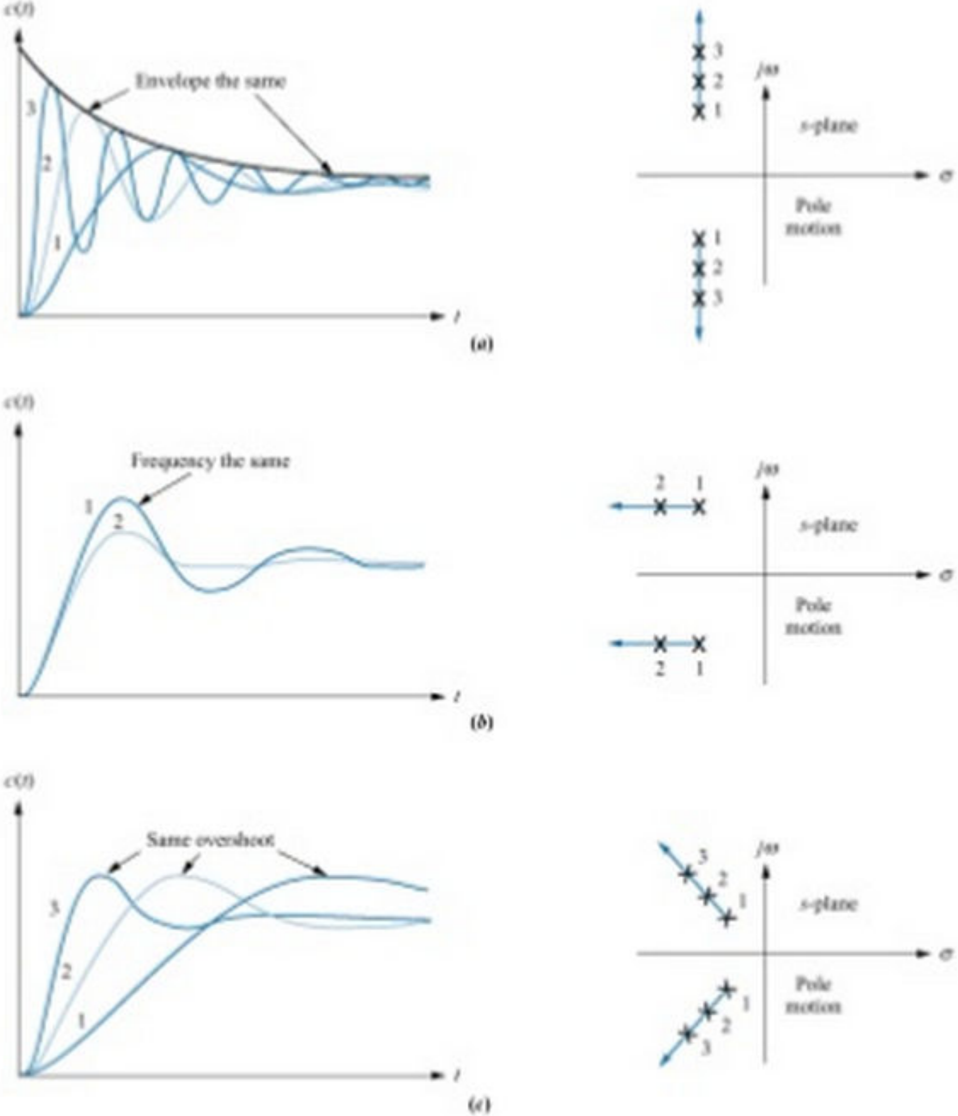
Streamline plots



Poles and zeros






c(t)
Envelope the same
jω
s-plane
σ
Pole motion
t
(a)
c(t)
Frequency the same
jω
s-plane
σ
Pole motion
t
(b)
c(t)
Same overshoot
jω
s-plane
σ
Pole motion
t
(c)

APPLICATIONS TO HCI RESEARCH

Moving beyond Fitts' law?

- Fitts proposed that the time MT to move to a target area is a function of the distance to the target (A) and the size of the target W,
  - $MT = a + b\ ID$
  - $ID = \log_2 (2A/W)$
- Movement times and error rates are important aspects of human interaction, but they do not provide a complete picture.

First order controller

If we imagine a step change, r from initial state $x = 0$, then the response of the first order lag will be an exponential response

$$x(t) = r(1 - e^{-kt}).$$

For a target sized w centered on r, then the time taken to get within $\frac{1}{2}w$ of r is

$$\begin{aligned}
x(t) &= r - \frac{1}{2}w \\
r(1 - e^{-kt}) &= r - \frac{1}{2}w \\
e^{-kt} &= \frac{w}{2r} \\
-kt &= \ln \frac{w}{2r} \\
t &= -\frac{1}{k} \ln \frac{w}{2r}
\end{aligned}$$

which, after converting to a base 2 logarithm, via $\log_a x = \frac{\ln x}{\ln a}$, is

$$t = \frac{\ln 2}{k} \log_2 \frac{2r}{w}, \tag{3}$$

which is similar in form to Fitts' ID, in equation (2). The gain k affects the speed of acquisition – the time constant for such a first order lag is $\frac{1}{k}$, the time it takes to reach 63% of the steady state response.

2^{nd} order controller

$$x(t) = r\left(1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\sin\left(\omega_n t\sqrt{1-\zeta^2} + \tan^{-1}\left(\frac{\sqrt{1-\zeta^2}}{\zeta}\right)\right)\right).$$

Here we assume that the target capture occurs at maximum and minimum oscillation, so the sinusoidal aspect is disregarded, giving

$$\begin{aligned}
x(t) &= r - \frac{1}{2}w \\
r\left(1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\right) &= r - \frac{1}{2}w \\
\frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}} &= \frac{w}{2r} \\
e^{-\zeta\omega_n t} &= \sqrt{1-\zeta^2}\,\frac{w}{2r} \\
-\zeta\omega_n t &= \ln\sqrt{1-\zeta^2} + \ln\frac{w}{2r} \\
t &= \frac{-1}{\zeta\omega_n}\ln\sqrt{1-\zeta^2} + \frac{\ln 2}{\zeta\omega_n}\log_2\frac{2r}{w}
\end{aligned}$$

which again leads to a movement time which is a linear function of the ID, with the time constant of $\frac{1}{\zeta\omega_n}$.

How do these compare to reality?

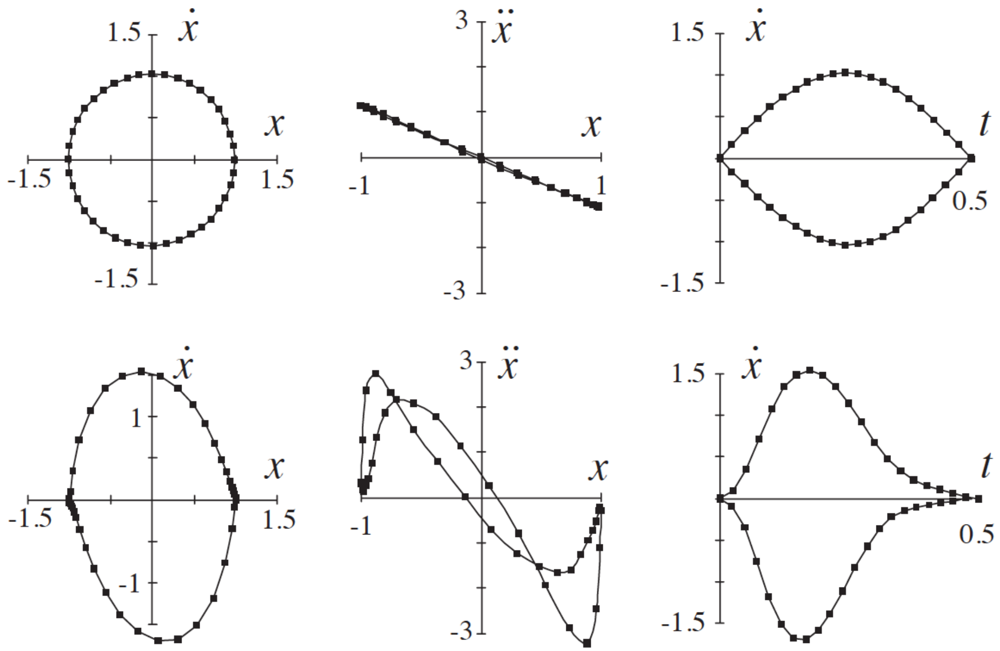
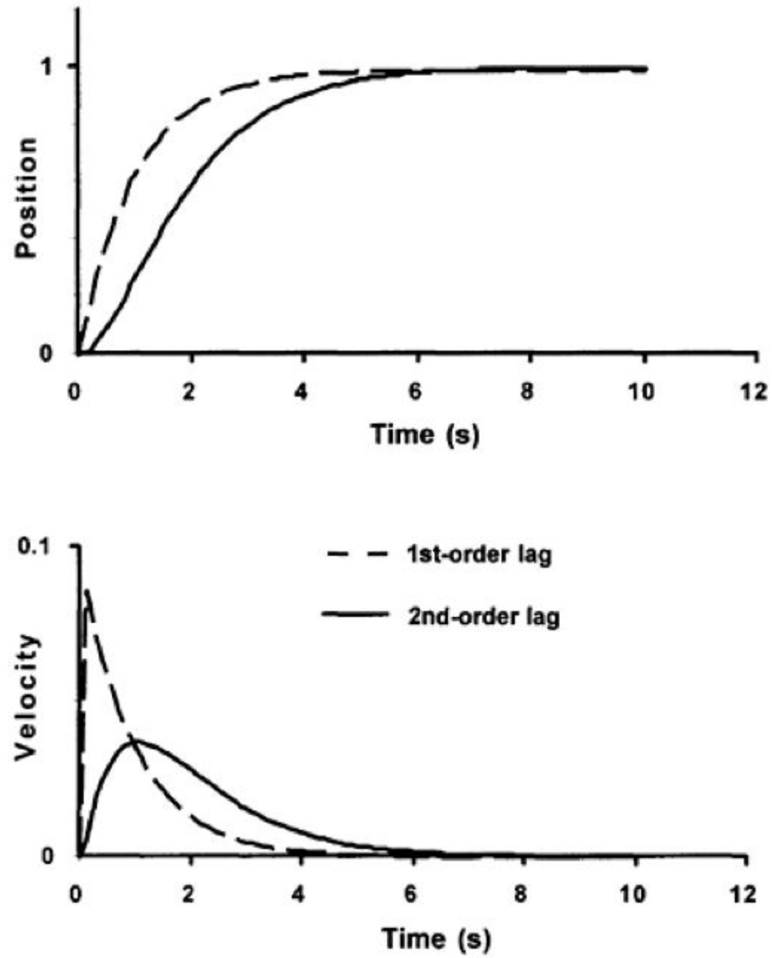
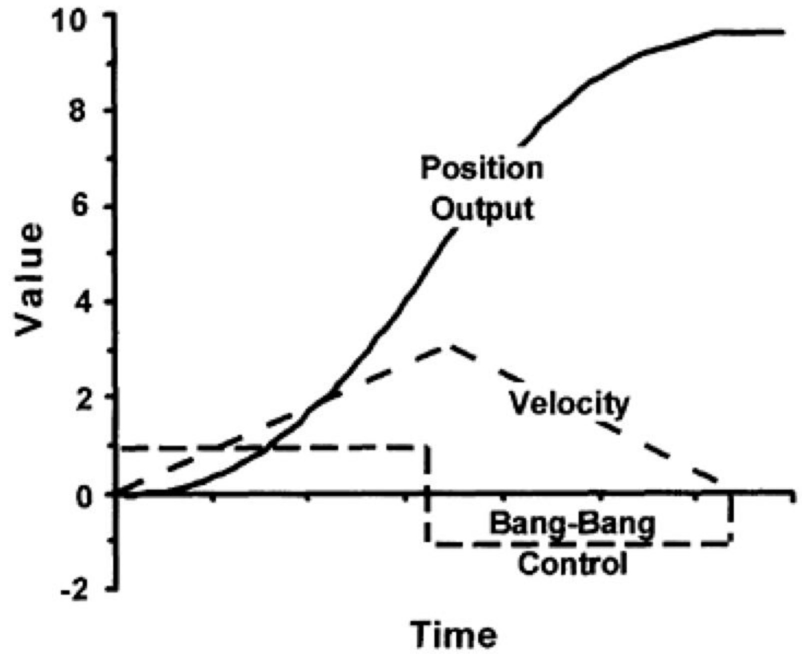
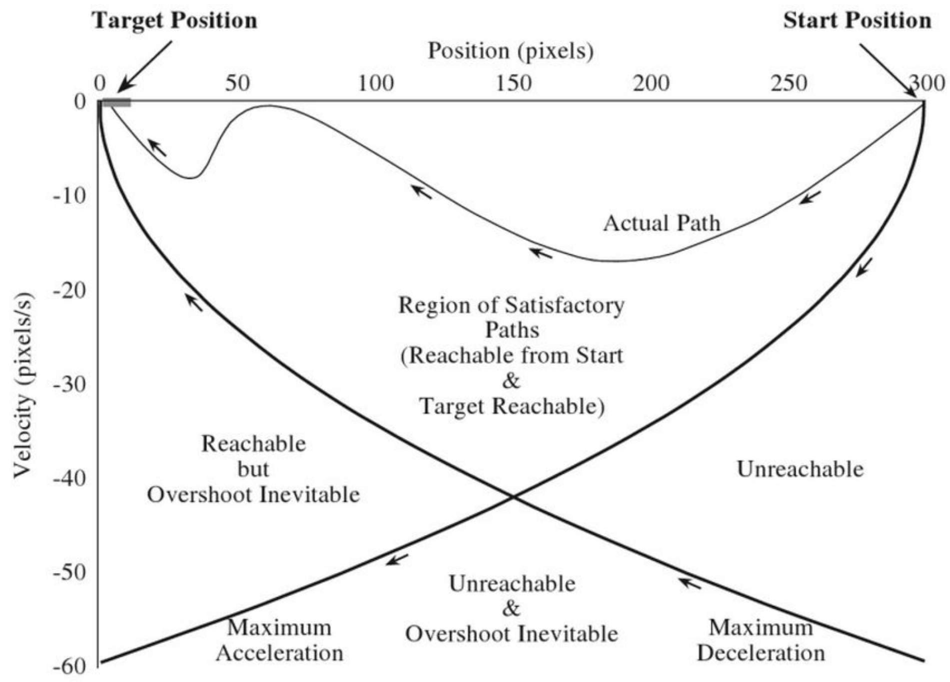


Fig. 1. Typical phase plot, Hooke plot and velocity profiles (from left to right) obtained in reciproca
aiming with $ID = 3$ (upper panels) and with $ID = 7$ (lower panels). When task difficulty is raised, phas
plot becomes skewed to the top-left, Hooke plot takes an asymmetric N shape, and velocity profiles ar
asymmetric bell shape with a longer deceleration phase.



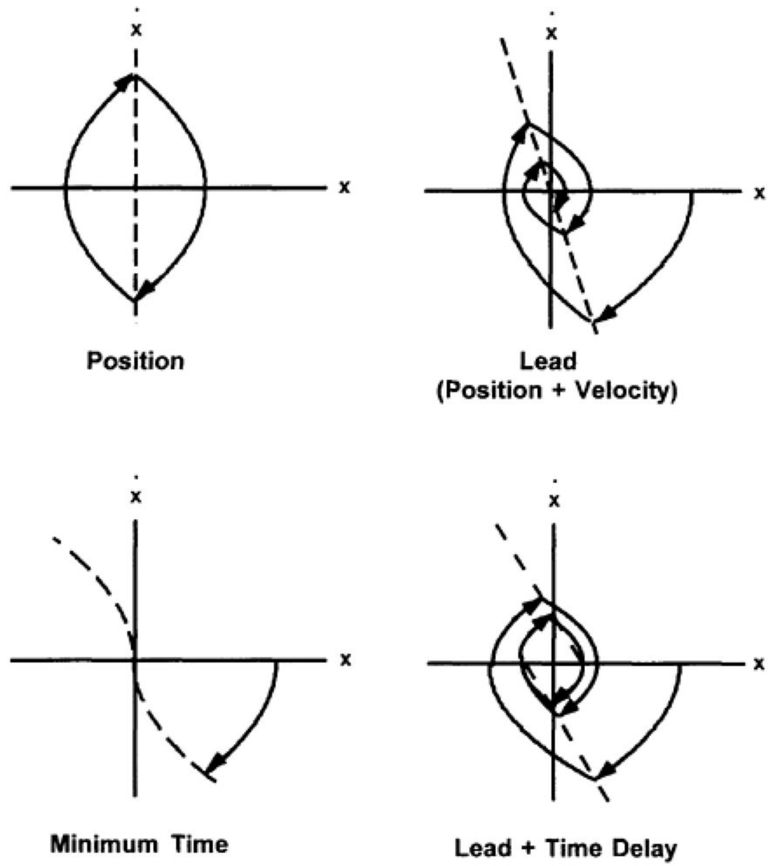
Switching controller





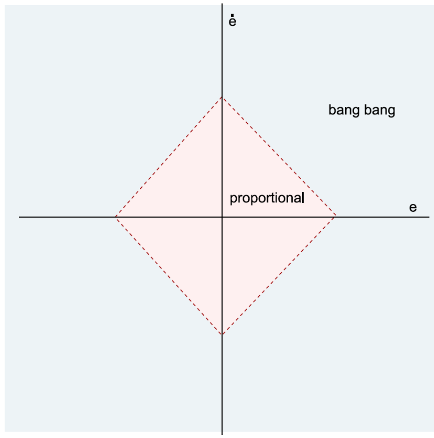
Switching functions

- Multiple actions can get you to goal
- Iterative approximations model can also lead to Fitts-like time responses.



Surge control

- current state lies outside the diamond
  - bang-bang (full on or full off) control is applied,
- Until signal enters central region
  - where proportional line control takes over.



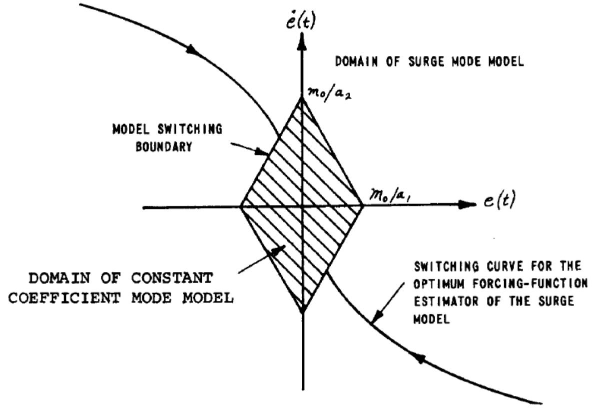


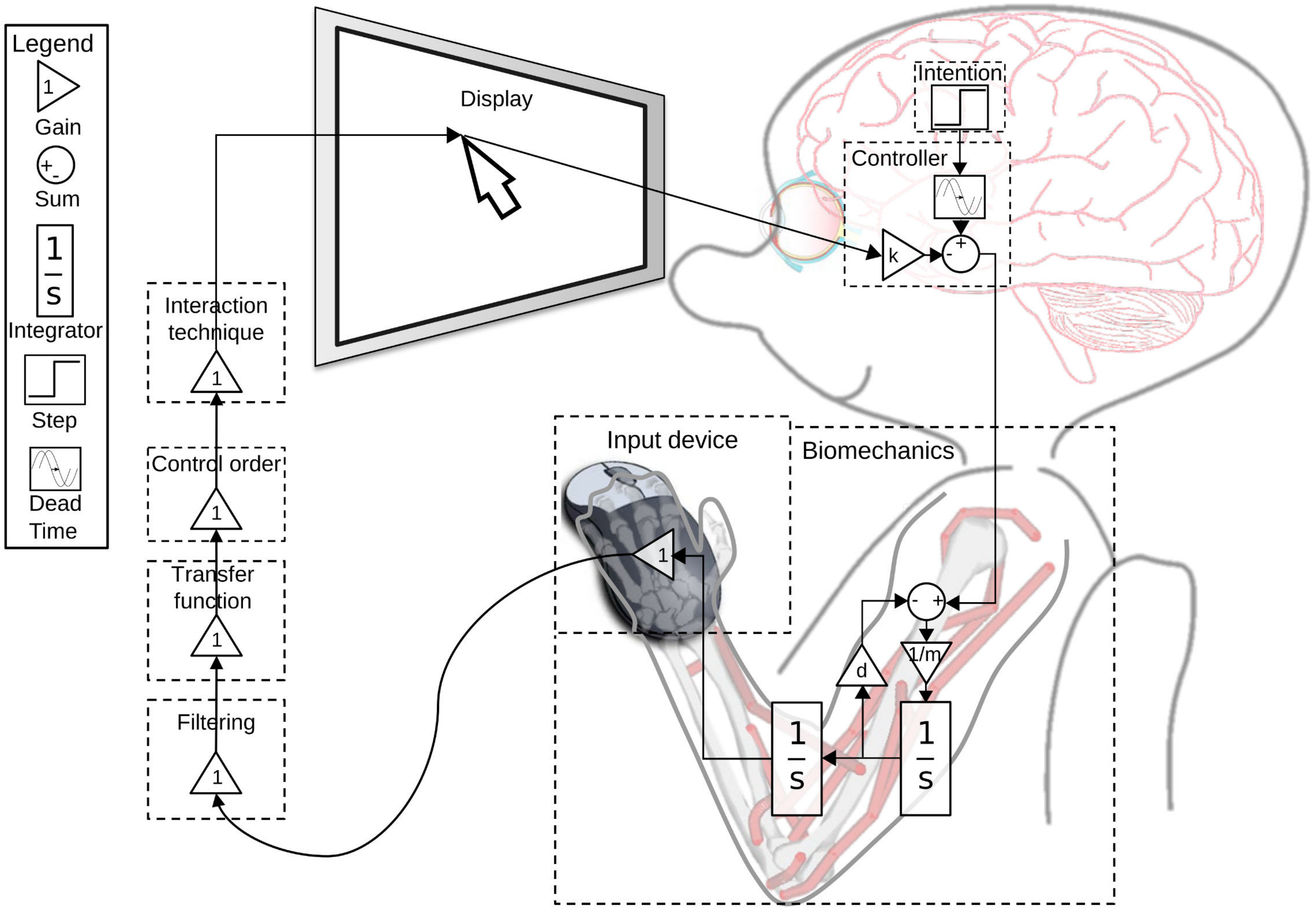
Fig. 4. The switching curve of the proposed optimum forcing-function estimator, superimposed on the model switching boundary.

Control Theoretic Models of Pointing,
J. Müller, A. Oulasvirta, R. Murray-Smith

ToCHI - to appear

EXPERIMENTAL WORK

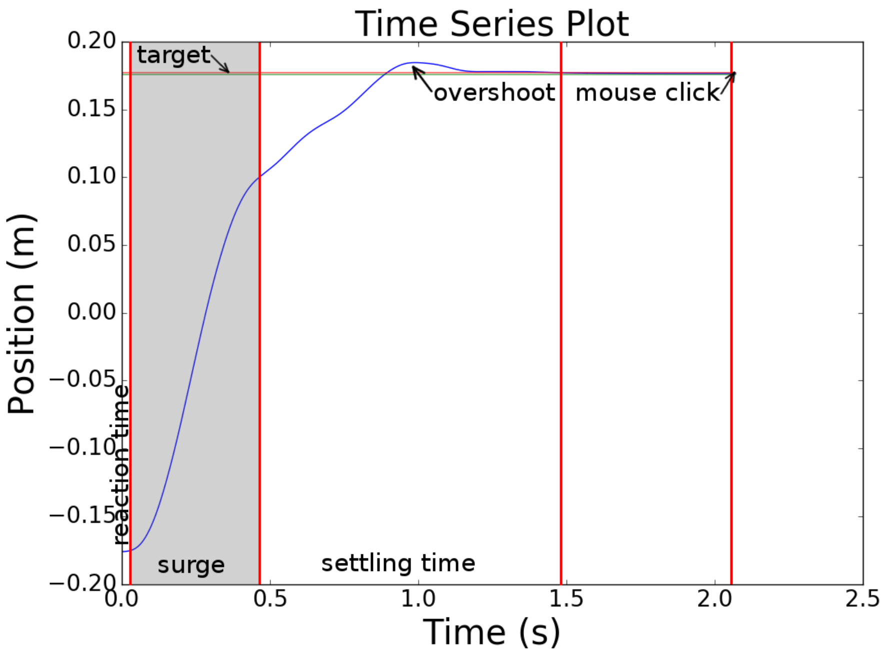
Control block diagram for continuous HCI

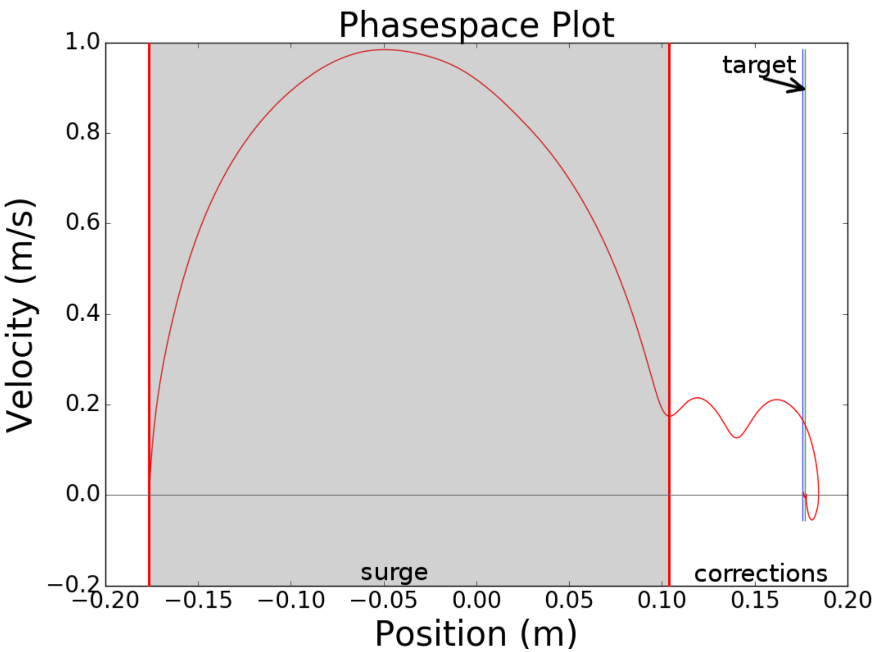


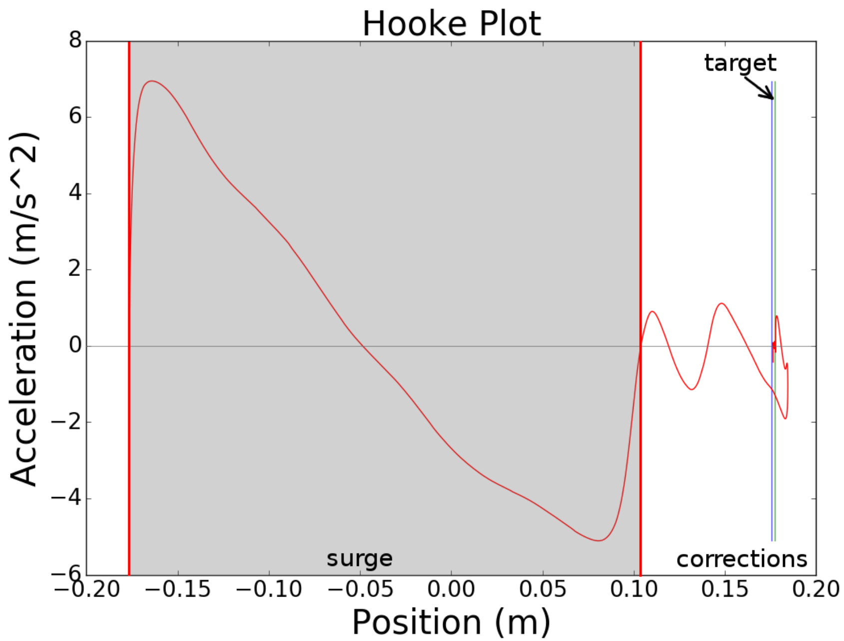
Manual control models

In this work we compared 4 classical manual control models:

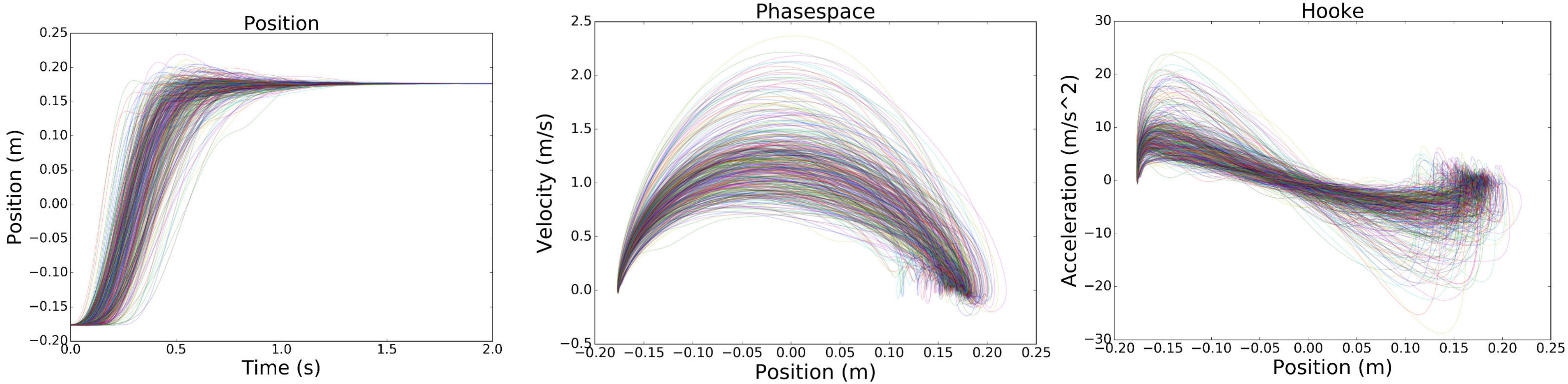
1. Second-order lag, linear model with delay
2. McRuer's Crossover model
3. Bang-Bang control, with delay
4. Costello's Surge model, with delay

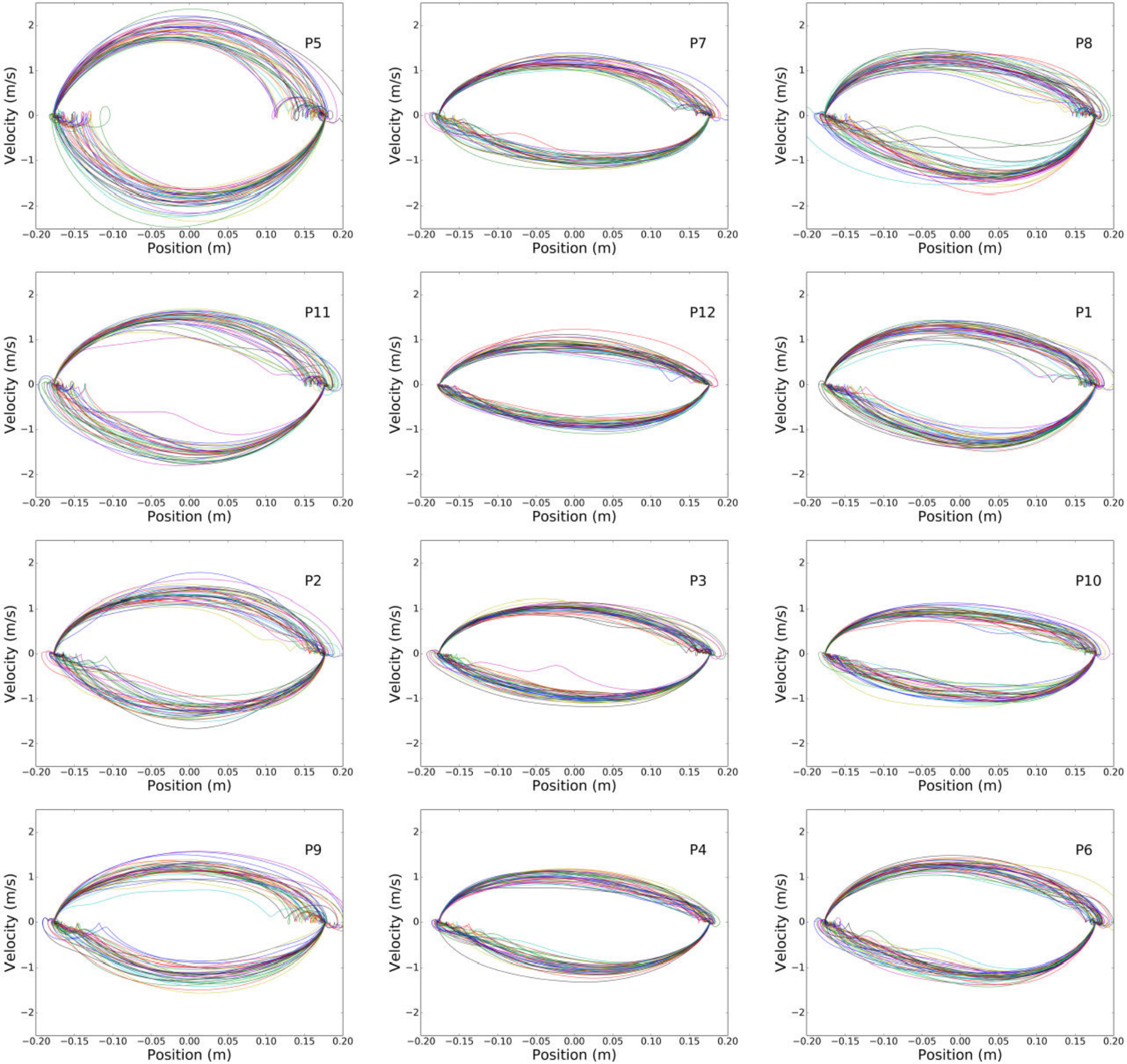

Time Series Plot
target
overshoot
mouse click
reaction time
surge
settling time
Position (m)
Time (s)
0.20
0.15
0.10
0.05
0.00
−0.05
−0.10
−0.15
−0.20
0.0
0.5
1.0
1.5
2.0
2.5

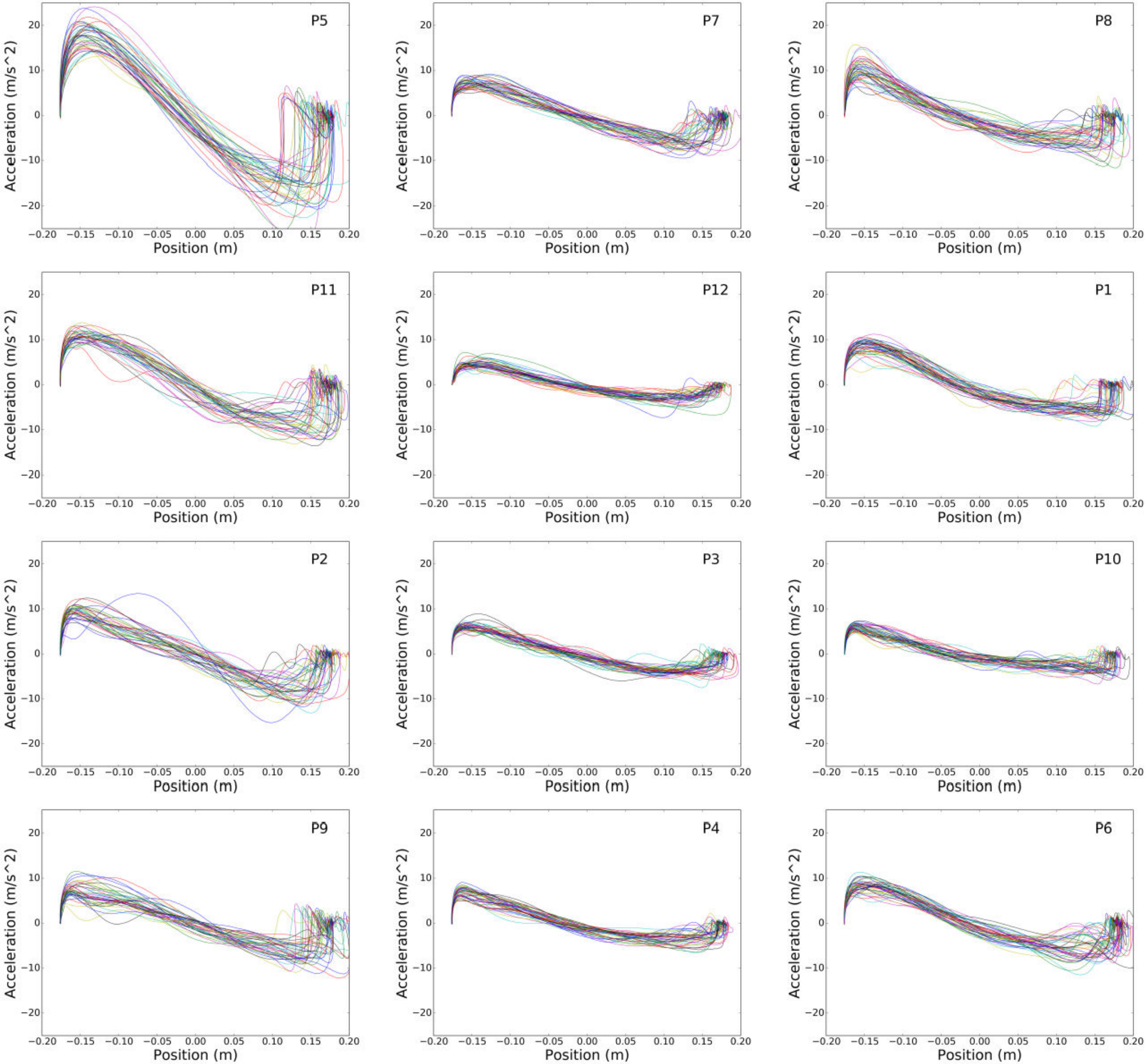

Phasespace Plot
target
surge
corrections
Velocity (m/s)
Position (m)
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.20
−0.15
−0.10
−0.05
0.00
0.05
0.10
0.15
0.20

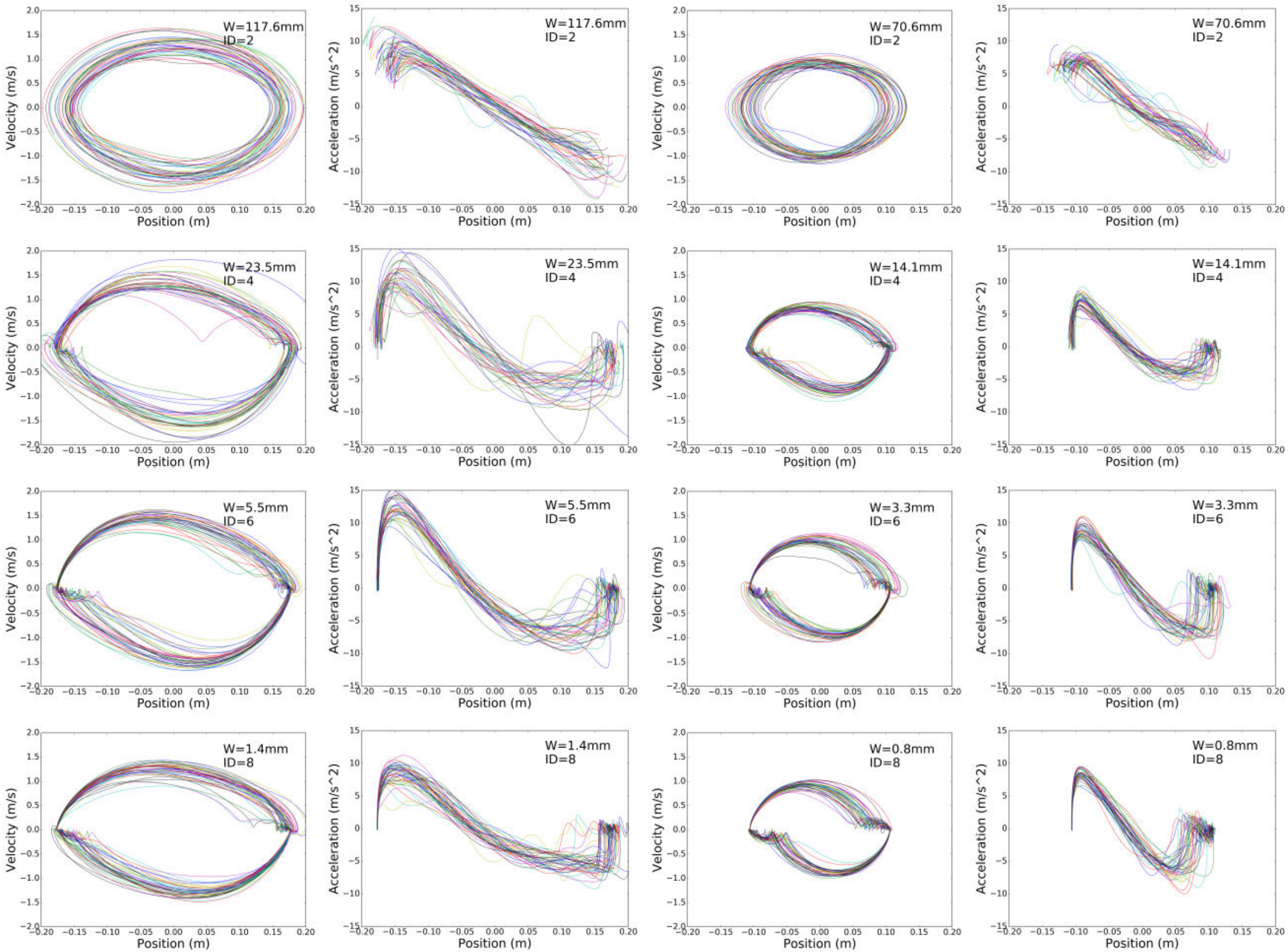

Hooke Plot
target
surge
corrections
Acceleration (m/s^2)
Position (m)
8
6
4
2
0
−2
−4
−6
−0.20
−0.15
−0.10
−0.05
0.00
0.05
0.10
0.15
0.20

Time series of derivatives



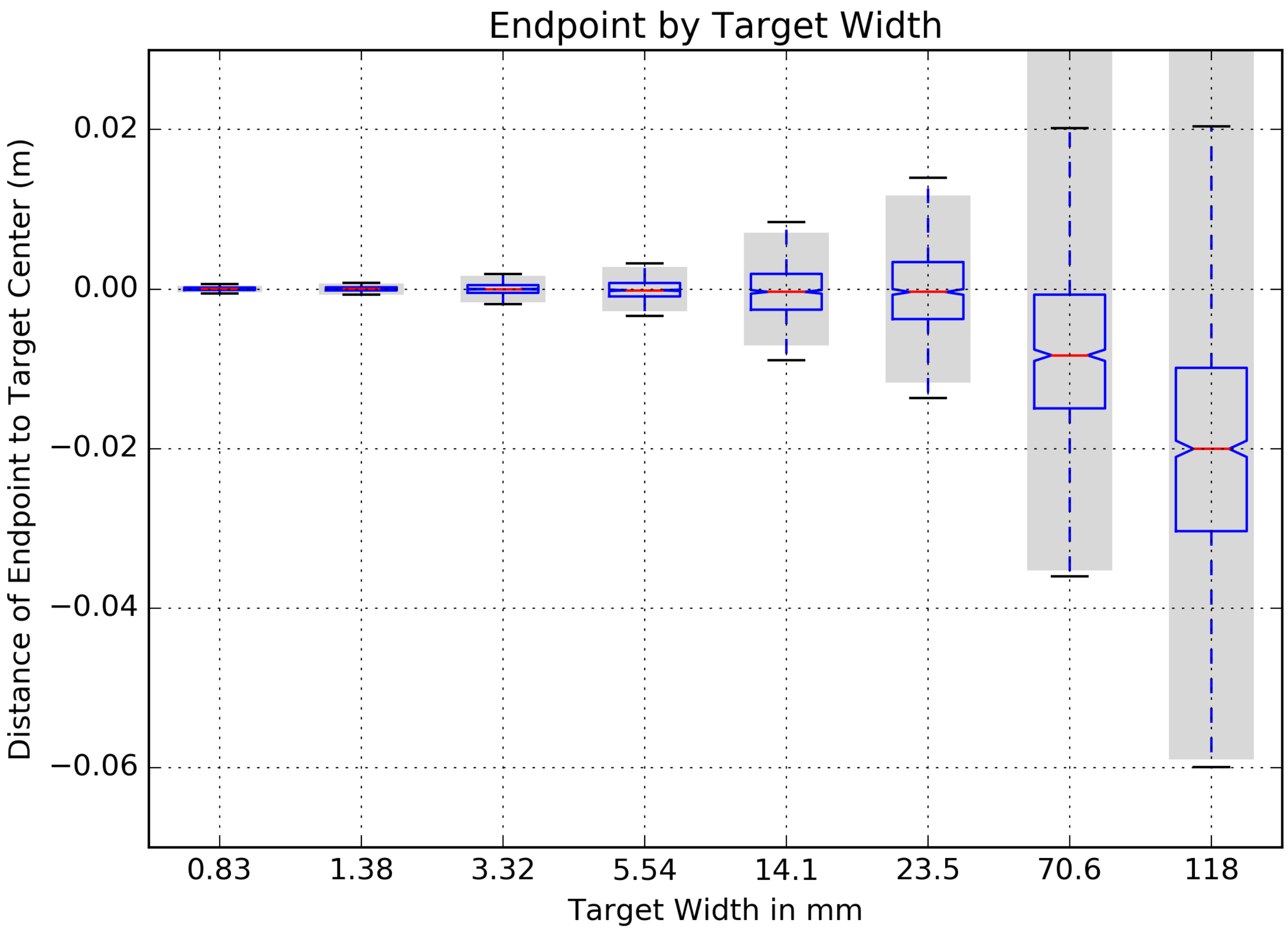

P5
P7
P8
P11
P12
P1
P2
P3
P10
P9
P4
P6
Velocity (m/s)
Position (m)


P5
P7
P8
P11
P12
P1
P2
P3
P10
P9
P4
P6
Acceleration (m/s^2)
Position (m)



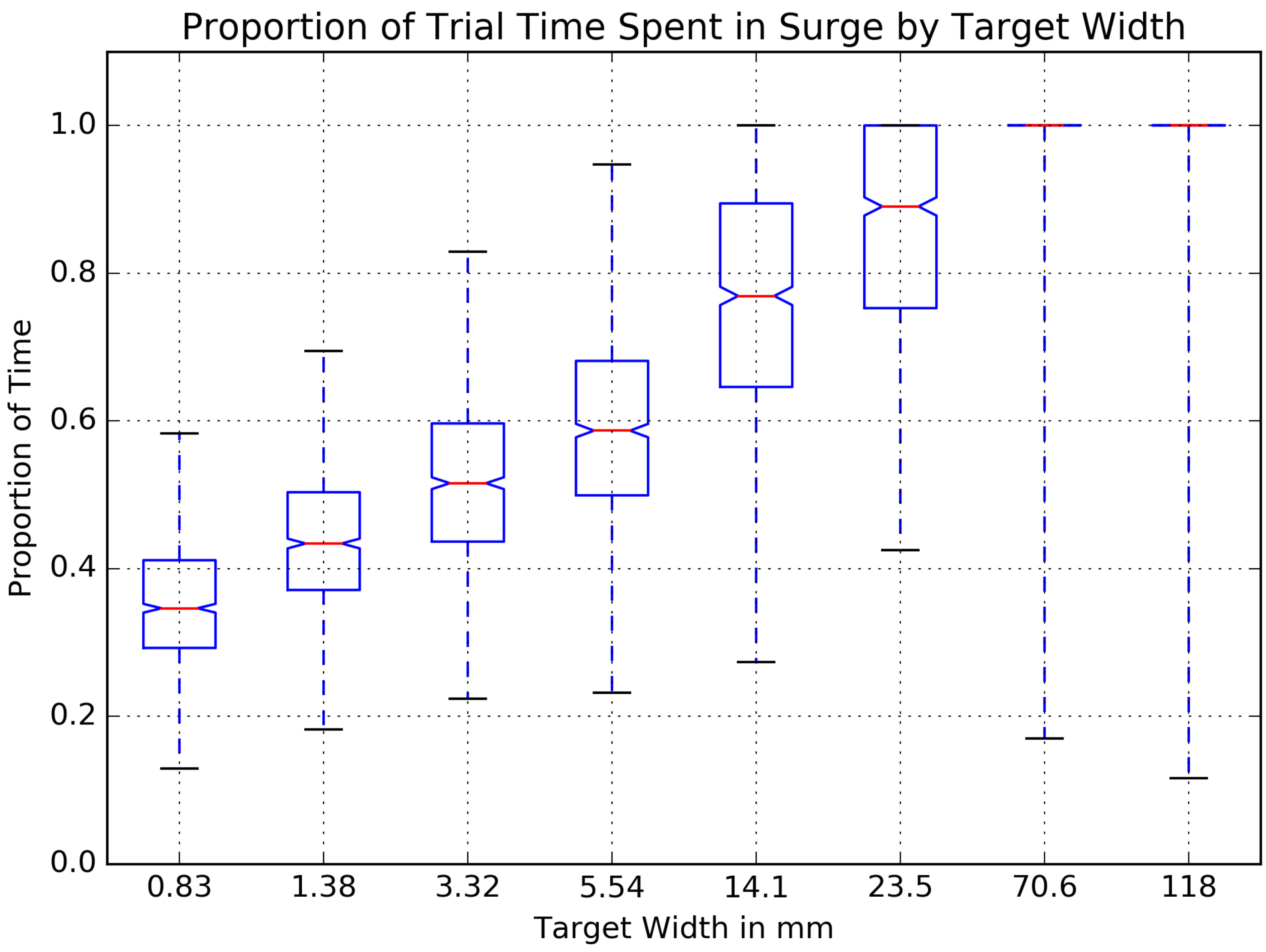
W=117.6mm
ID=2
W=70.6mm
ID=2
W=23.5mm
ID=4
W=14.1mm
ID=4
W=5.5mm
ID=6
W=3.3mm
ID=6
W=1.4mm
ID=8
W=0.8mm
ID=8
Velocity (m/s)
Acceleration (m/s^2)
Position (m)

Endpoint position

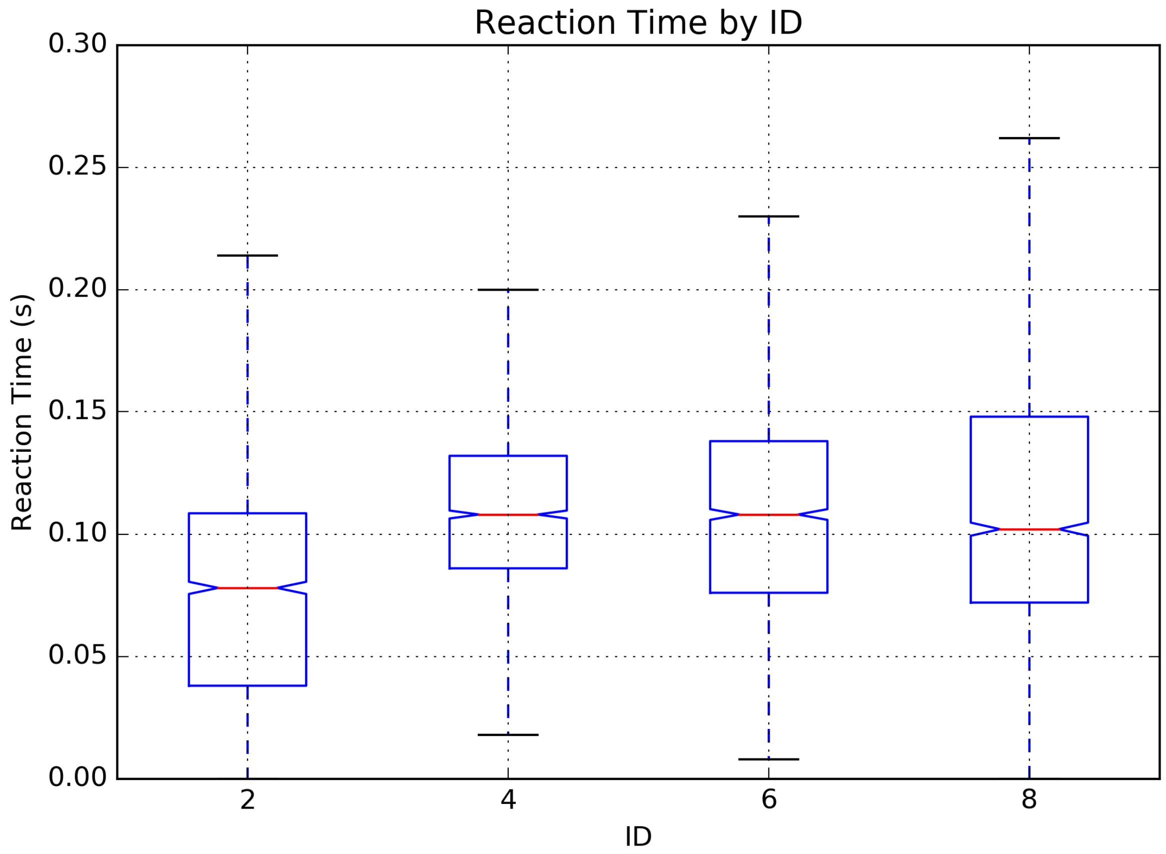


- Relates to *effective width* in Fitts analayses
- Note the bias towards undershoot for larger targets
    - no longer zero mean

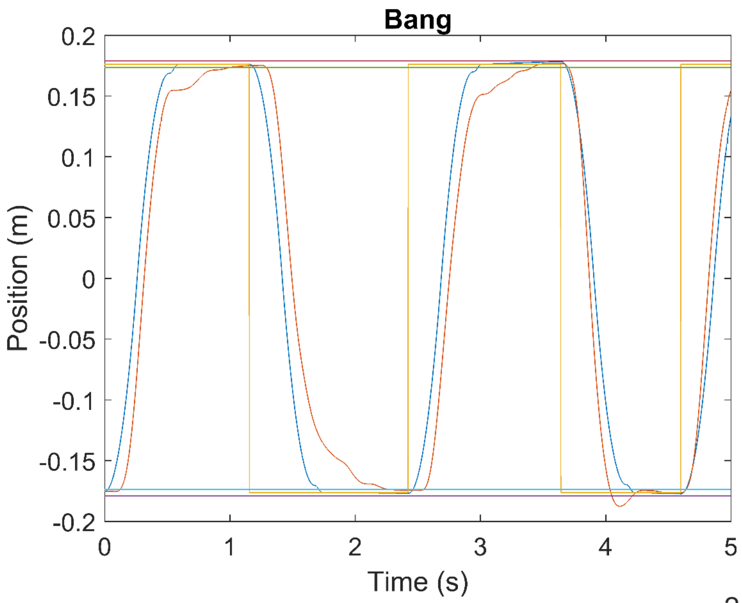
Proportion of movement in surge

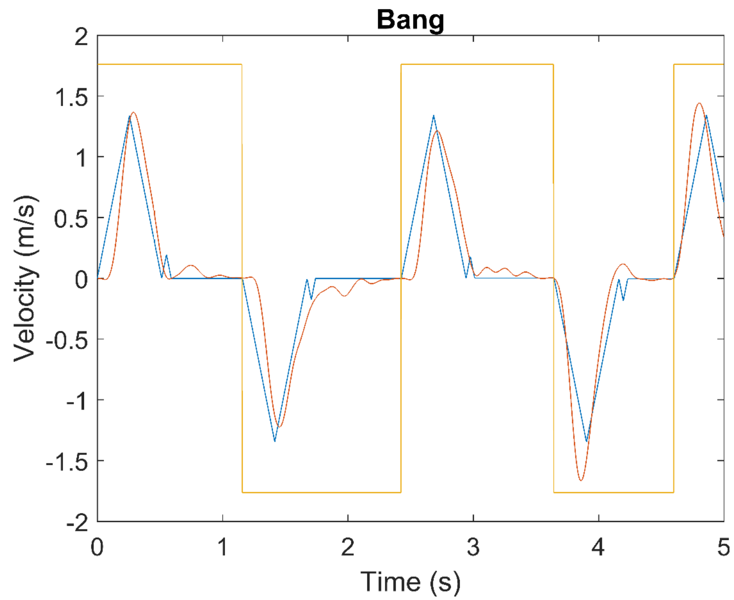


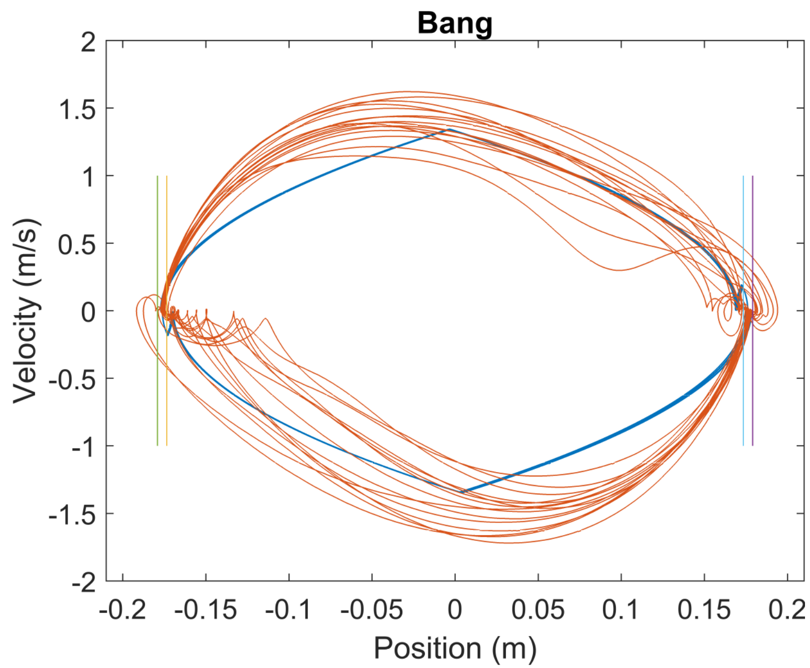
Reaction time by ID



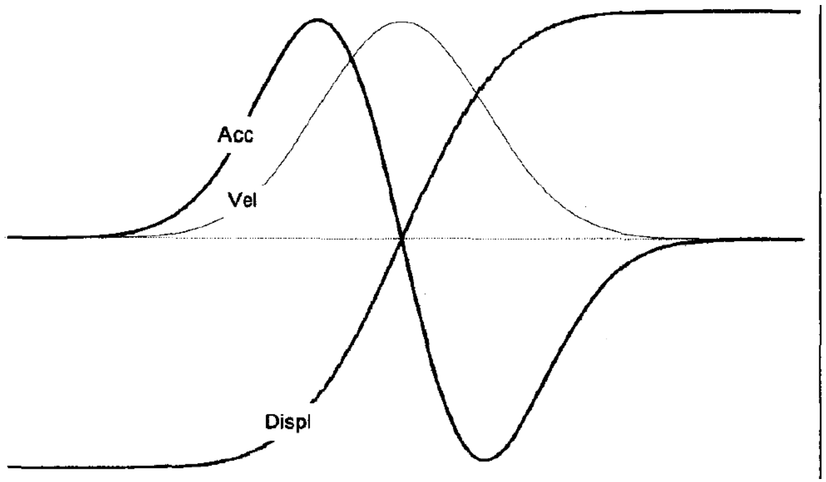
Bang-Bang model







Complexity of higher derivatives



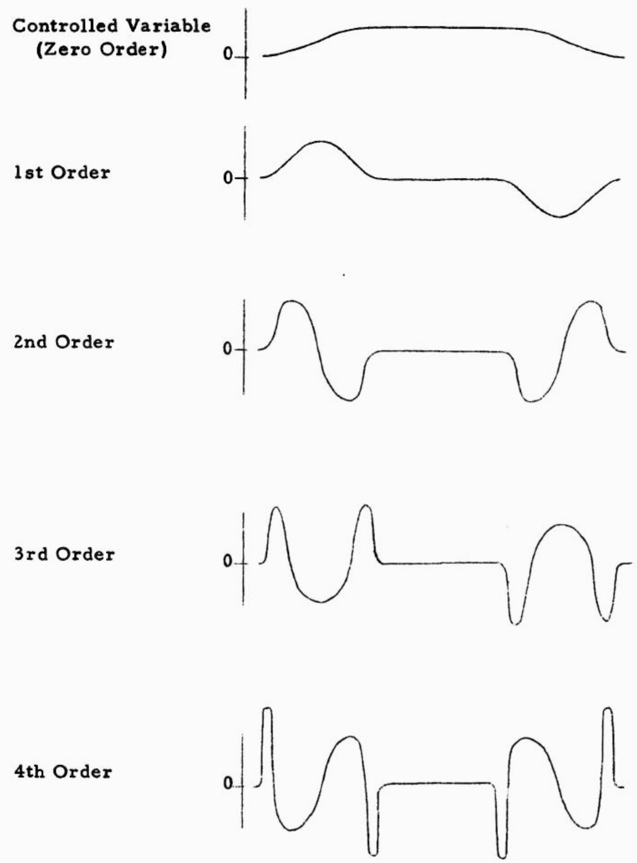
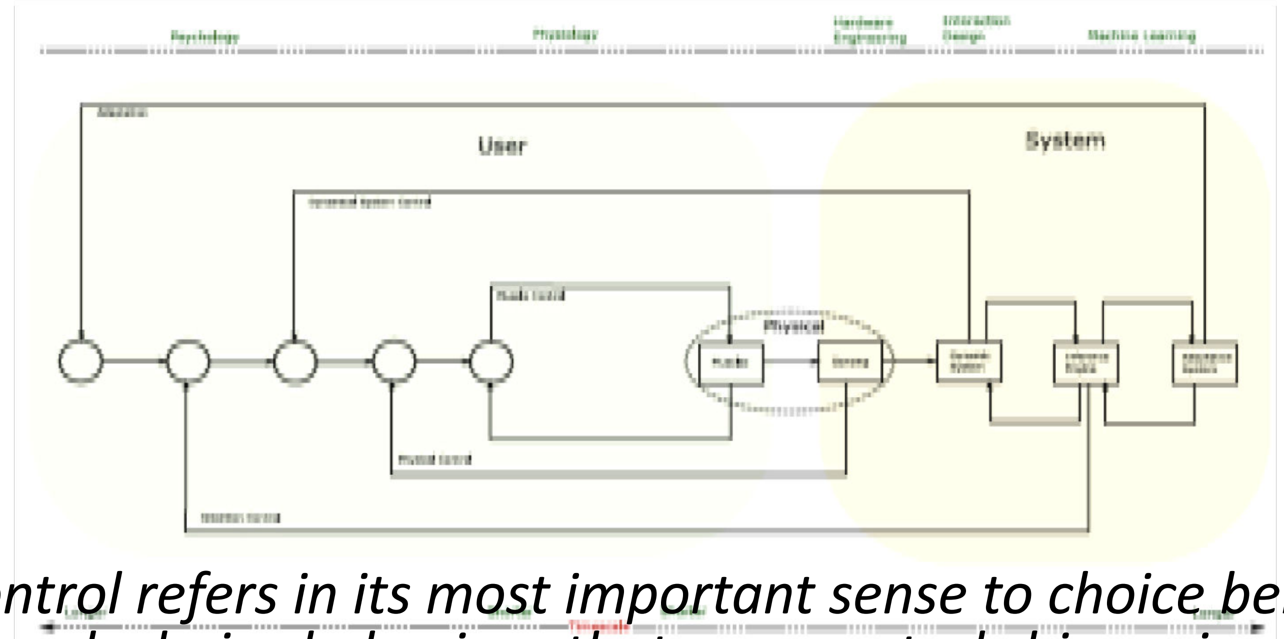


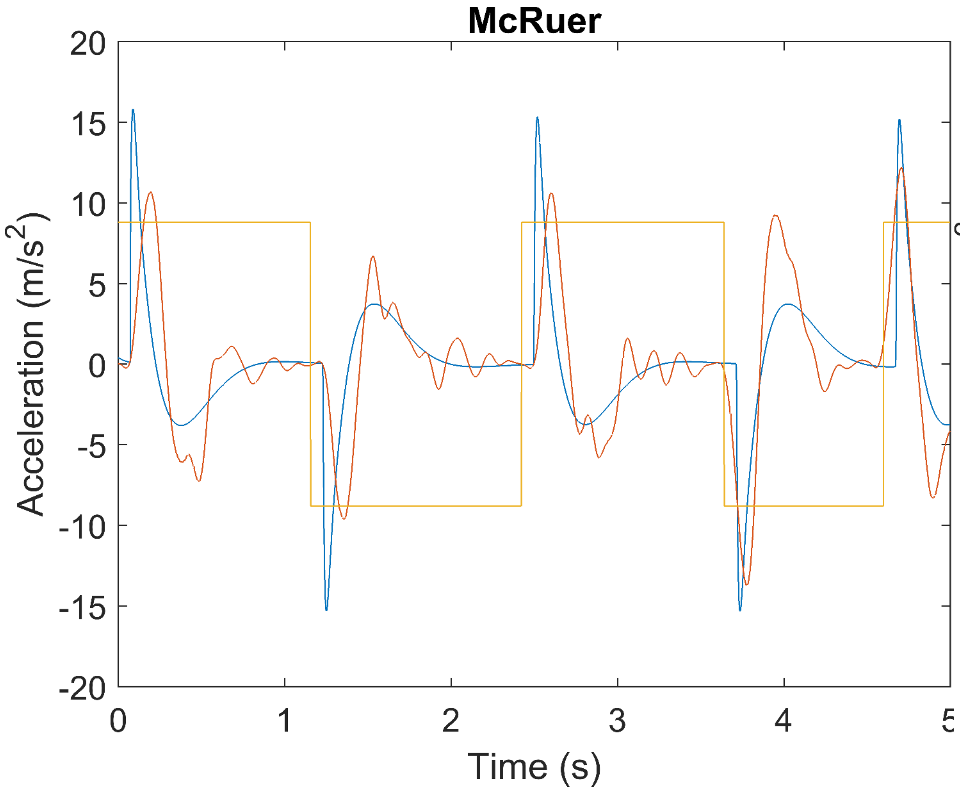
Figure III-3. Control Order illustrated by changes in a controlled variable and in its first four derivatives.

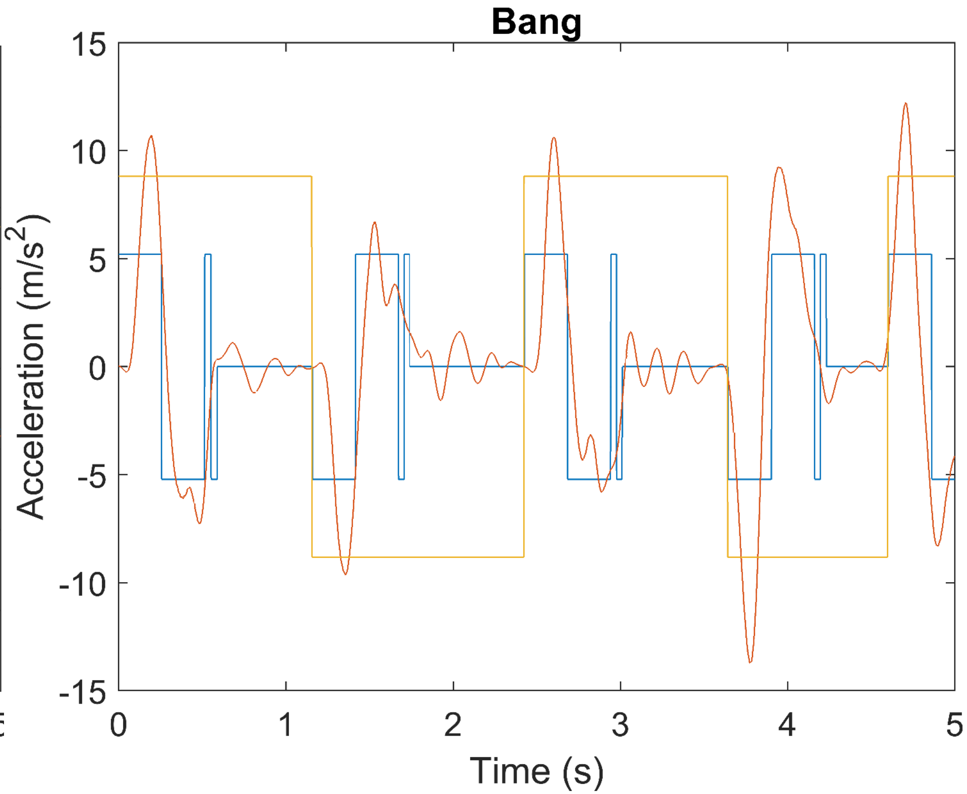
Hierarchical control



"Control refers in its most important sense to choice behaviour; it is through choice behaviour that man controls his environment. An important ingredient of such control is the operation, often automatic, of inner loops which regulate processes required in order that choices made in outer loops may be realised. Both outer-loop choice behaviour and the inner-loop automatic behaviour are aspects of control." (Kelley 1968)

c.f. McRuer model



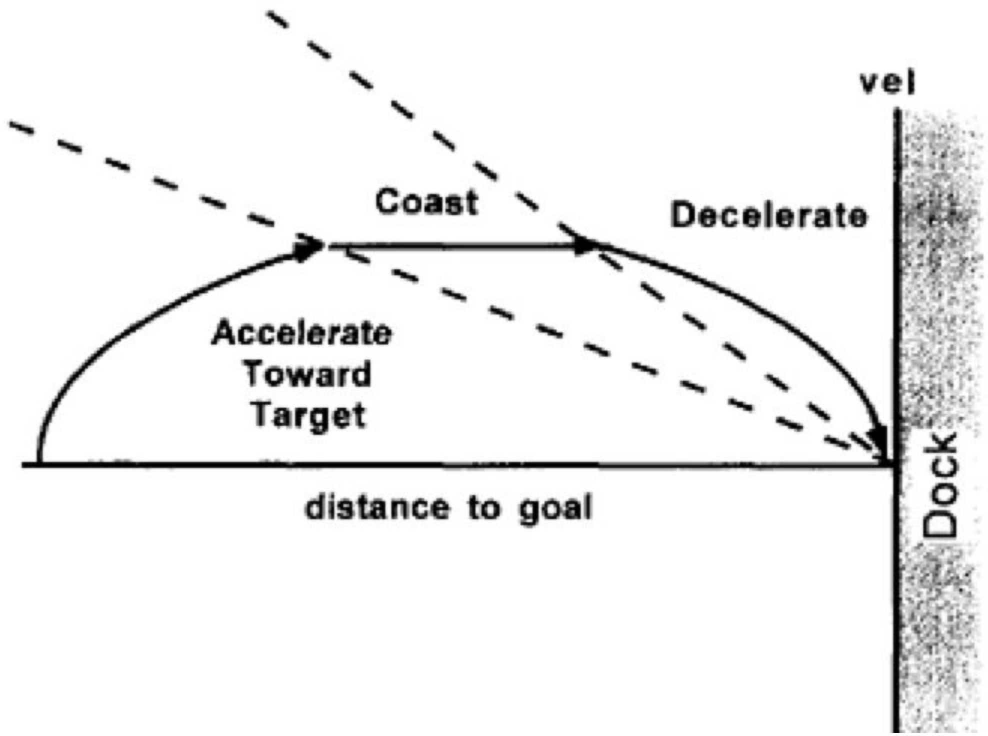


Open FittsLab.ipynb

PRACTICAL 1 – DYNAMICS & FITTS' LAW

How would you describe scrolling to a target in a document?

Finite-state controls



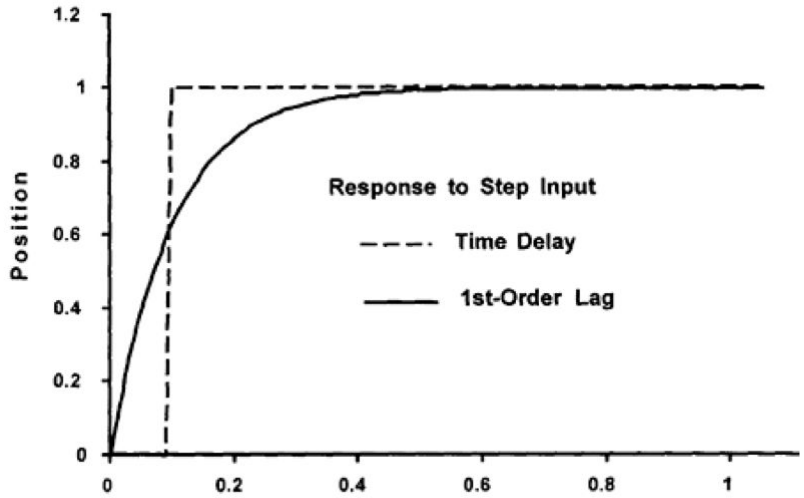
MODELS OF HUMAN CONTROL BEHAVIOUR

Theoretical frameworks for interaction?

- We do have some models of:
  - *Human* aspects, e.g. low-level perception, motor control, short-term memory and attention
  - *Machine* aspects, e.g. inference algorithms, FSMs, Statecharts etc.
- But less theoretical support in the area of *interaction*
  - Manual control theory provided first steps, with aircraft and automobile control
  - But computers then were essentially discrete interaction
  - Now things are becoming more continuous again.

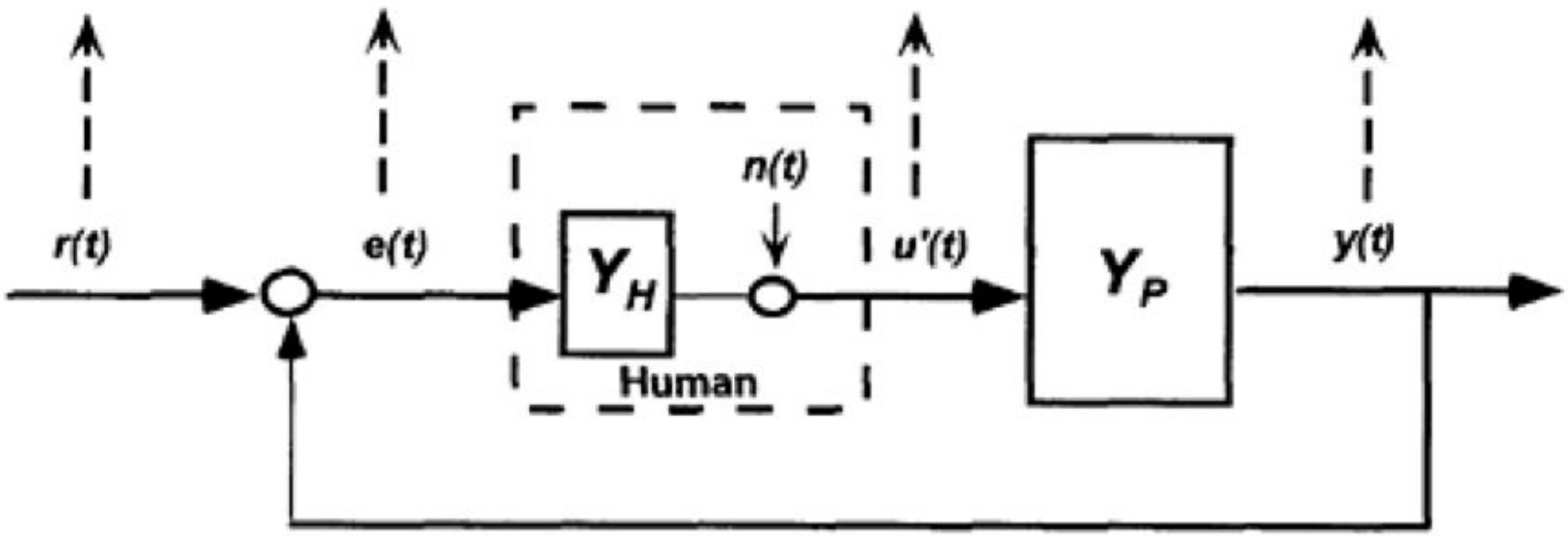
Delays

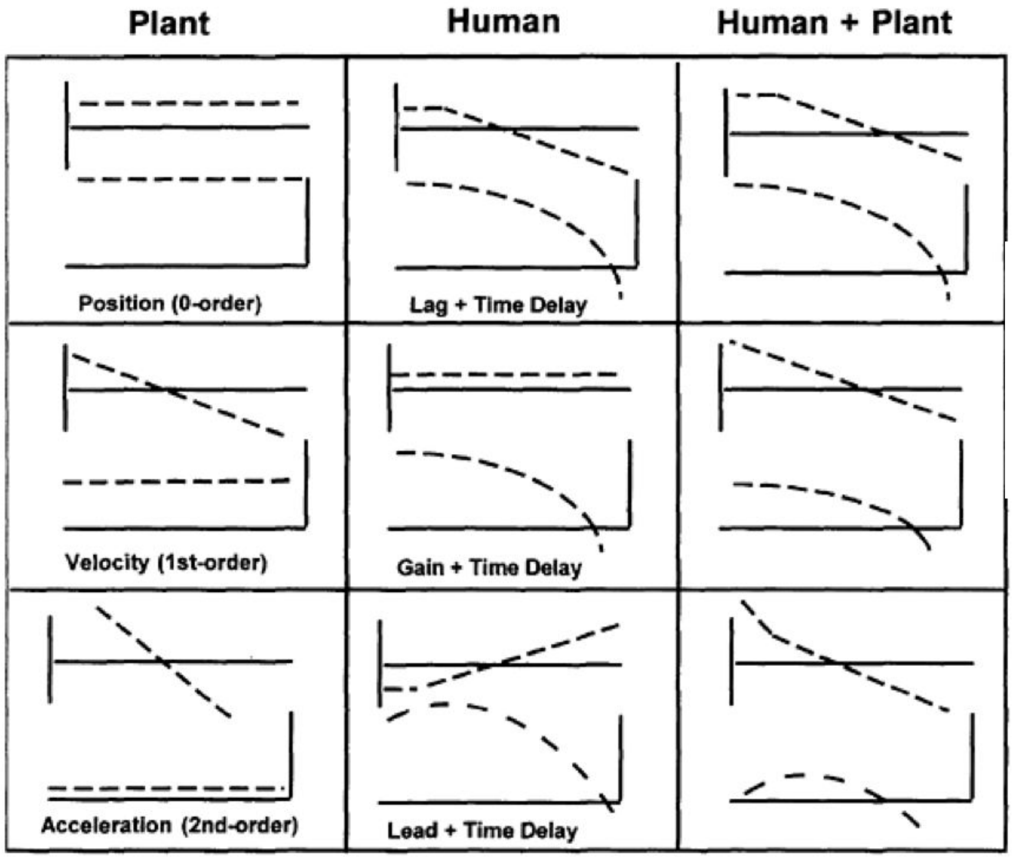
- Delay – pure time delay
- Lag – 1st or 2nd order system response
- Sometimes combination
  - E.g. cognitive pure time delay combined with time lag associated with muscle dynamics
- Variability in delay can be worse than fixed delay (e.g. network jitter)



Delay and lag model

$$Y_H(j\omega) = \frac{Ke^{-\tau j\omega}}{(T_I j\omega + 1)}$$





$$Y_H(j\omega)Y_P(j\omega) = \frac{\omega_c e^{-\tau j\omega}}{j\omega}$$

FIG. 14.6. The schematic illustration shows the adaptive nature of the human controller. Note that the human transfer function (amplitude ratio and phase shift) changes depending on the system being controlled. The changes reflect the stability constraints on the total system (human+plant). The invariant form of the

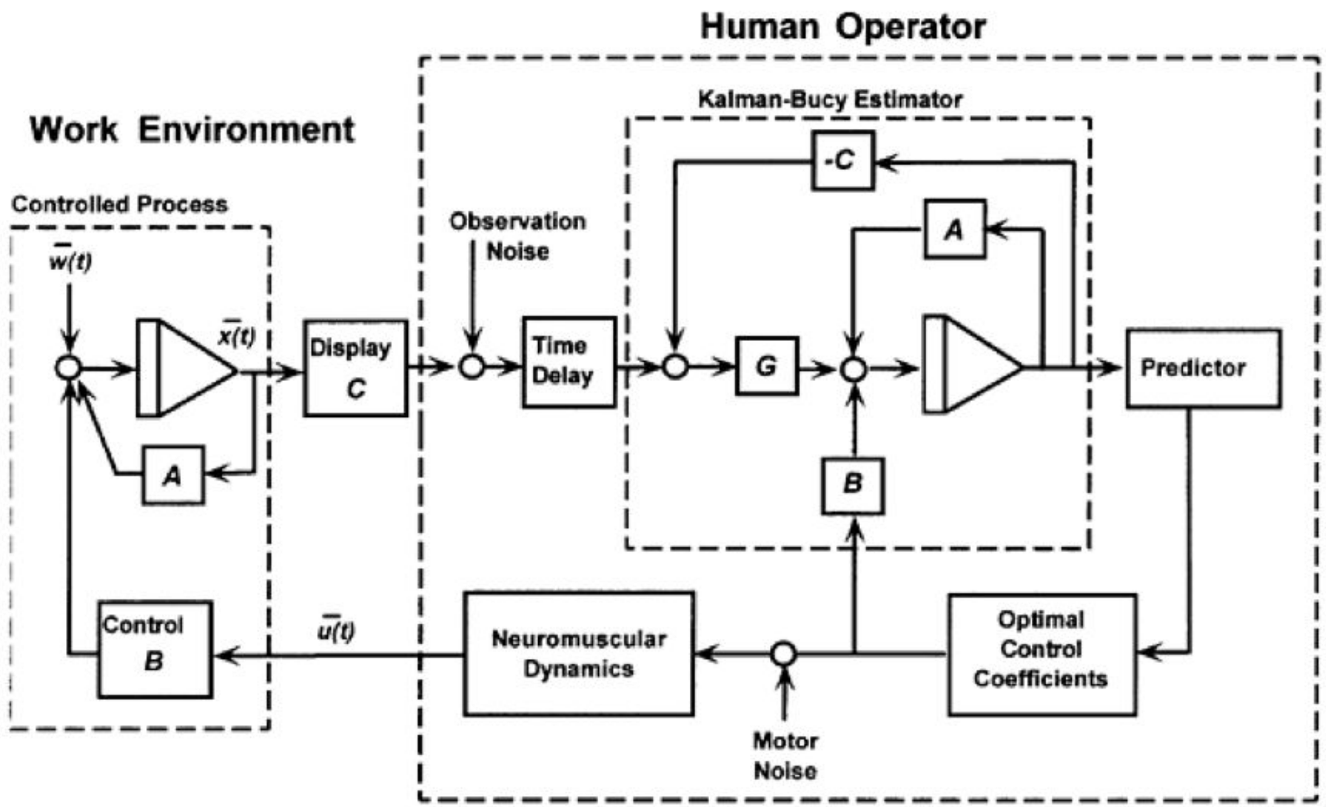
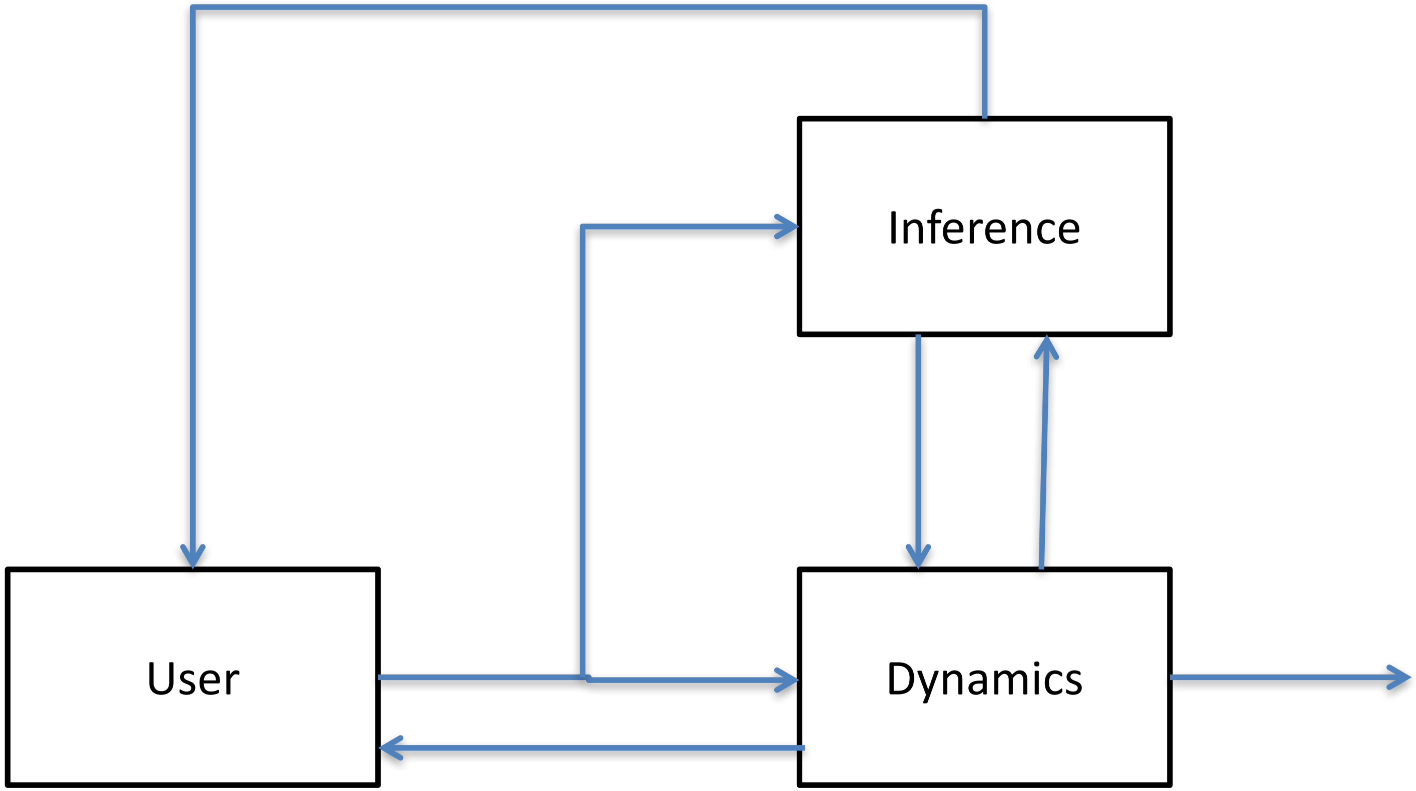


FIG. 17.4. The Baron-Kleinman-Levison optimal control model. From *"Man-Machine Systems: Information, Control and Decision Models of Human Performance"'* (p. 254) by T.B.Sheridan and W.R.Ferrell, 1974, Cambridge, MA: MIT Press. Adapted by permission.


Inference
User
Dynamics

- Control theory perspective
  - We have evolved to control our perceptions. We require feedback, and there are upper limits on our bandwidth.
  - User interacting with interface object viewed as two coupled dynamic systems
  - Physical model-based approach to representation of interface objects
  - Dynamics allows us to slip in 'intelligence' into the closed-loop which couldn't be done with a static interaction technique

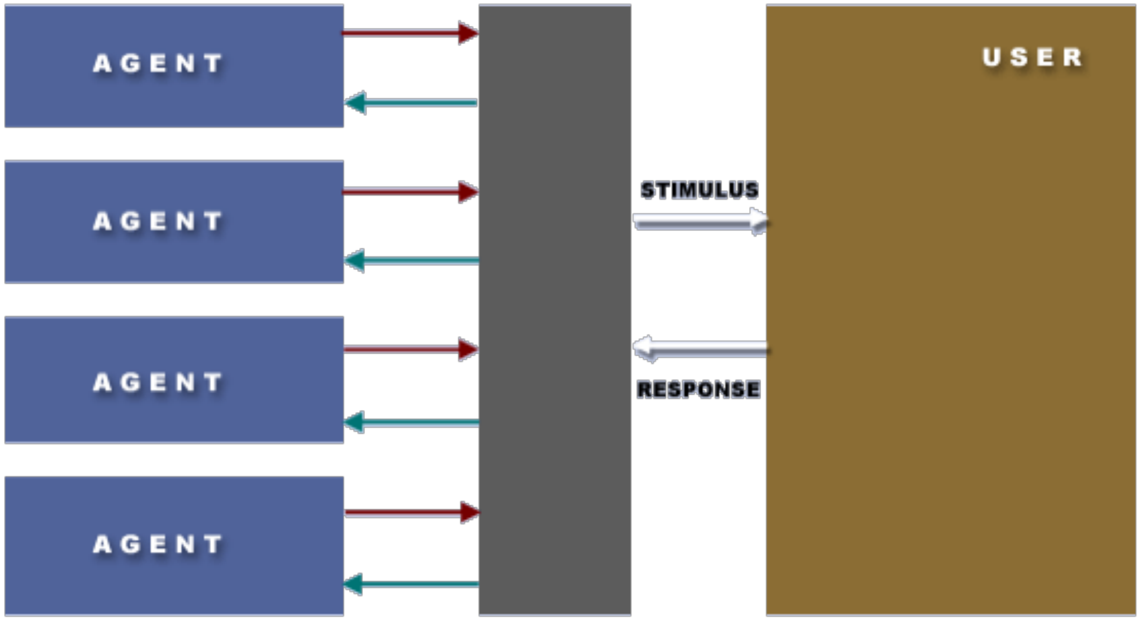
Feedback

- Feedback can dramatically cut down the space over which inference must be performed.
  - A communicating system which provides feedback knows precisely what it has communicated to the user previously (although not how it was interpreted).
  - Any inference performed can be conditioned on the feedback provided to the user, reducing the recognition problem significantly.

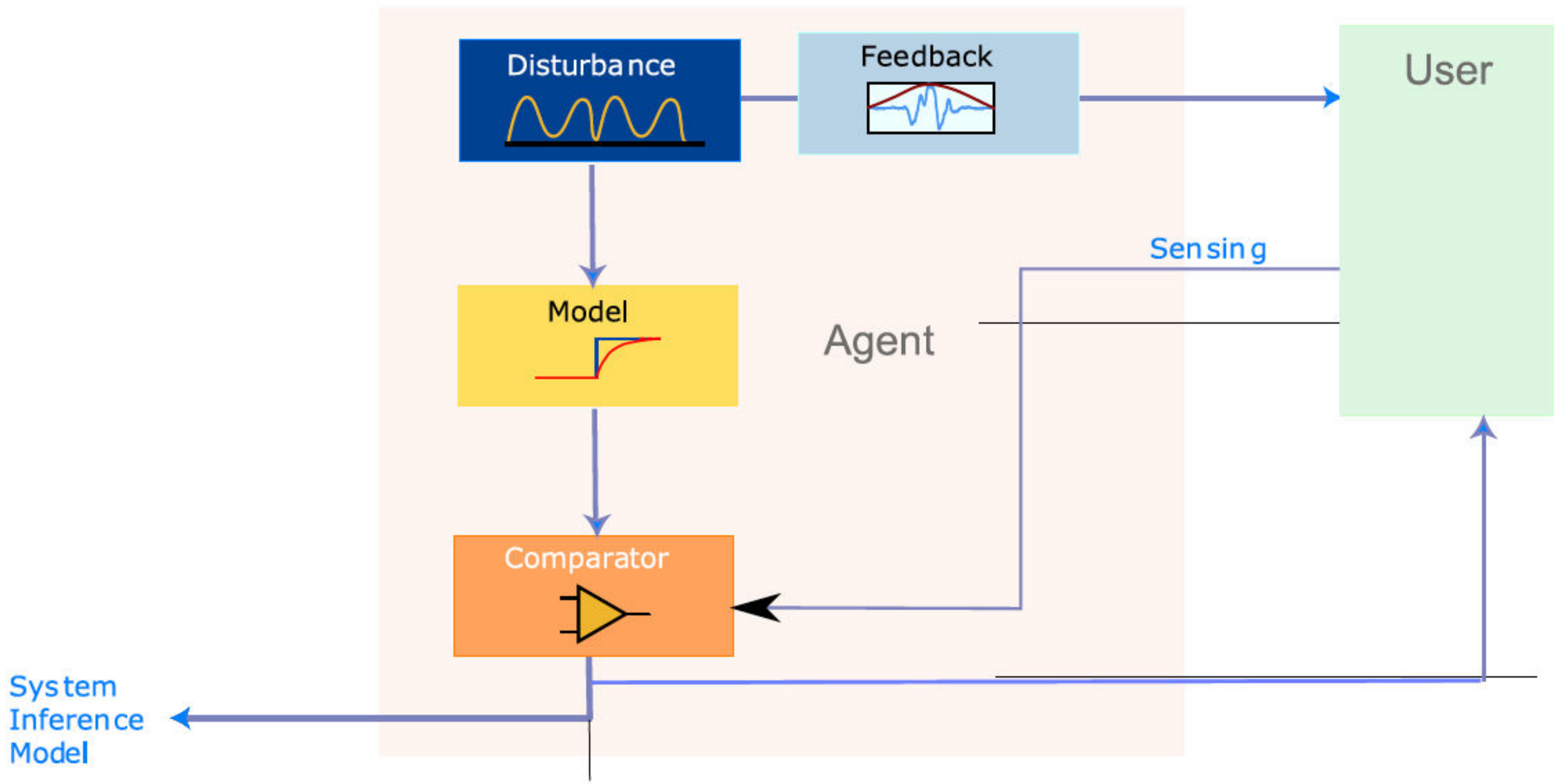
Perceptual Control theory

- The basic hypothesis of the work is that behaviour is the control of perception . In Powers' own words (Powers [1973a]):
  - The purpose of any given behaviour is to prevent controlled perceptions from changing away from the reference condition.
- One consequence of this view is that many activities can be seen as disturbance rejection , where a perceived variable is being held at some reference value by a controller. Actions are performed to counteract any observed changes in this variable.
  - These are not just low-level, physically sensed measurements such as "intensity of light falling on the retina". They can be quite complex mental constructs – for example, "the position of my hand relative to yours".
- This view provided Powers and his colleagues with a powerful experimental technique for testing the theories they laid out.

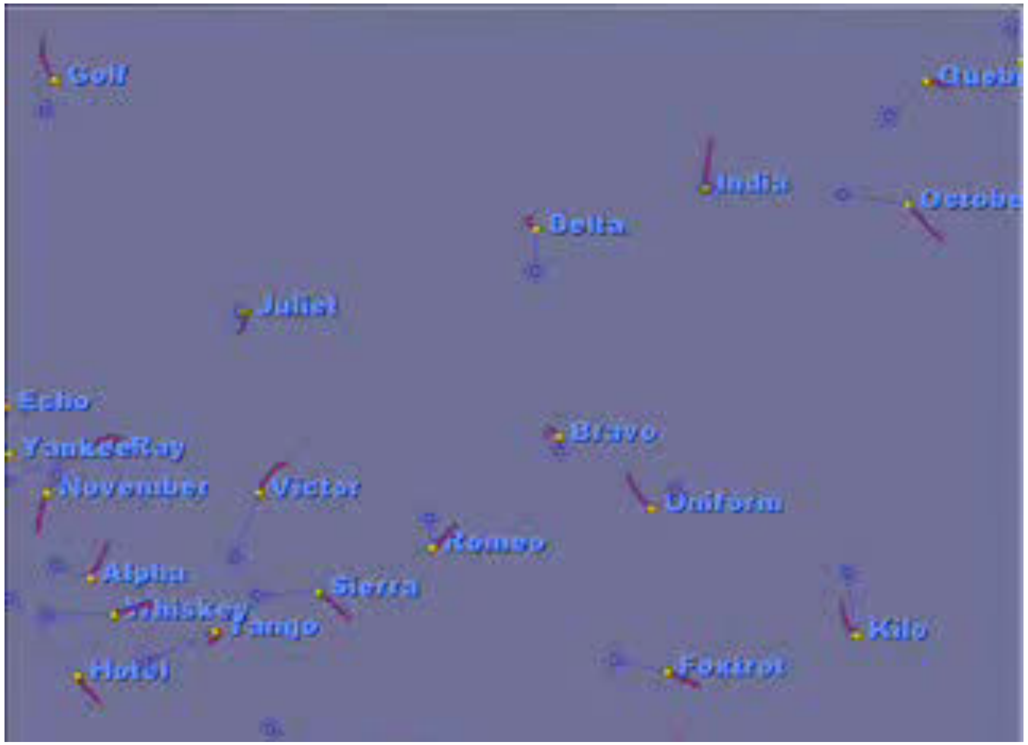
An Agent Perspective



- Reformulate selection in terms of **agents**
  - Each goal or item is considered an independent agent
- Agents probe user
  - "**experiment**", look for response
  - Evaluate probabilities p(*$selected_i$*)
  - Akin to MCMC Sampling from users mind!


Disturbance
Feedback
User
Sensing
Model
Agent
Comparator
System
Inference
Model

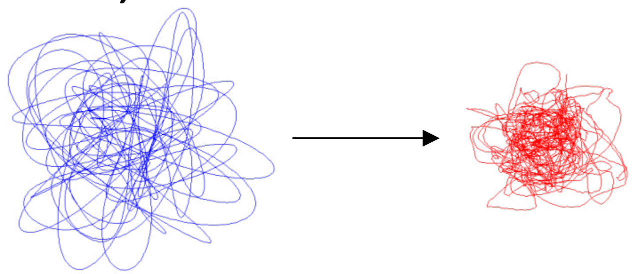
Demos


Golf
India
Delta
Juliet
Echo
YankeeRay
November
Victor
Bravo
Uniform
Romeo
Alpha
Sierra
Tango
Kilo
Hotel
Foxtrot

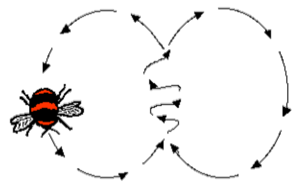

Log Song
TAP

Interpretations

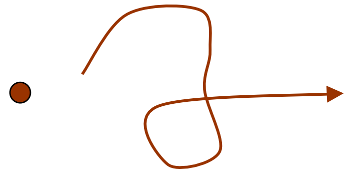
- Can be seen as control, damping, imitation, gesture recognition or excitation
  - Control/damping



  - Imitation of the motion of the disturbance (pursuit task)



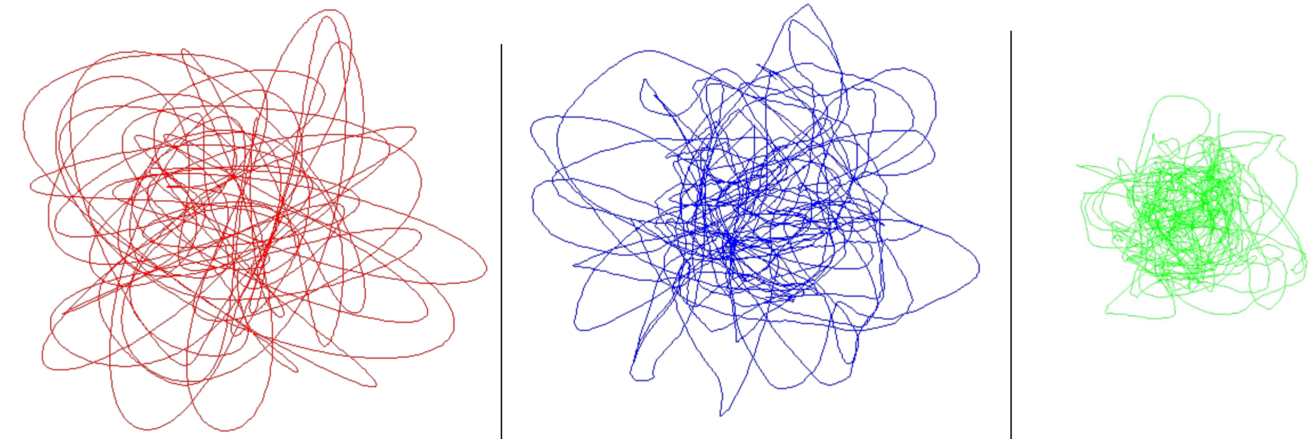
  - Gesture recognition with dynamically created gestures

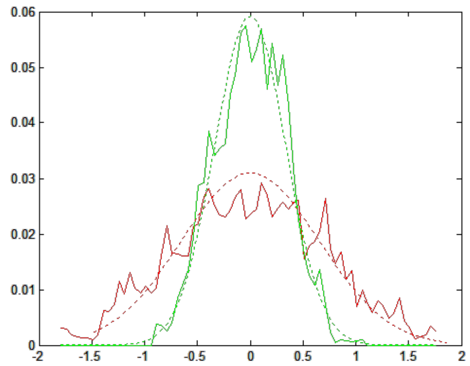


  - User excites “modes” of the object, inducing a meaningful disturbance in the object

Example: Movement

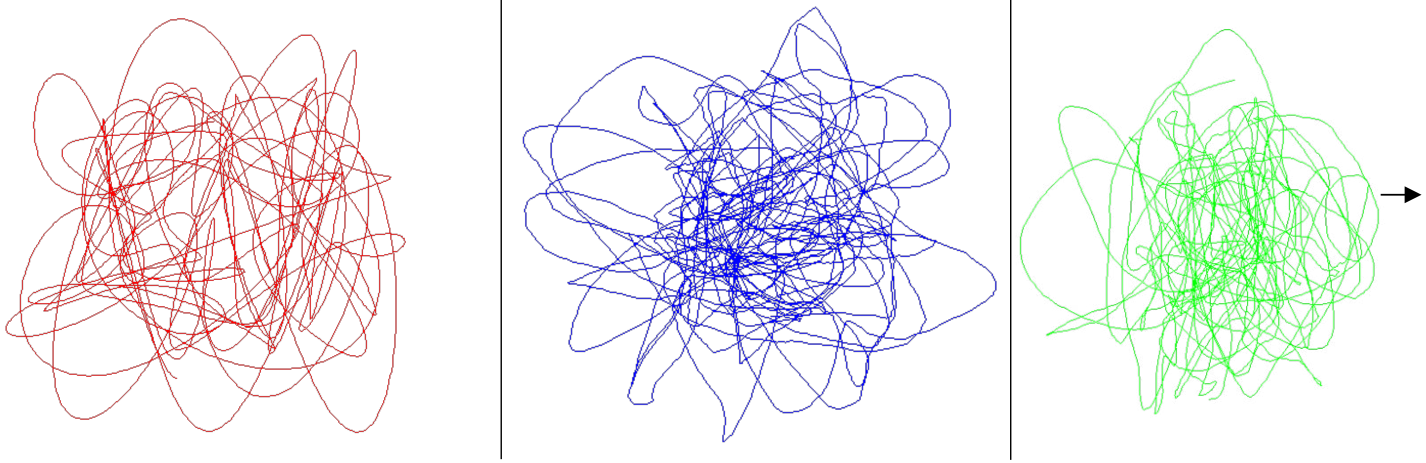
Test: Compare distribution of histories over some time window

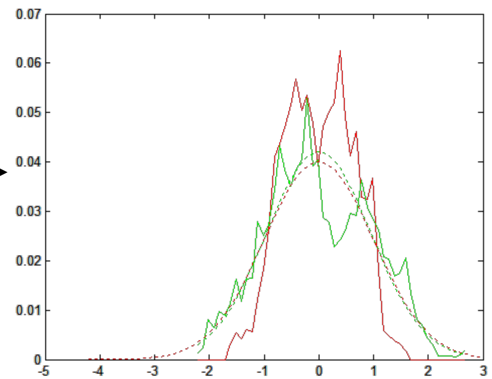




Agent Disturbance → User Control → Result

If controlled, $\dfrac{\sigma_e}{\sigma_a} > 1$





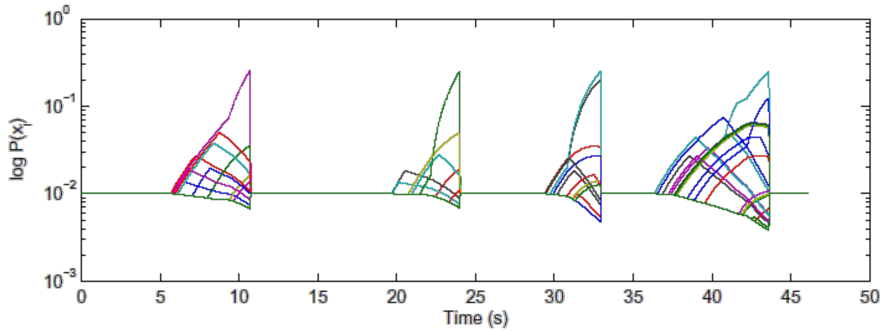
If no control, $\dfrac{\sigma_e}{\sigma_a} \approx 1$

Hence, we have: $p_i = f\left(\dfrac{\sigma_e}{\sigma_a}\right)$

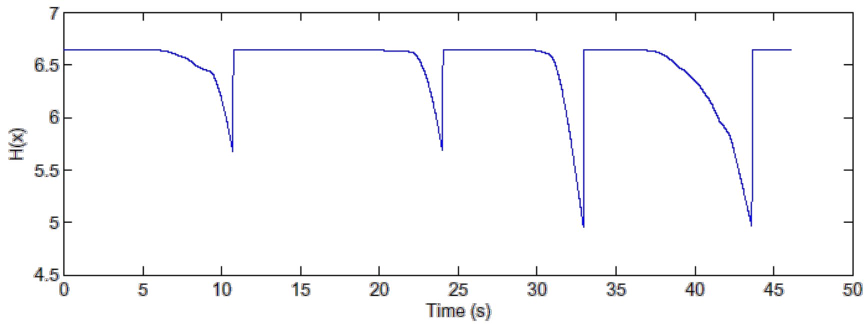
Feedback

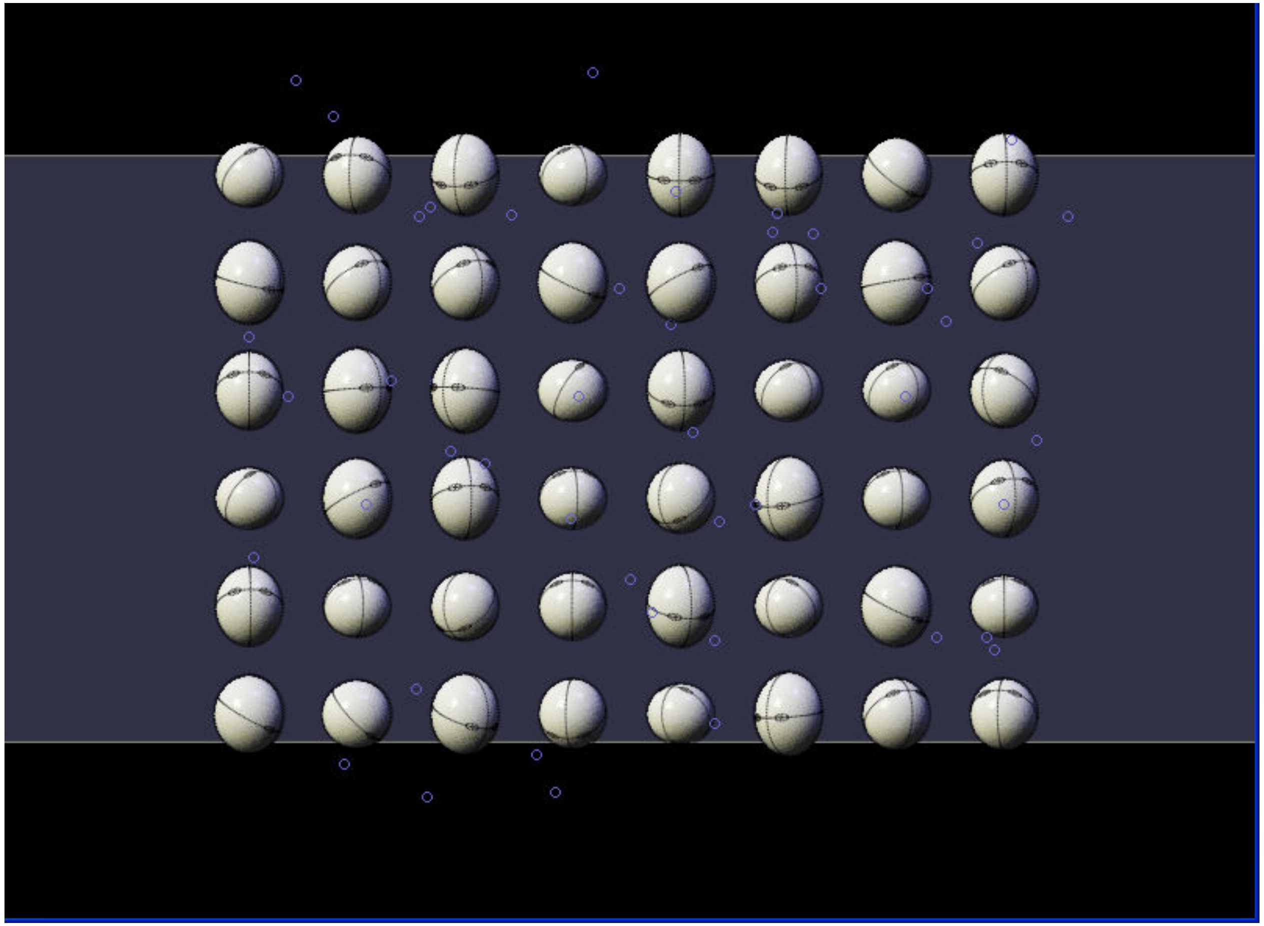
- Real-time feedback on potential goals
  - Visual example in demo
  - Mapping entropy to audio dissonance

Log P(*i*)



Entropy





Eggheads

- Make eye-contact with heads
- Experiment with ‘typical’ glances etc
- Interaction more timing related than smooth control.

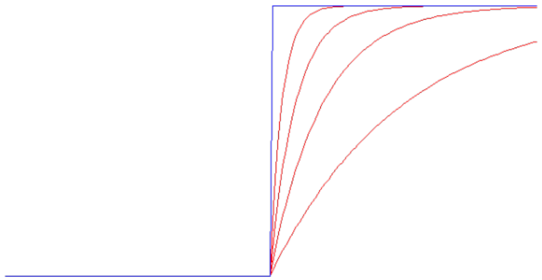
User Modelling

- So far: static behaviour models
  - Ignored dynamics of control
- Take into account models of human behaviour; e.g. from manual control theory
  - First and second order lags are reasonable approximations of human behaviour for tracking tasks
- E.g. first order lag:

$$Y = A - Ae^{-kt}$$

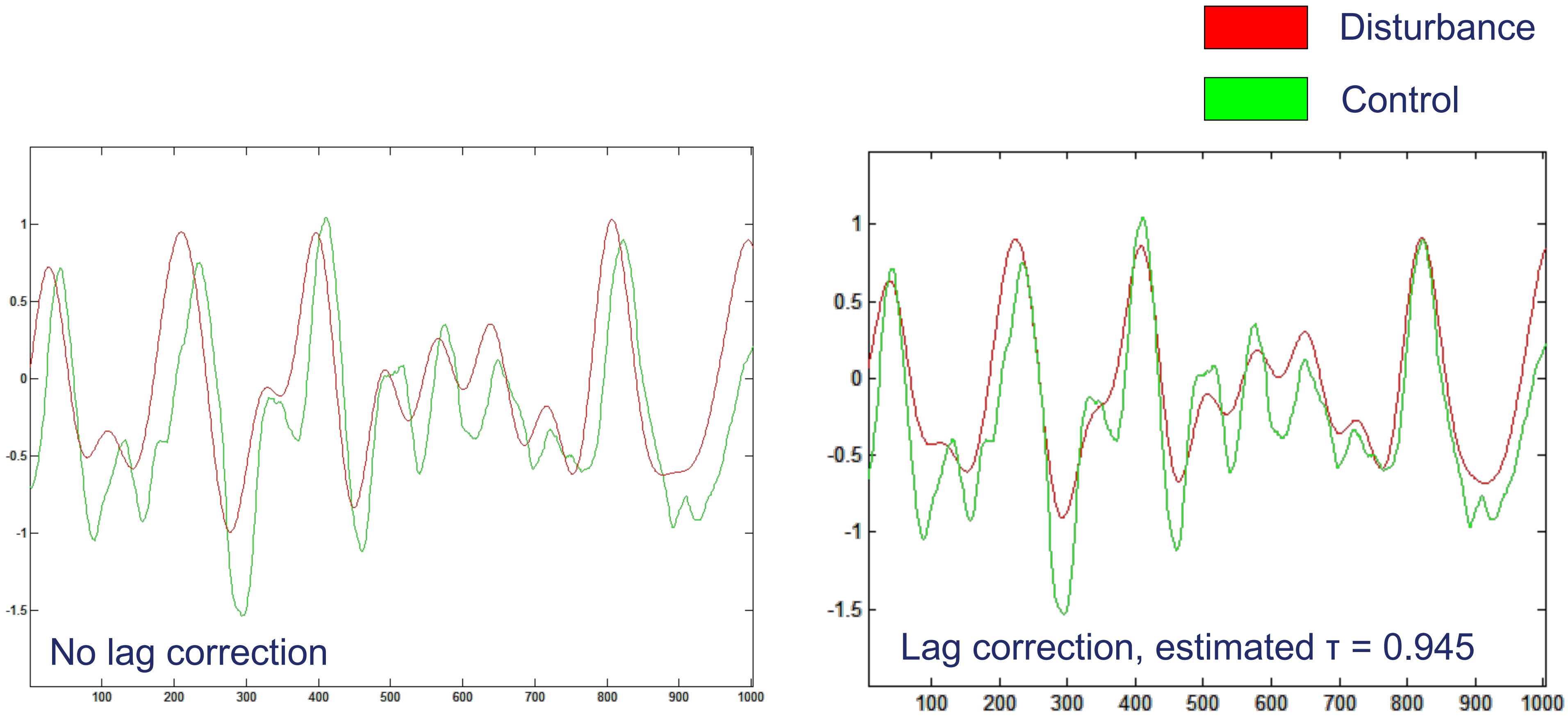
or

$$Yn = \tau Y + (1- \tau)X$$



First order lag: Step response

First-Order Lag Analysis



SSE = 1036.4

SSE = 424.9

University of Glasgow

QUANTIC

Inferring Hand pose with Deep Convolutional Autoencoders

Daniel Buschek, Roderick Murray-Smith,
Anna Feit, Antti Oulasvirta

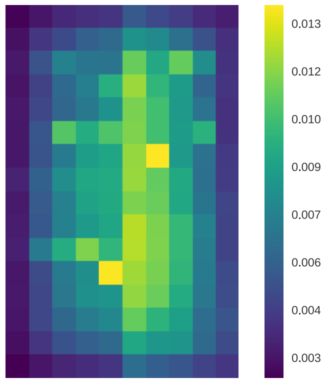
School of Computing Science,
University of Glasgow
Roderick.Murray-Smith@glasgow.ac.uk
http://www.dcs.gla.ac.uk/~rod
http://www.dcs.gla.ac.uk/~rod/Videos.html

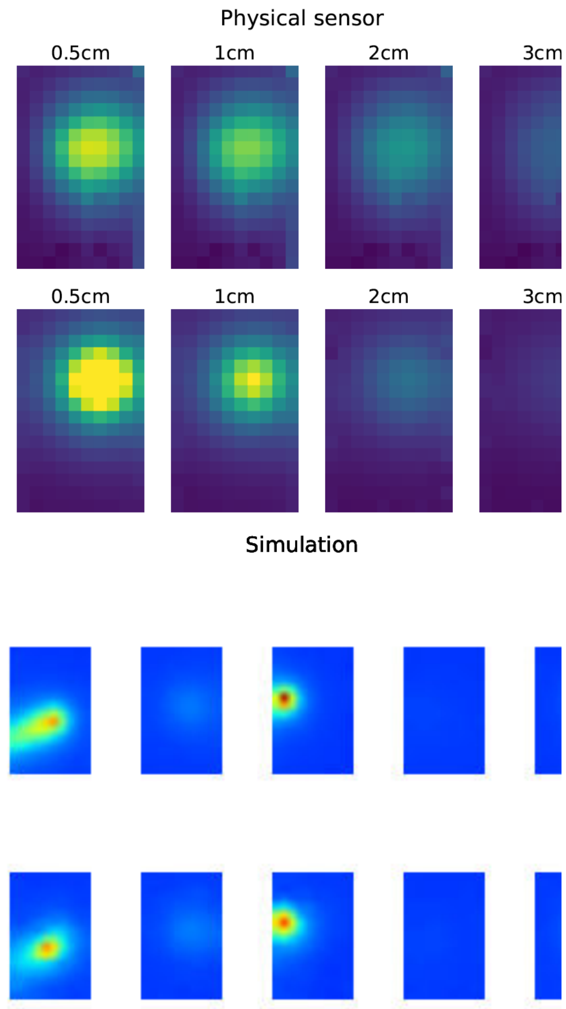
sicsa* The Scottish Informatics & Computer Science Alliance

60 YEARS OF COMPUTING AT GLASGOW

Why do input interfaces work the way they do?

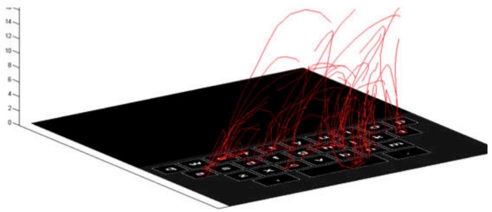
- E.g. prototype transparent touch screen - capacitive sensor with a depth range of between

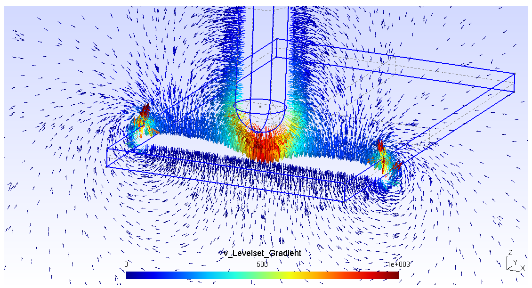




creen

s with

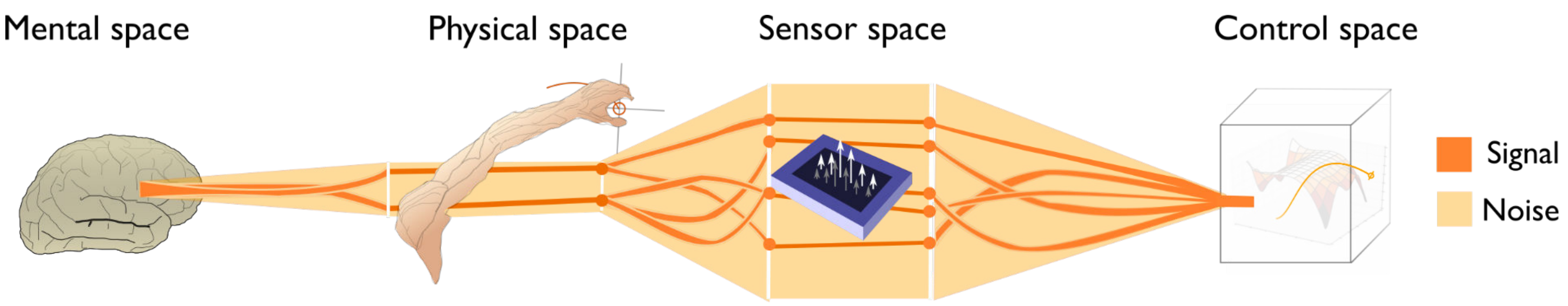




-

Engineering input devices

(slides stolen from John Williamson!)

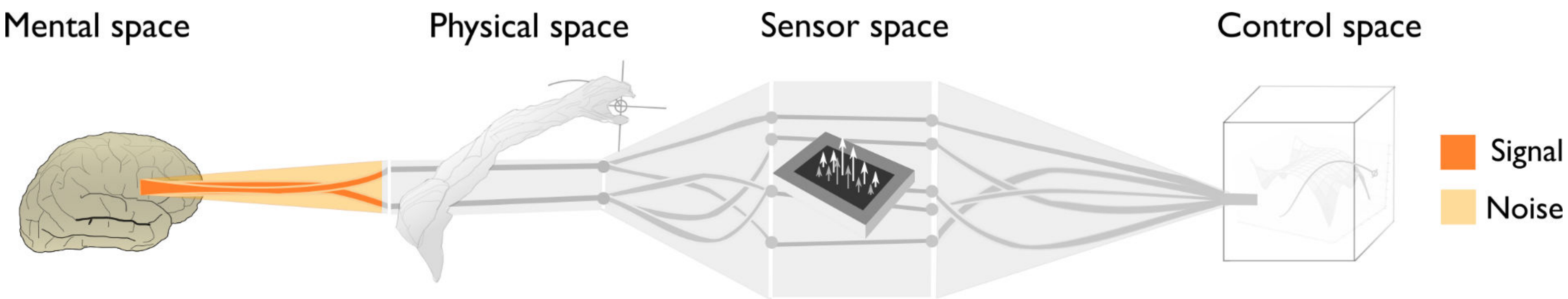


Objectives

Faster Reduce time and ingenuity needed to make sensors into input devices.
Better Increase expressivity and efficiency of input devices using novel sensors.
Tailored Precisely match interfaces to user needs and sensor capabilities.

Engineering input devices

Intention starts simple.



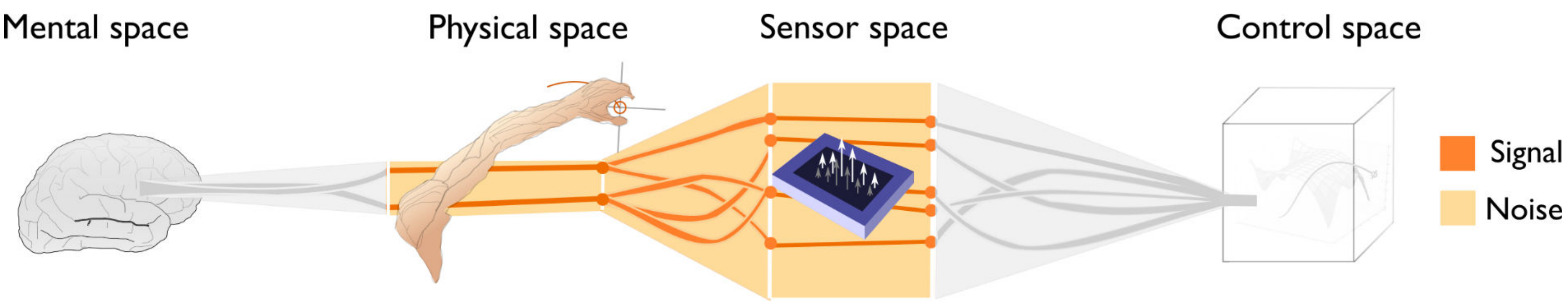
Objectives

Faster Reduce time and ingenuity needed to make sensors into input devices.
Better Increase expressivity and efficiency of input devices using novel sensors.
Tailored Precisely match interfaces to user needs and sensor capabilities.

Engineering input devices

Intention starts simple.

Gets tangled and distorted
into noisy, correlated signals.



Objectives

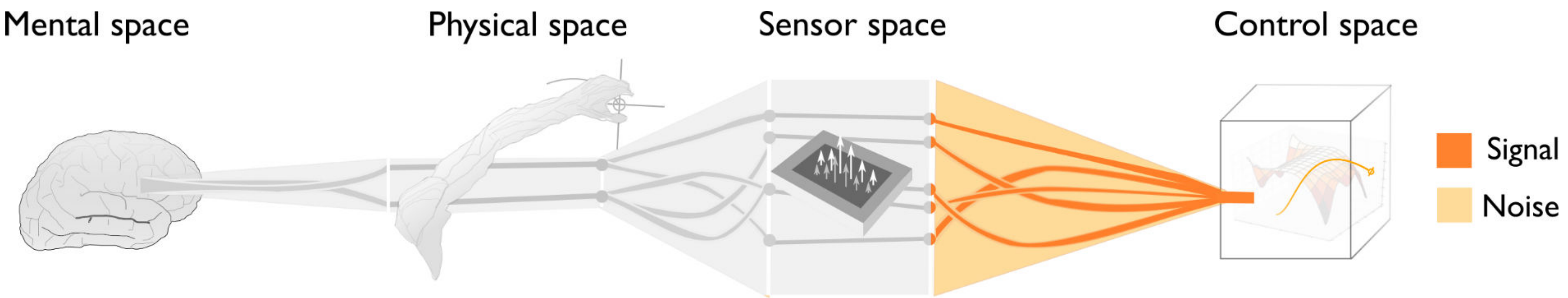
Faster Reduce time and ingenuity needed to make sensors into input devices.
Better Increase expressivity and efficiency of input devices using novel sensors.
Tailored Precisely match interfaces to user needs and sensor capabilities.

Engineering input devices

Intention starts simple.

Gets tangled and distorted
into noisy, correlated signals.

Untangle intention by learning
meaningful variations from data.



Objectives

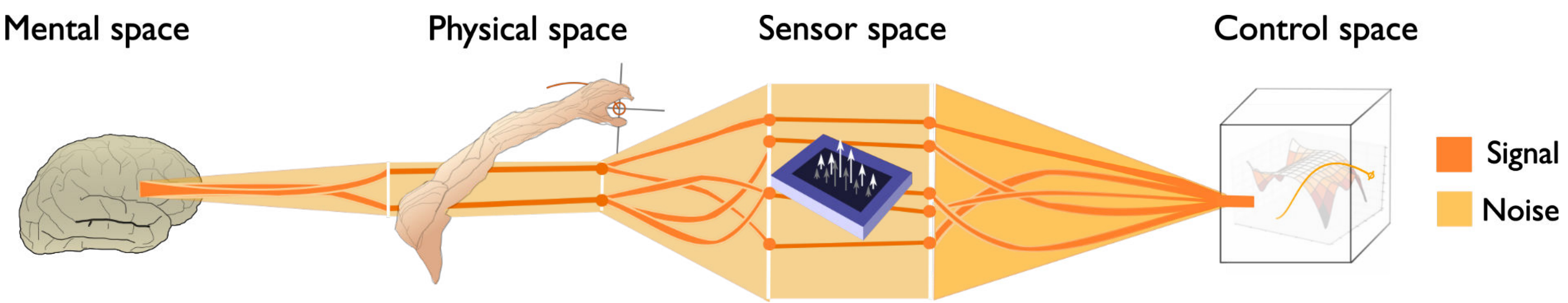
Faster Reduce time and ingenuity needed to make sensors into input devices.
Better Increase expressivity and efficiency of input devices using novel sensors.
Tailored Precisely match interfaces to user needs and sensor capabilities.

Engineering input devices

Intention starts simple.

Gets tangled and distorted
into noisy, correlated signals.

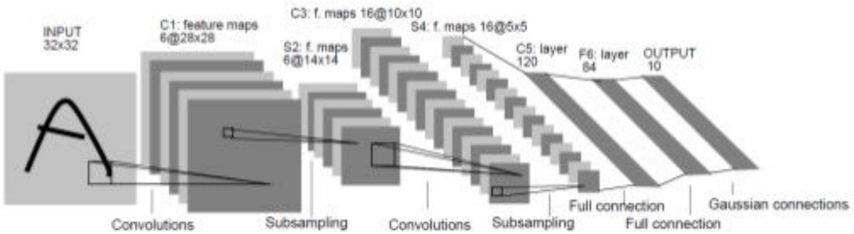
Untangle intention by learning
meaningful variations from data.



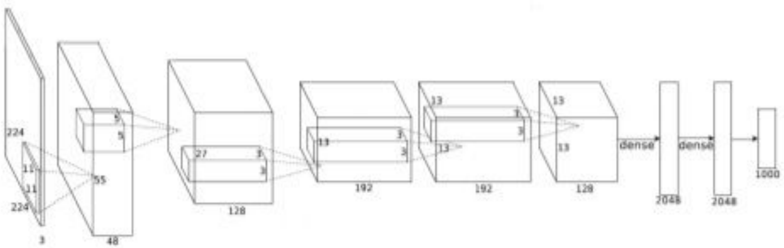
Objectives

Faster Reduce time and ingenuity needed to make sensors into input devices.
Better Increase expressivity and efficiency of input devices using novel sensors.
Tailored Precisely match interfaces to user needs and sensor capabilities.

Deep Convolutional networks



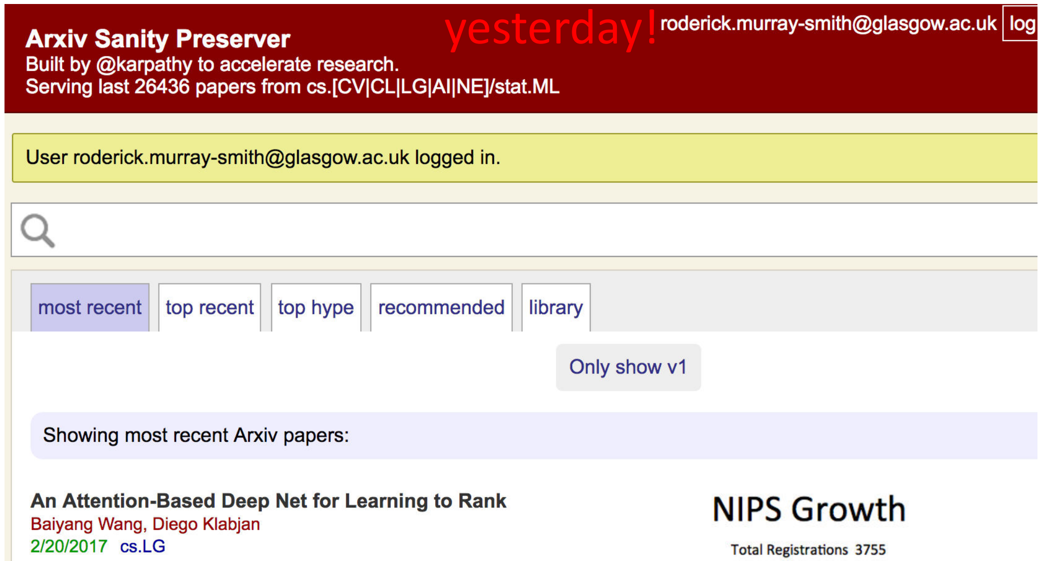
Gradient-Based Learning Applied to Document Recognition, LeCun, Bottou, Bengio and Haffner, Proc. of the IEEE, **1998**

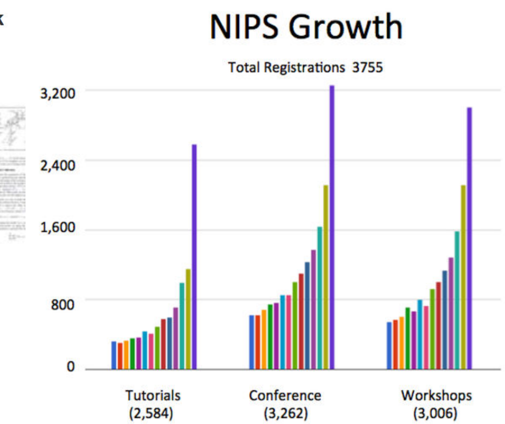


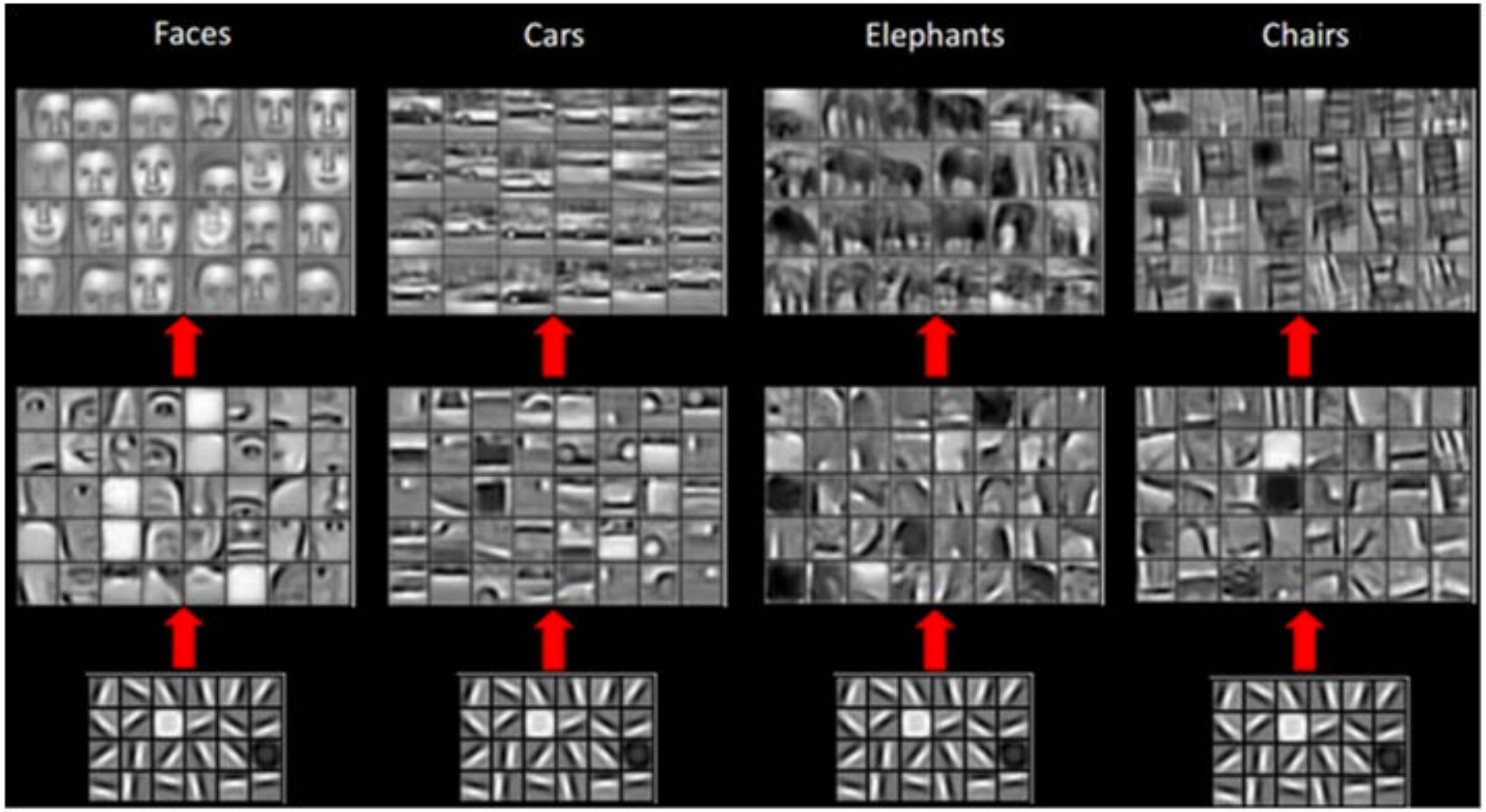
Imagenet Classification with Deep Convolutional Neural Networks, Krizhevsky, Sutskever, and Hinton, NIPS **2012**

Slide Credit: L. Zitnick

40 new papers published yesterday!






Faces
Cars
Elephants
Chairs

Autoencoder:
Learning a compressed representation

Image (28x28 pixels)

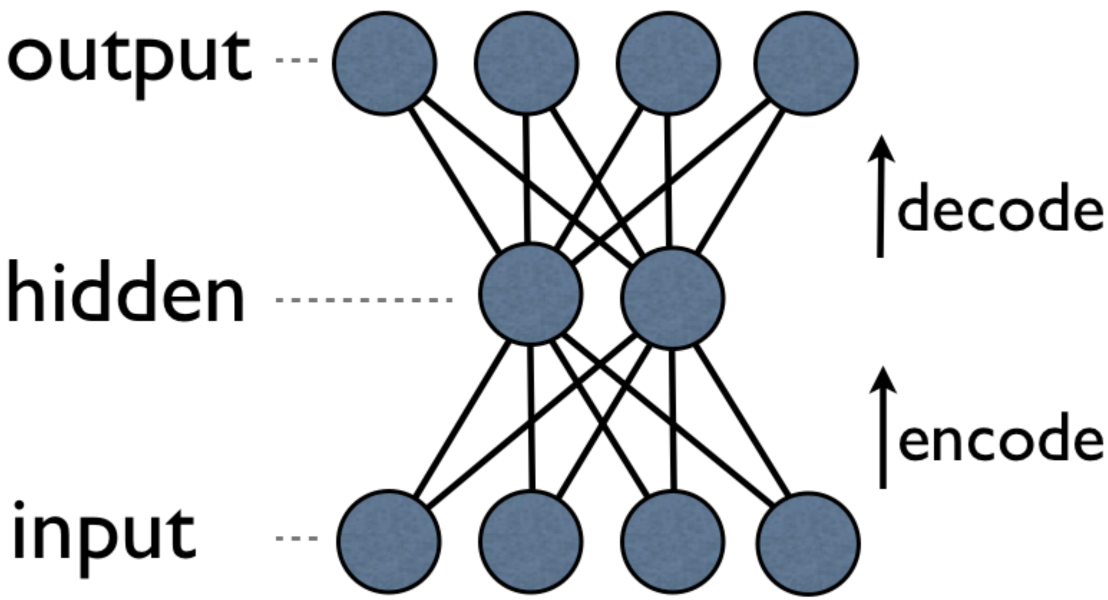
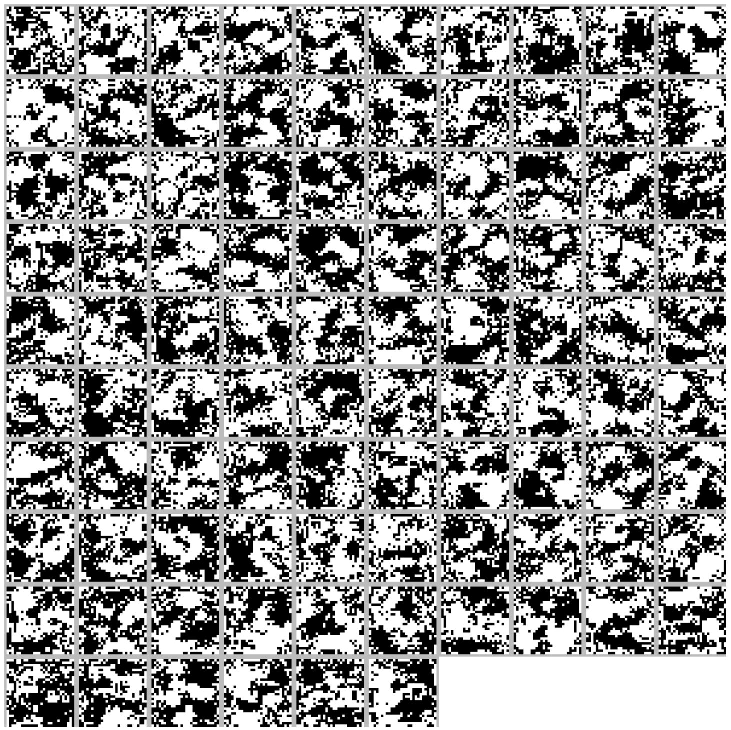


Image (28x28 pixels)

Optimised mirror bases



MNIST handwriting example

Inputs (28x28 pixels)

Reconstructed from 96 bases

How: Unsupervised machine learning

Unsupervised learning

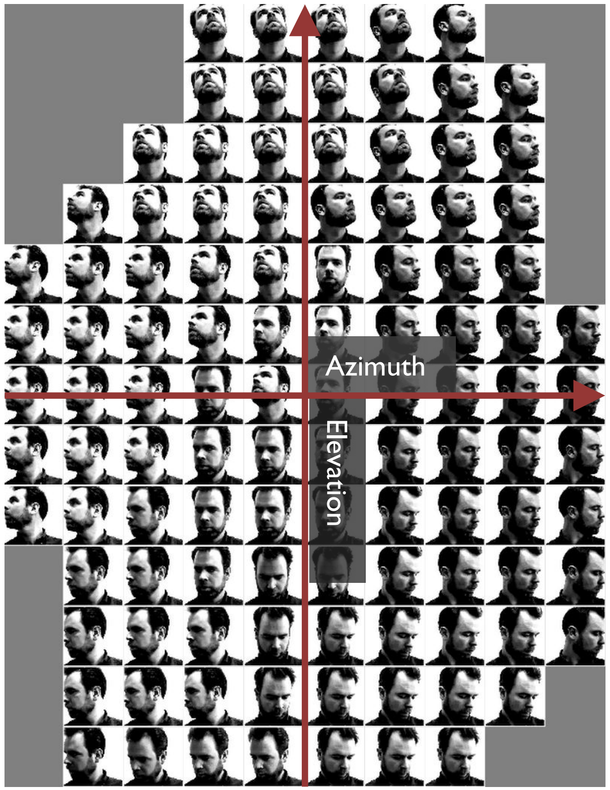
Structure from data

Take (apparently) complex data with many dimensions. Reduce to a simple low-dimensional **manifold.**

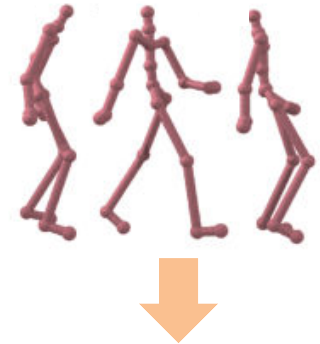
Left: Images (4096 dimensions) where semantically **meaningful** dimensions have been discovered from **raw pixels.**
Could extract **2D pointing control** directly from unsorted photographs of heads in different poses.

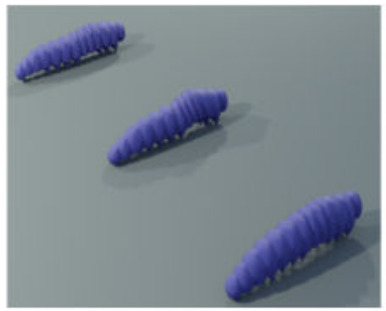
Right: Unsupervised learning finding meaningful, smooth structure in **human motion** capture data (>100 dimensions) to remap actor motions to expressive animation of nonhumanoid characters.

Orientation manifold



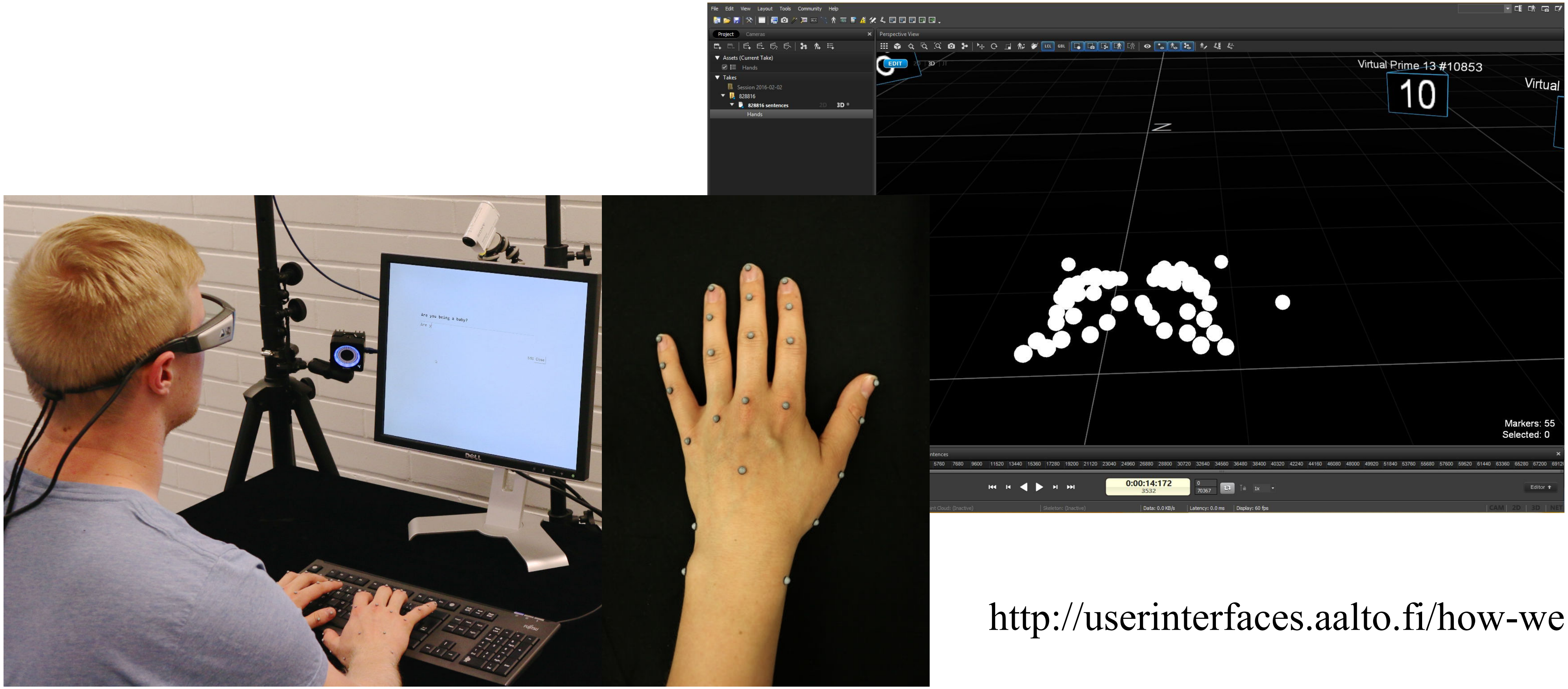
Human motion



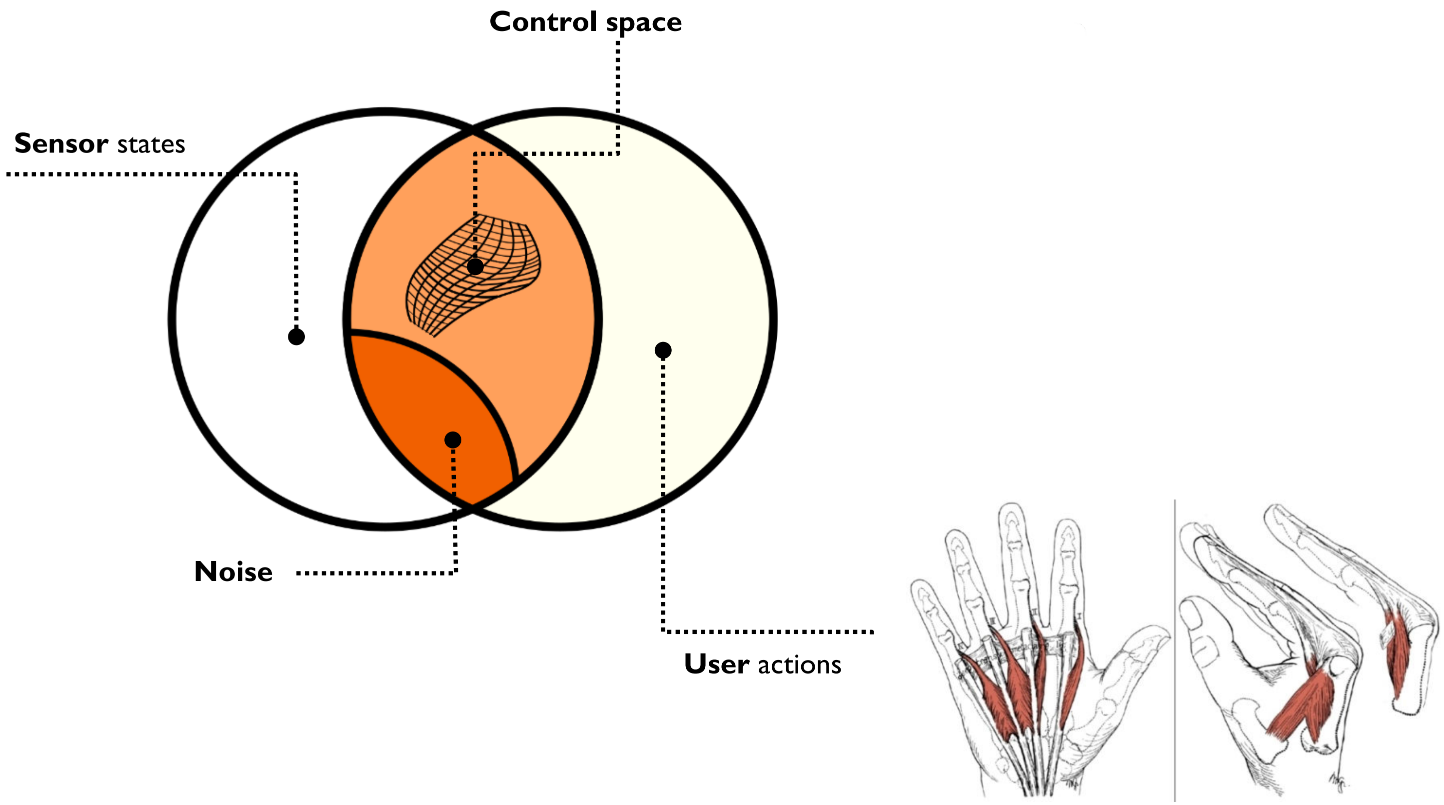


"**Interactive motion mapping for real-time character control**" (2014), Rhodin, H., et al., Computer Graphics Forum.

Hand motion capture -



http://userinterfaces.aalto.fi/how-we


Control space
Sensor states
Noise
User actions