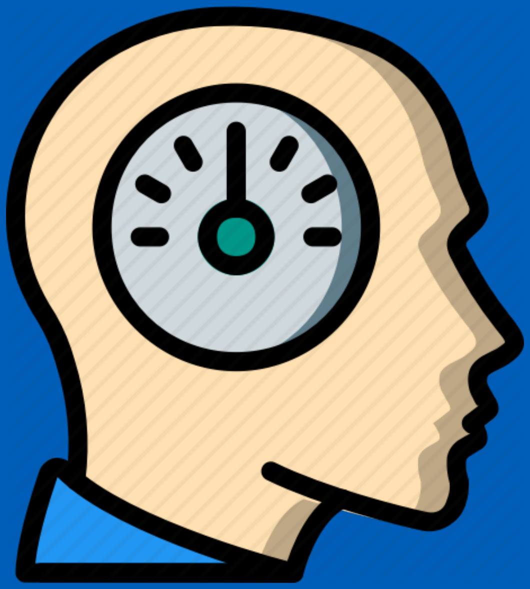
Aalto University

ELEC-D7010 Engineering for Humans



Lecture 1: Human Performance

April 27, 2021
Antti Oulasvirta
Aalto University
userinterfaces.aalto.fi

Empirical research on human performance

Models of human performance

<u>What</u>: Link human- and design/task variables mathematically

<u>Why</u>: Accurate and practical models to inform design

Statistical methods used for

- **Model construction**
- **Model fitting**
- **Model validation**
- **Model selection**

Engineering models of human performance try to find the best trade-off between model-complexity and predictive validity

Learning objectives in this lecture

1. Response process models
Human performance in discrete input tasks, including aiming and choice

- Fitts' law
- Hick-Hyman law

2. Task performance models
Decomposition of task performance into motor-cognitive actions

- KLM

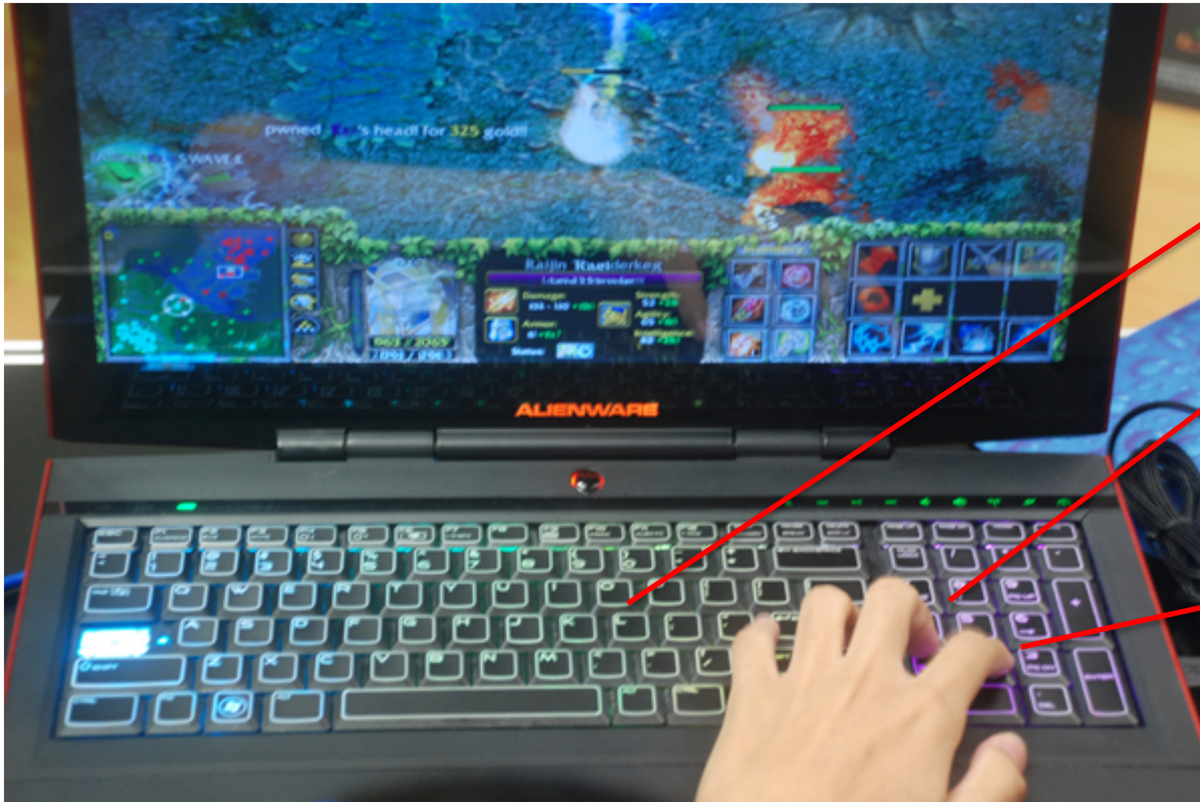
Aalto University

Introduction

Response processes

Overview of response processes

Terminology: Response set; a transducer; feedback



Aiming

Choice reaction

Simple reaction

Response process models are defined by *movement demands*

Spatial demand

Temporal demand

	'As accurately as possible'	**Distance**	**Width**
'As quickly as possible'	Simple reaction / Choice reaction	Selecting a point target	Selecting a button target
Distance	Synchronization		
Width	Temporal pointing		Interception

Why is this topic important?

Basic capabilities and limitations of humans in interaction

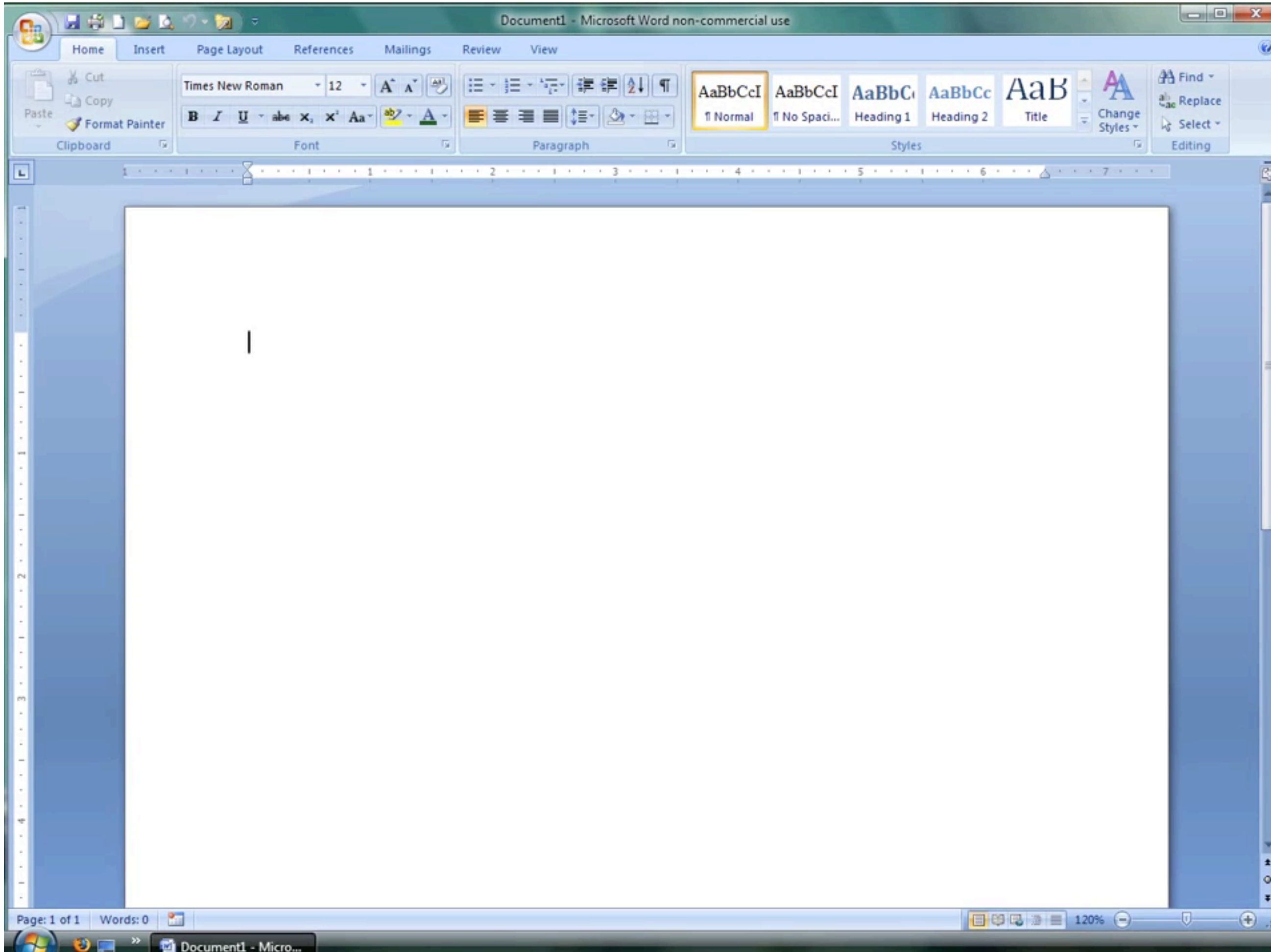
The "atoms of interaction": Sensorimotor responses underpin almost all interaction with user interfaces

Models allow you to find optimal tradeoffs among design decisions

You can exploit them computationally in the generation, refinement, and adaptation of user interfaces

Example: Subway Surf

Example: Microsoft Word



Patient monitoring

Q: What’s assumed in this design?

Definitions

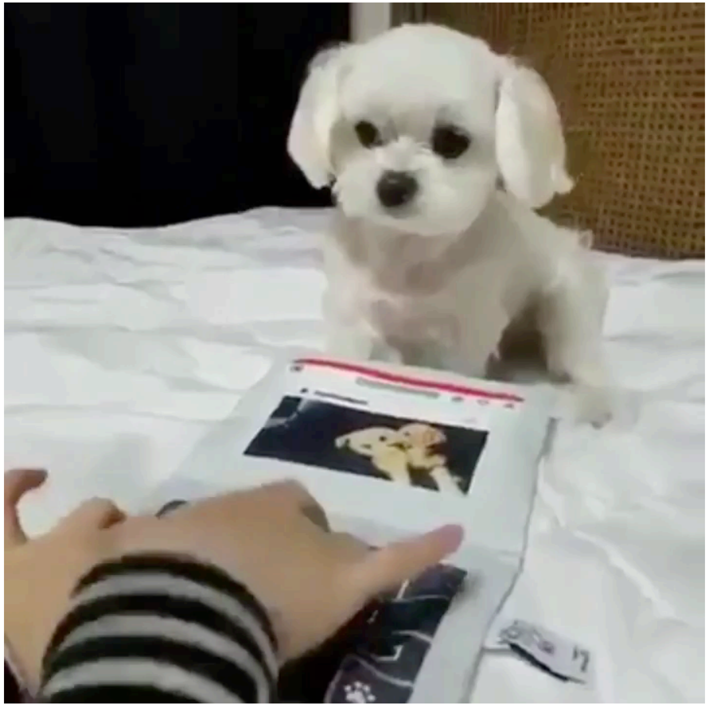
A response is action taken by user within a constrained set of options defined by the computer.

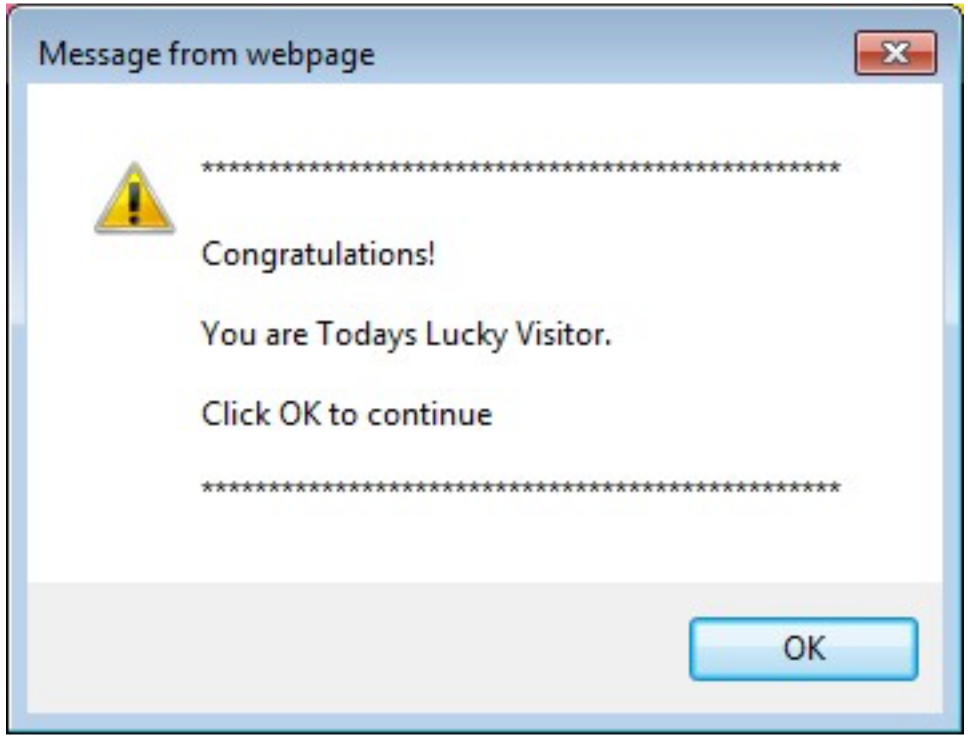
- **The computer is in some state with a limited set of options, which is transformed according to the user's input.**

A response process refers to the temporal events that take place during a response and affect performance (speed and accuracy)

- **As we learned, different models defined by 1) set size (number of options), 2) spatial and 3) temporal demands.**

Q: Why is this NOT a response?

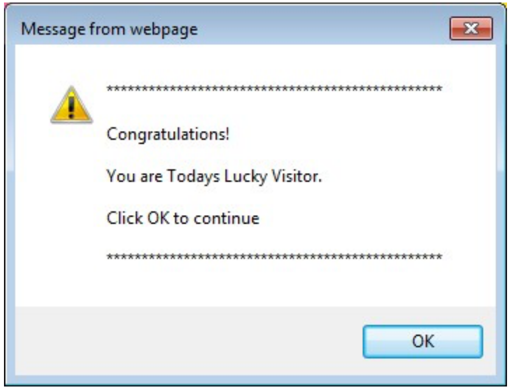



Message from webpage
Congratulations!
You are Todays Lucky Visitor.
Click OK to continue
OK

Response demands

"Response demands" characterize the requirements of the response that the user must give

1. A response set (e.g., "OK")
2. A transducing mechanism (e.g., keyboard)
3. Feedback (e.g., dialogue disappears)
4. Spatial objectives / constraints
5. Temporal objectives / constraints



Describe the response demands of emergency braking

1. The response set?
2. The transducing mechanism?
3. Temporal objective?
4. Feedback?

Response demands can be used to understand everyday tasks

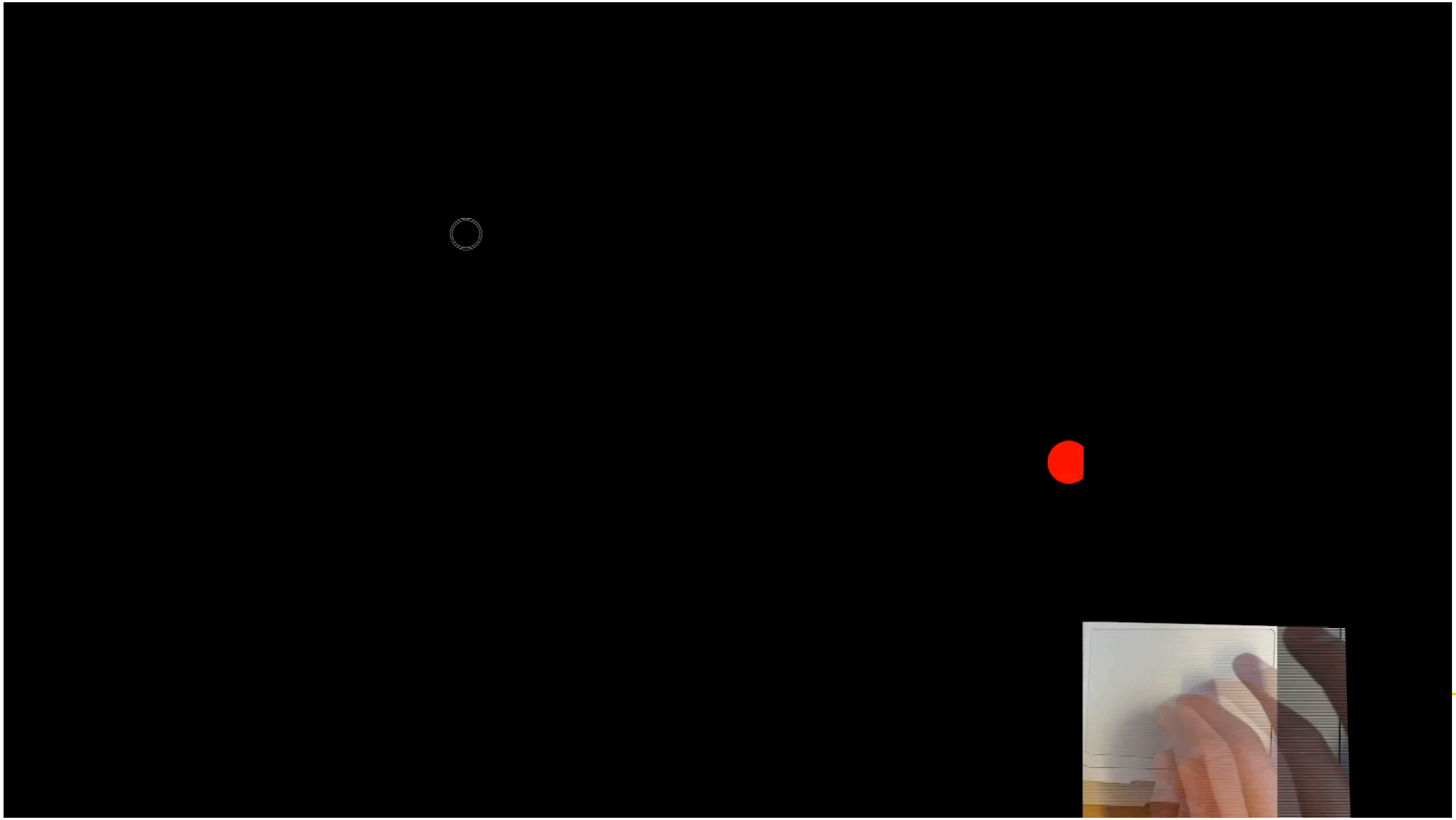
Emergency braking: Push the right pedal immediately

Calling an elevator: Hit the right button and get it activated; no hurry

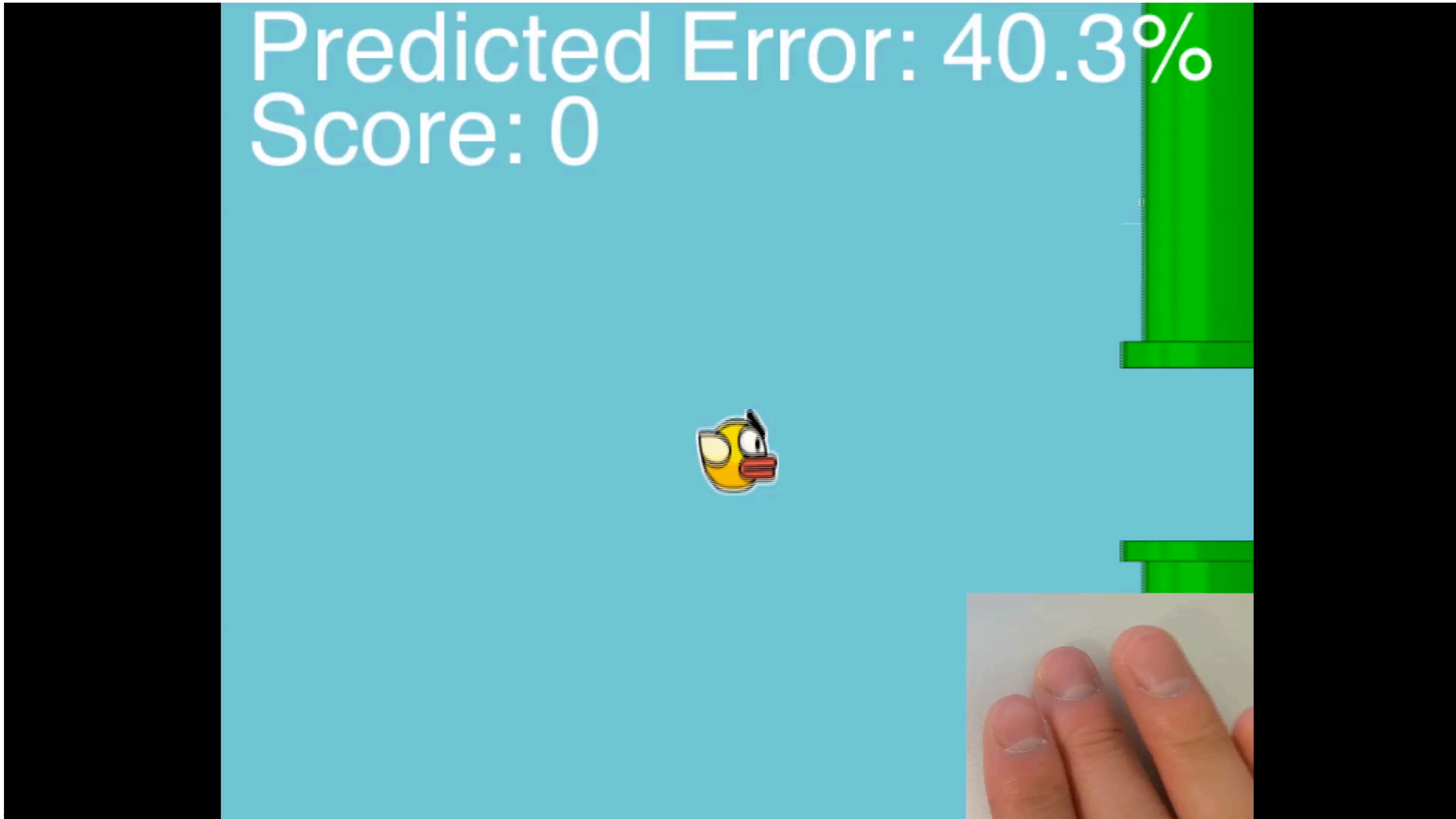
Choosing an item to buy in Amazon: Select the correct one item, but there's no hurry

…

Spatial demands



Temporal demands



Back to our axonomy of response processes

Spatial demand

Temporal demand

	Practically no requirement	Distance	Width
'As quickly as possible'	Simple reaction / Choice reaction	Selecting a point target	Selecting a button target
Distance	Synchronization		
Width	Temporal pointing		Interception

Lee & Oulasvirta CHI'16

Q: Which response type?

Interception

A spatially and temporally bound target

Empirical factors affecting response performance

Distractors

Preview time

Size of response set

Input device

Feedback

Learning Objectives for response process models

Recognize the right response process in a given HCI task

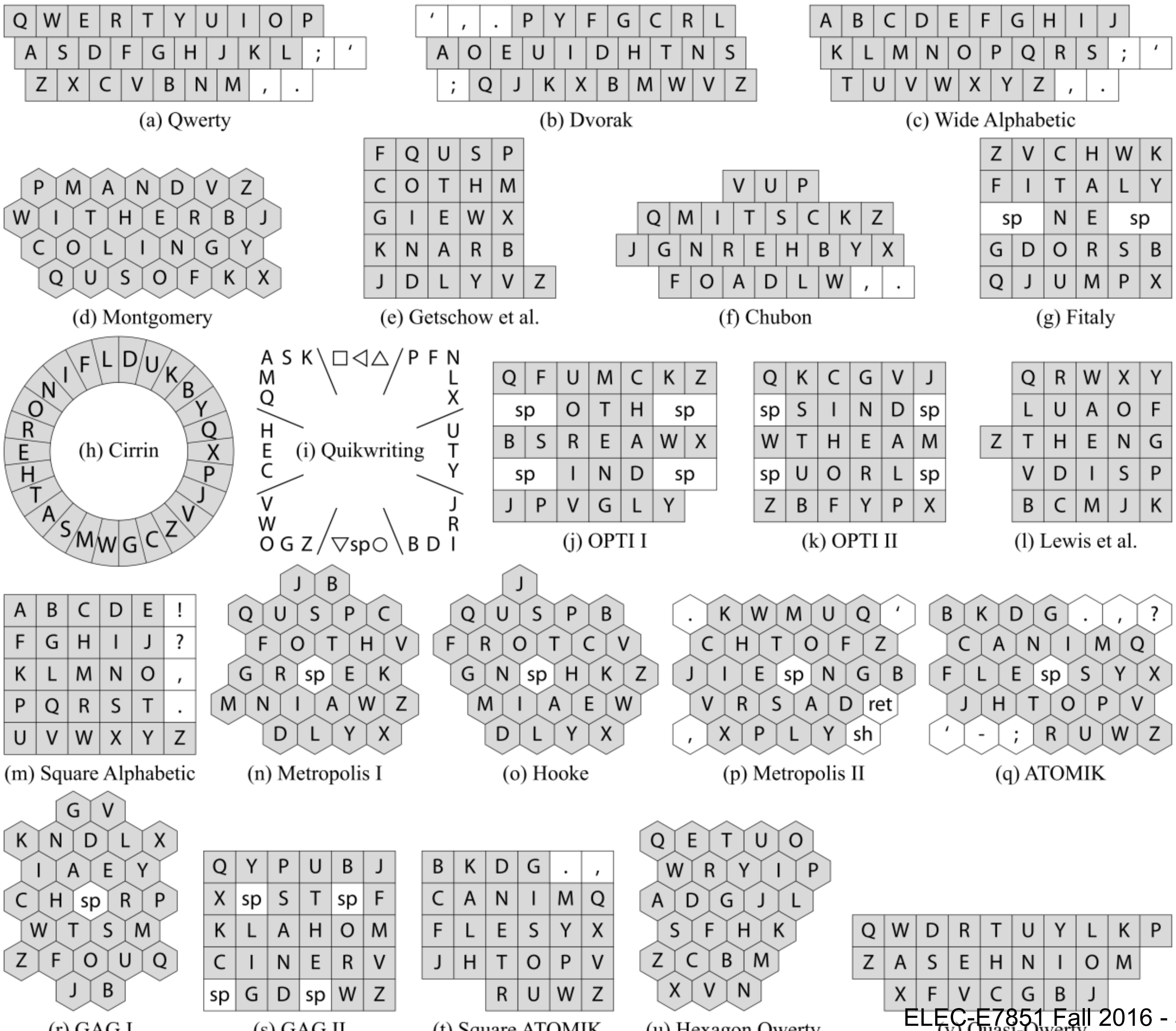
Know the basic models (Fitts' law and Hick's law) and understand their position among RP models

Analyze trade-offs using appropriate model mathematically

Use models to enhance designs

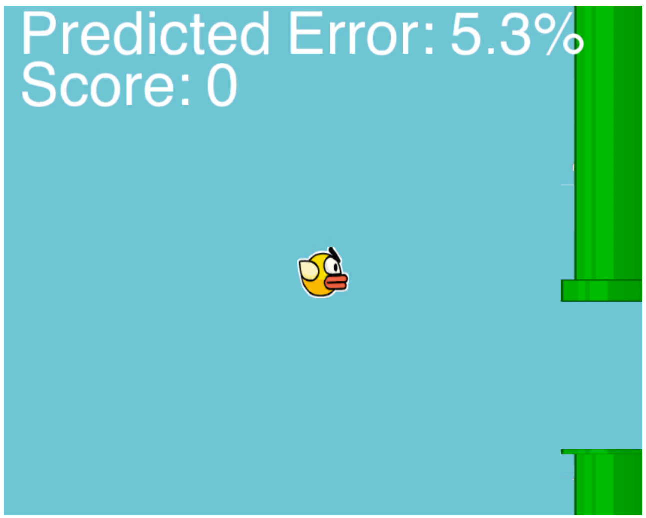
Improve layouts

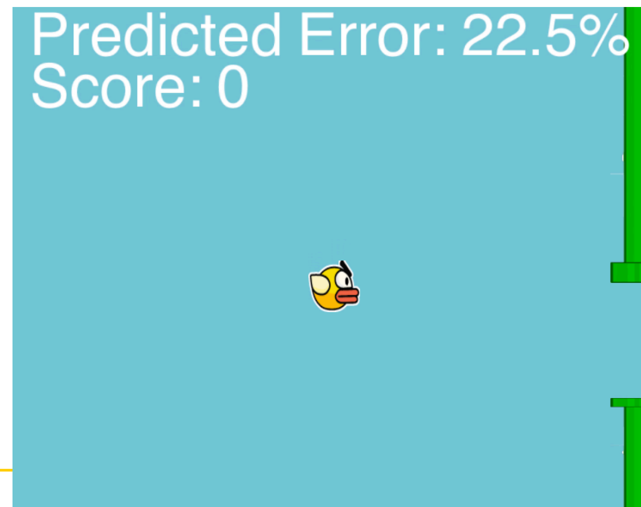
Figure 1. Pareto optimized Arabic keyboard layout

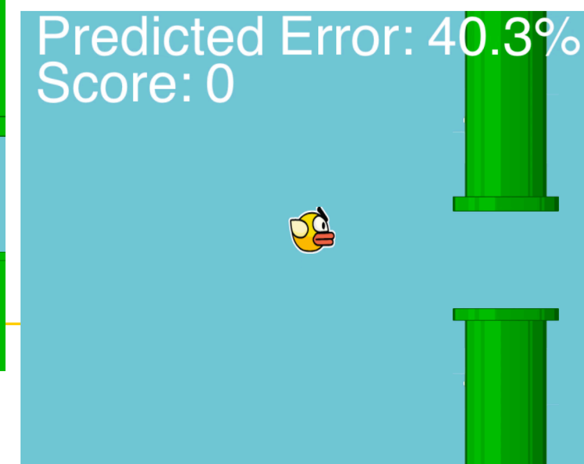

(a) Qwerty
(b) Dvorak
(c) Wide Alphabetic
(d) Montgomery
(e) Getschow et al.
(f) Chubon
(g) Fitaly
(h) Cirrin
(i) Quikwriting
(j) OPTI I
(k) OPTI II
(l) Lewis et al.
(m) Square Alphabetic
(n) Metropolis I
(o) Hooke
(p) Metropolis II
(q) ATOMIK
(r) GAG I
(s) GAG II
(t) Square ATOMIK
(u) Hexagon Qwerty
(v) Quasi-Qwerty

Control level of difficulty in user responses

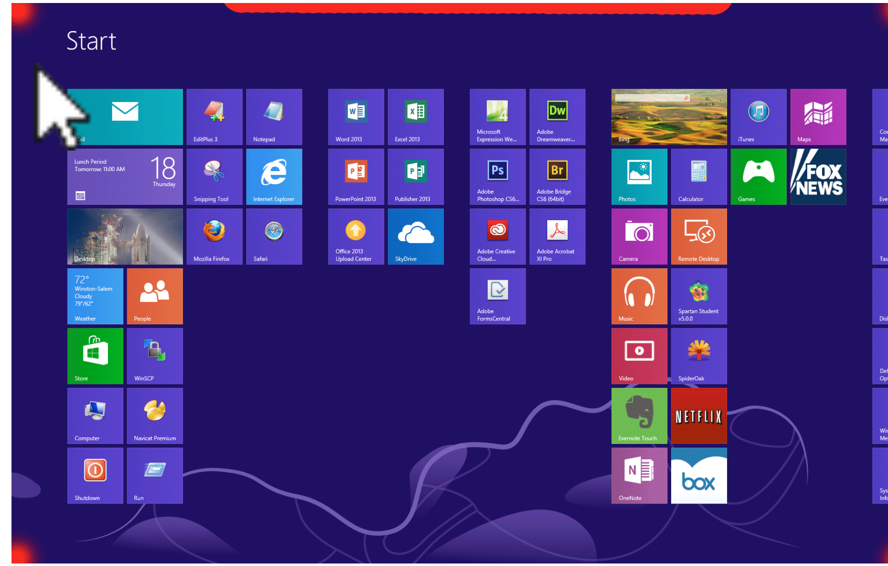
Example: Increasing temporal pointing demand to control the probability of game character dying







Sum up: Movement demands in discrete input



Performance is affected by

1. **Spatial constraints**
   - Size of target
   - Movement distance
2. **Temporal constraints**
   - Onset and offset of target in time
3. **Number of distractors (non-targets)**

Goals today:
Fitts' law
Hick-Hyman law

Response Process Models

From simple reaction to aimed movement

Taxonomy of response processes

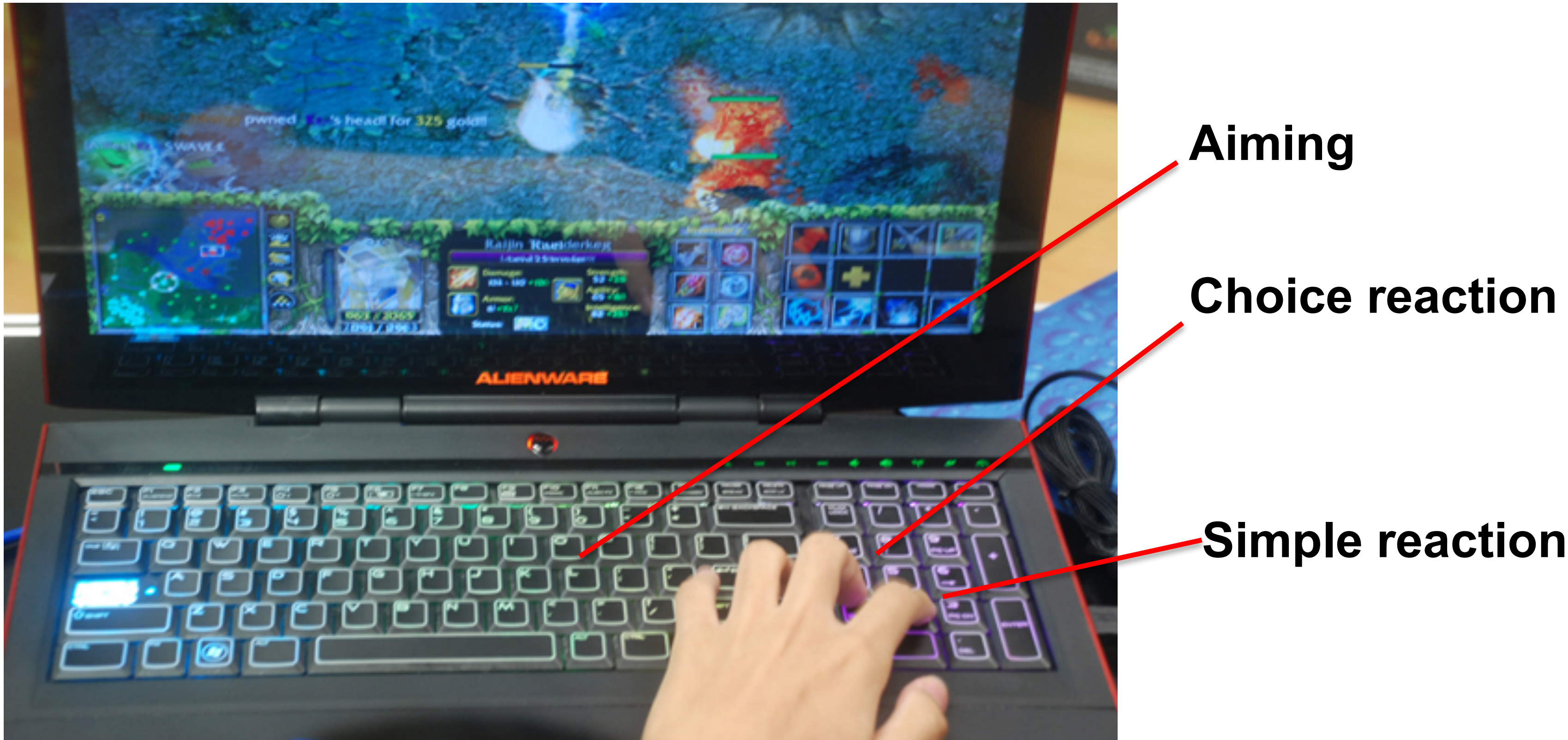
Spatial demand

Temporal demand

	Practically no requirement	Distance	Width
'As quickly as possible'	Simple reaction / Choice reaction	Selecting a point target	Selecting a button target
Distance	Synchronization		
Width	Temporal pointing		Interception

Lee & Oulasvirta CHI'16

Overview



Overview

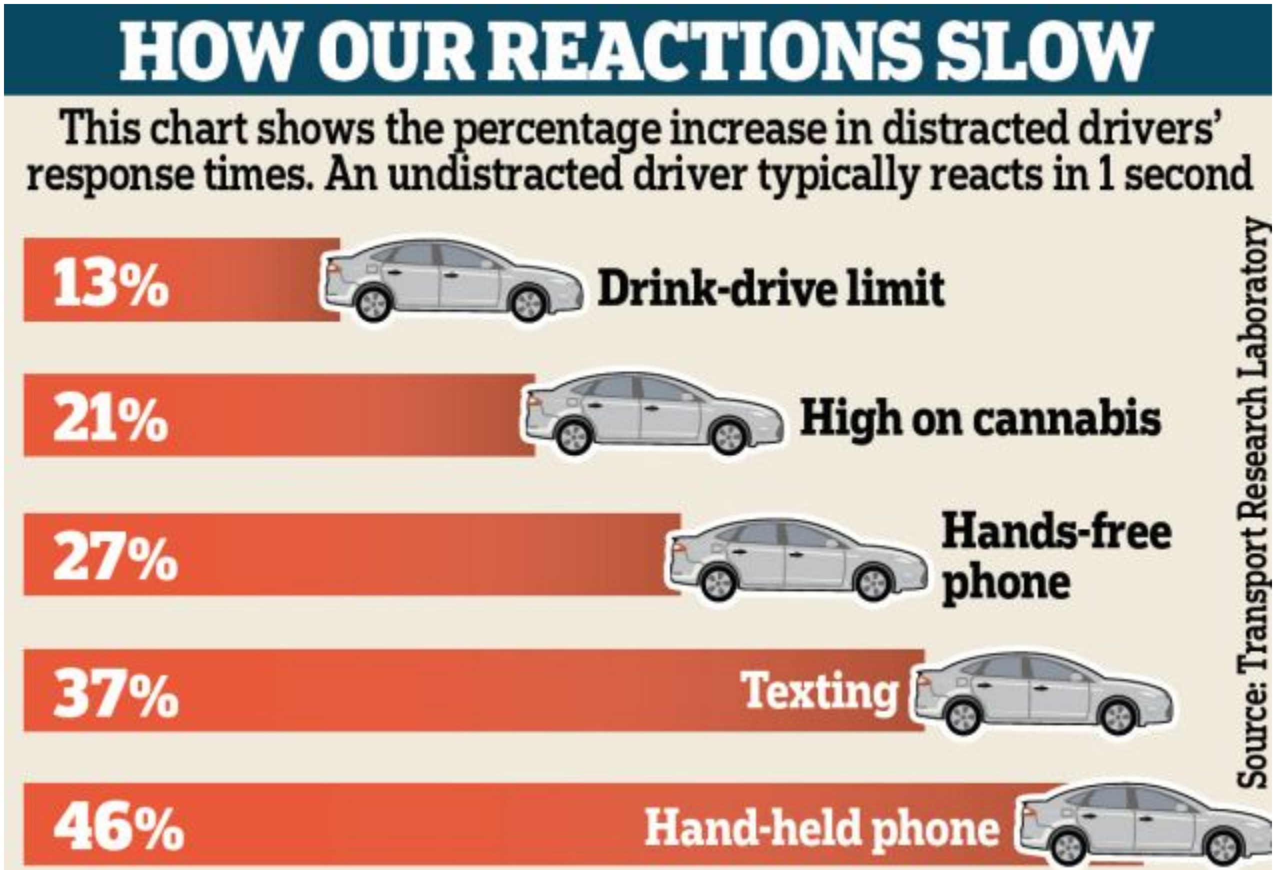
The mathematical formula will be given also in Assignments, we here focus on the main ideas

The models contain parameters that are task- and user-specific
→Empirically obtained or inferred from data

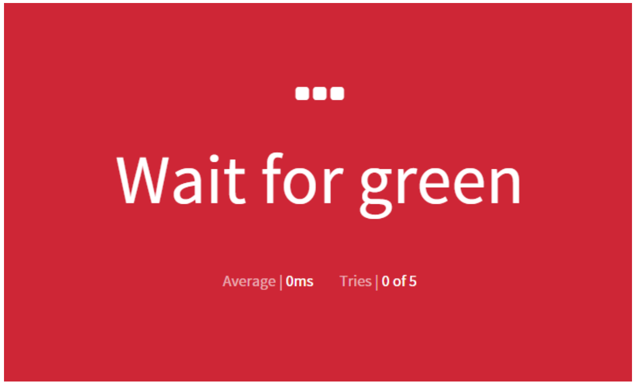
Aalto University

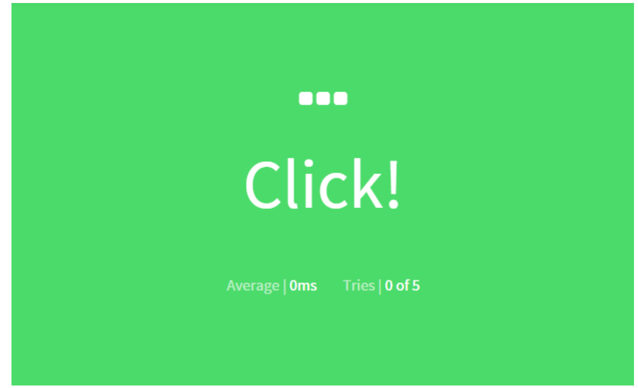
Simple reaction

Reaction times

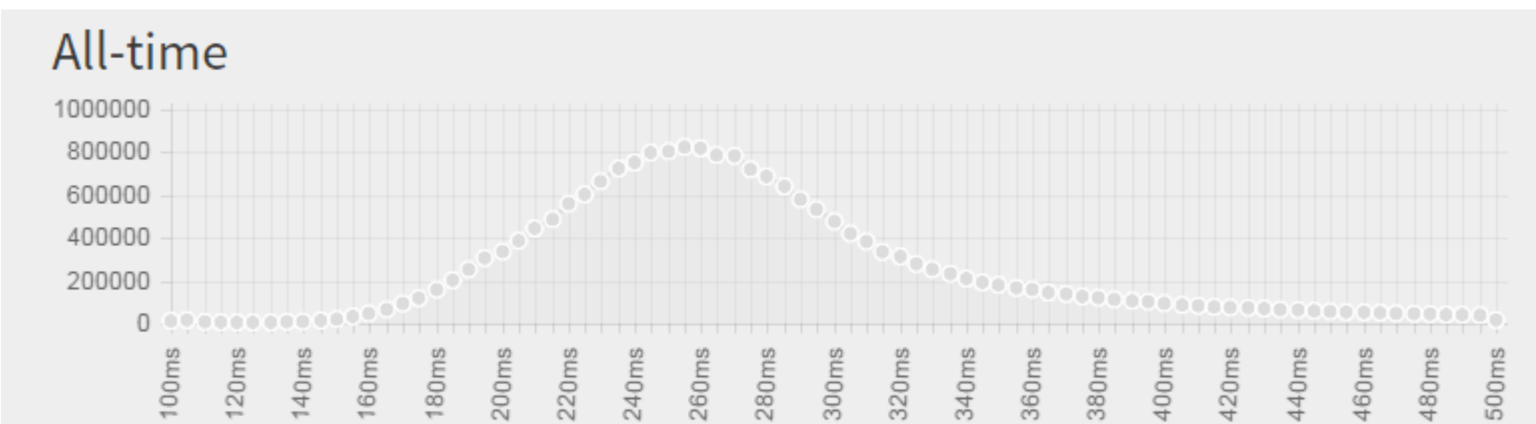


Reaction times “in the wild”



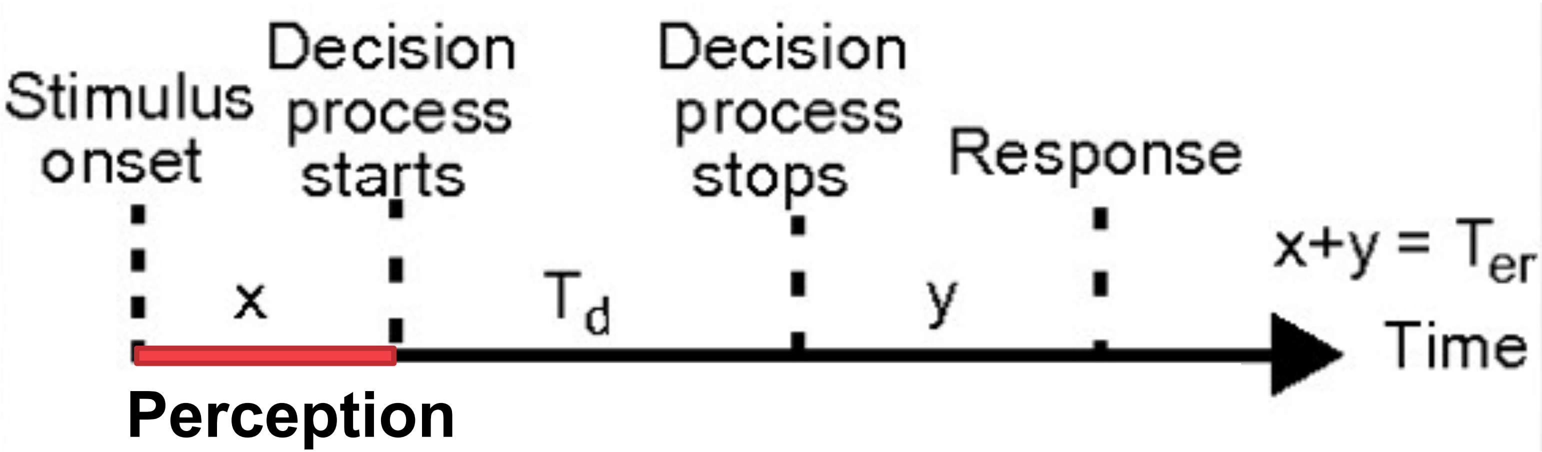


Over 22 million responses
Mean: 268 ms



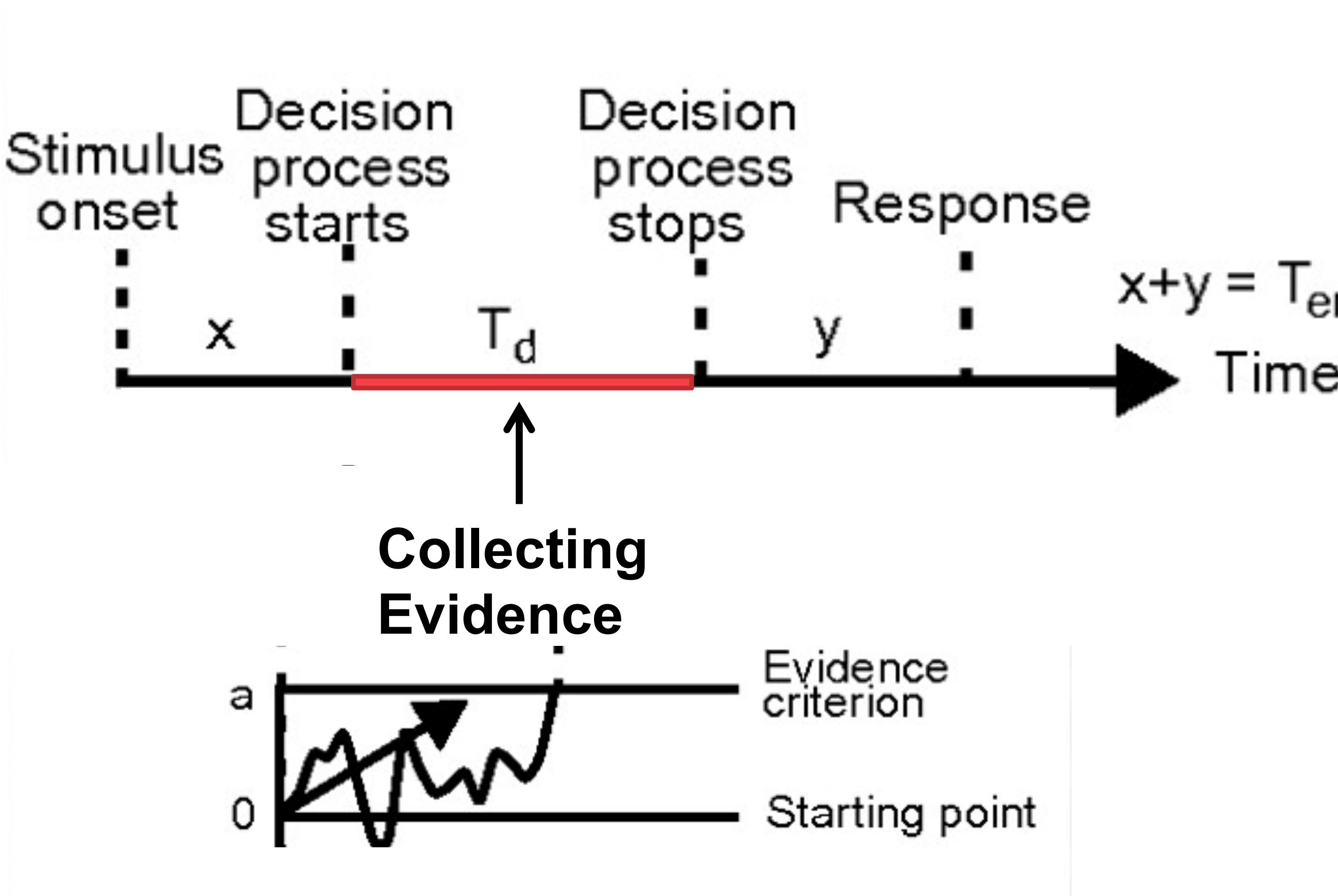
humanbenchmark.com

Ratcliff model

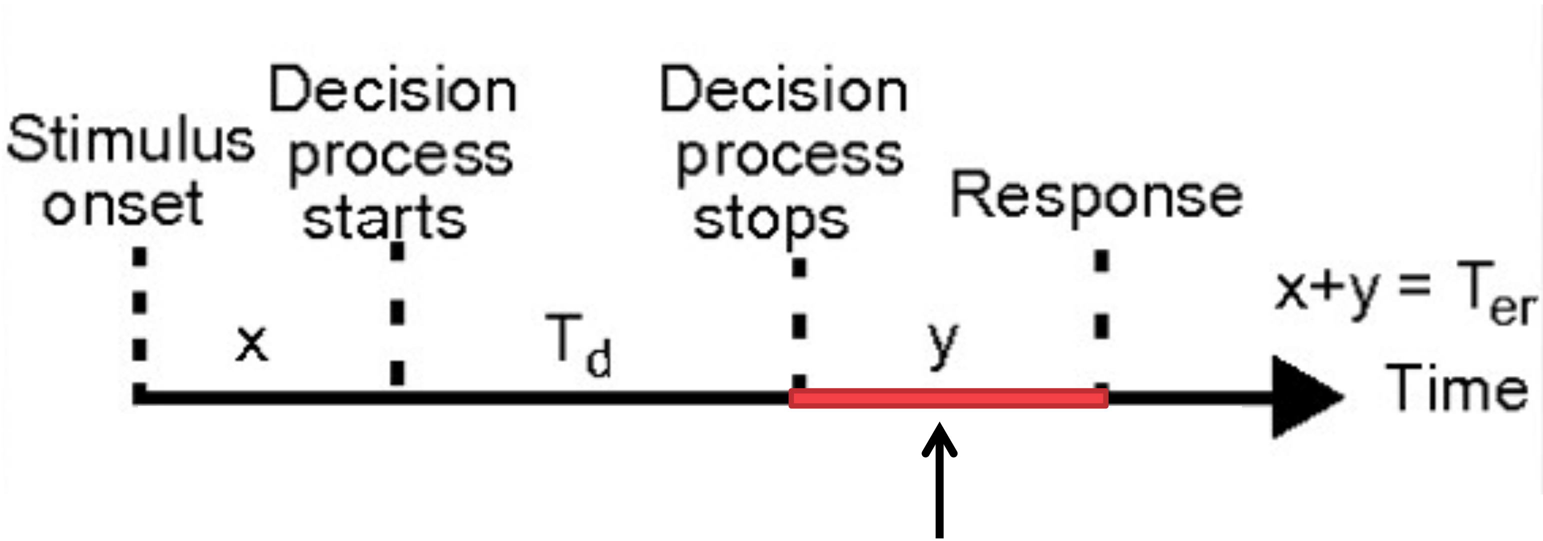


Perception and Encoding

Ratcliff model



Ratcliff model



Motor response

Impact of design on simple reaction

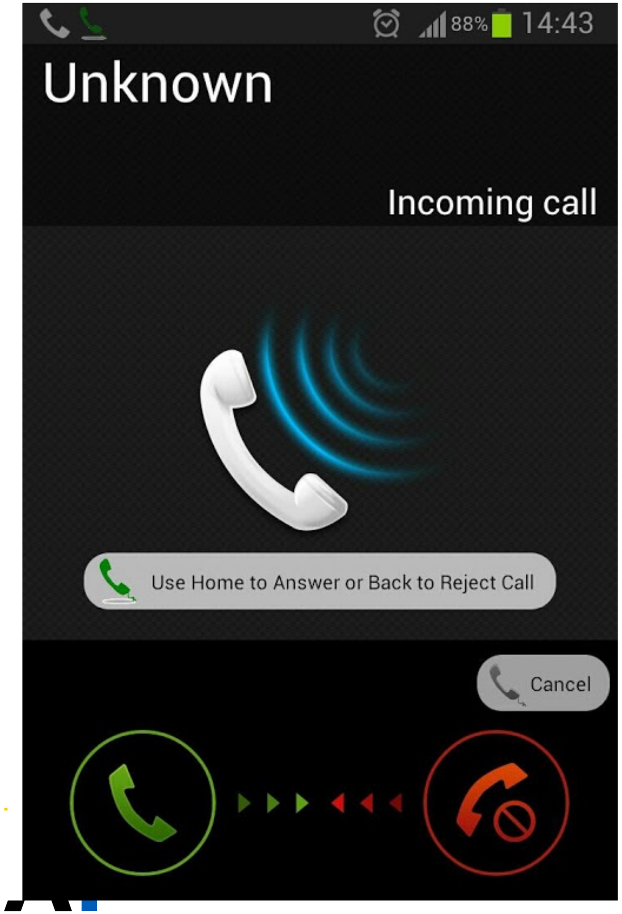
- **Perception and Encoding**
  e.g. Visual slower than auditory, stimulus complexity, stimulus duration and intensity

- **Collecting Evidence**
  user and task dependent, account for those, practice

- **Motor response**
  response complexity, practice

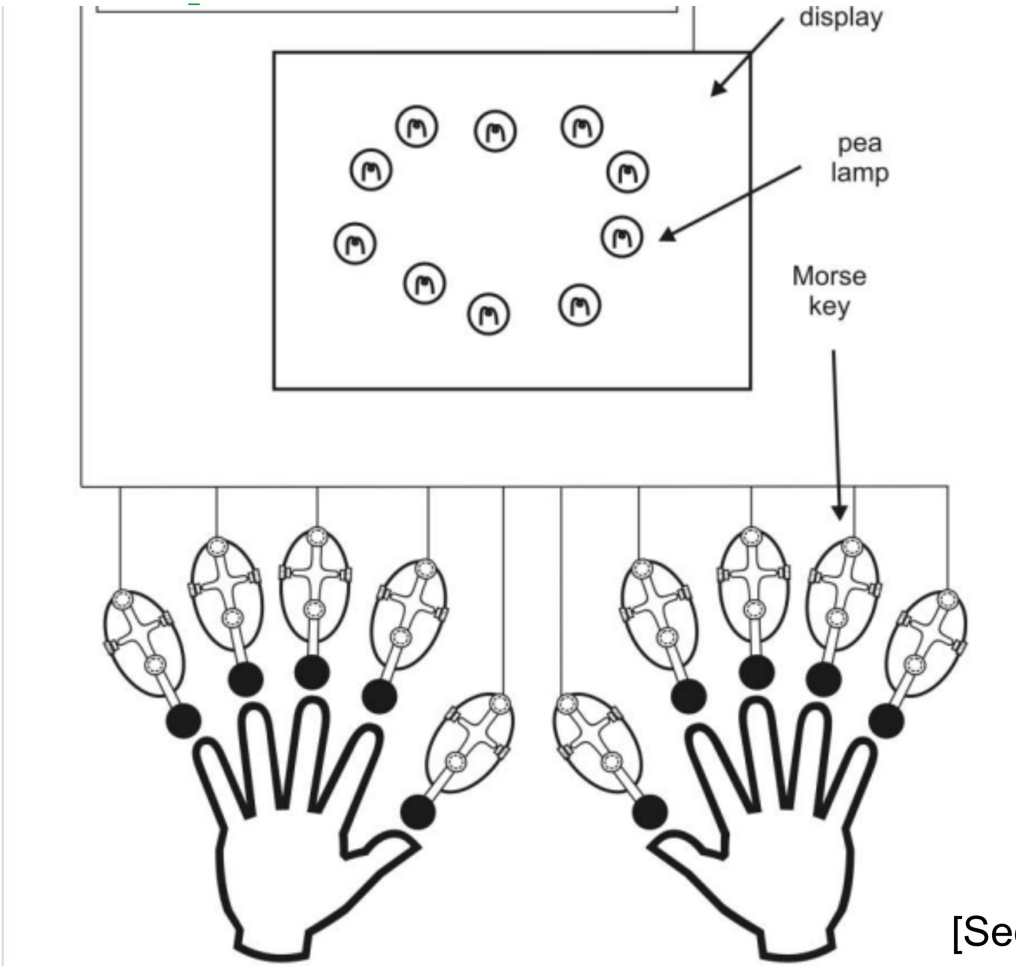
Choice reaction

Examples of choice reaction



Time taken to respond to a stimulus **appropriately**

Hick's experiment

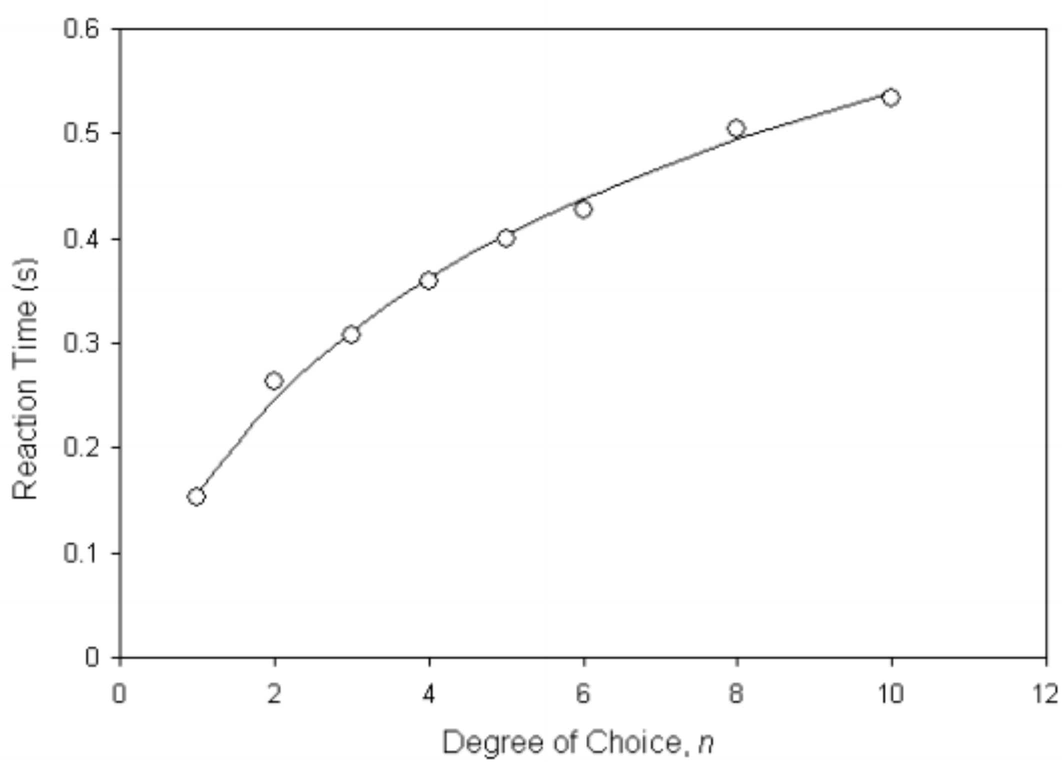


[Seow 2005]

CRT as a function of number of options

CRT = choice reaction time

$$RT = a + b\log_2(n)$$



Information-theoretical interpretation

Reaction time increases with the amount of ***information***

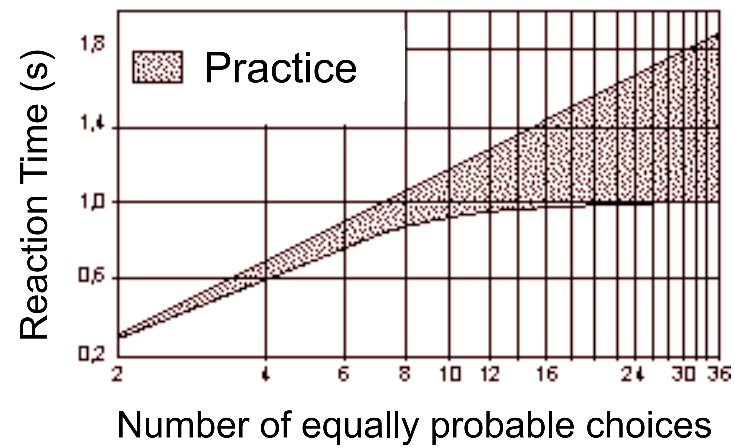
All choices have equal probability:

$$RT = a + b\log_2(n)$$

Choices have different probabilites:

$$RT = a + b \cdot H$$

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$



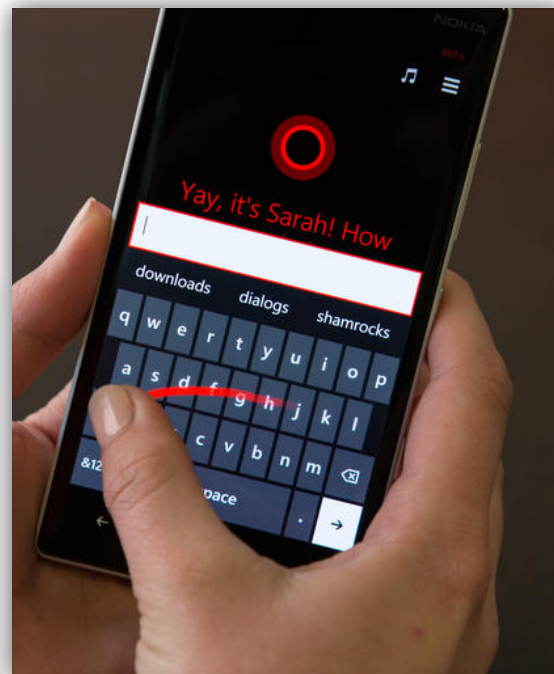
Example: Game

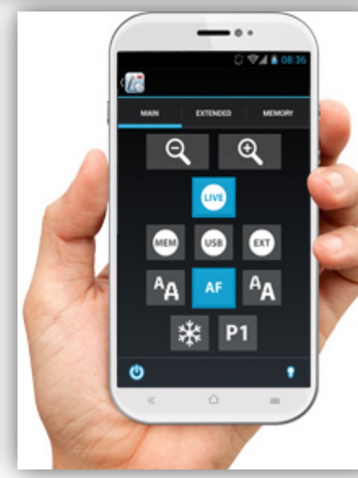
Press arrow key to start

Aalto University

Aimed movements: Fitts' law

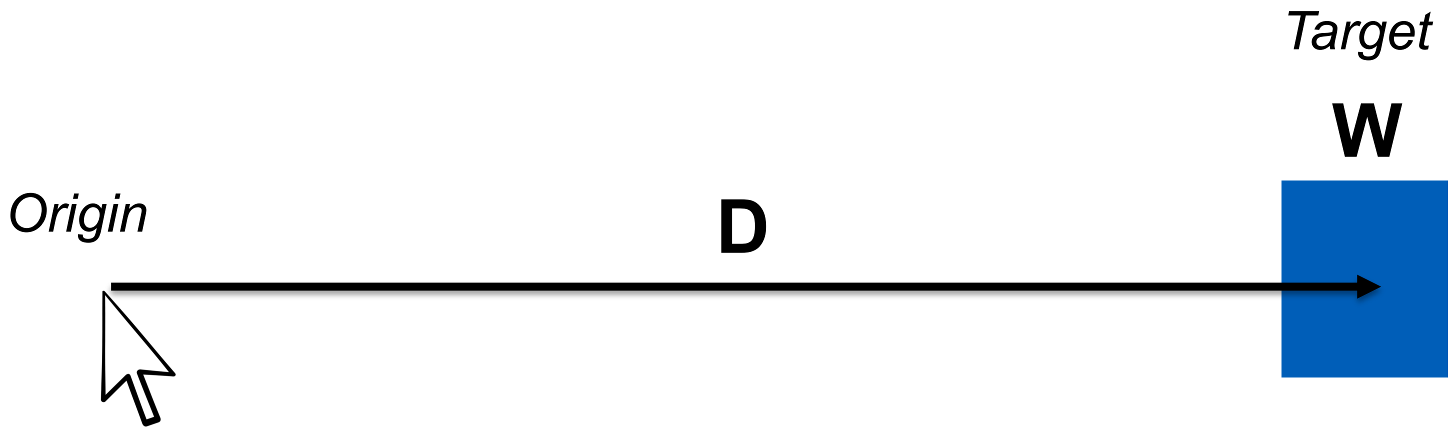
Aimed Movements



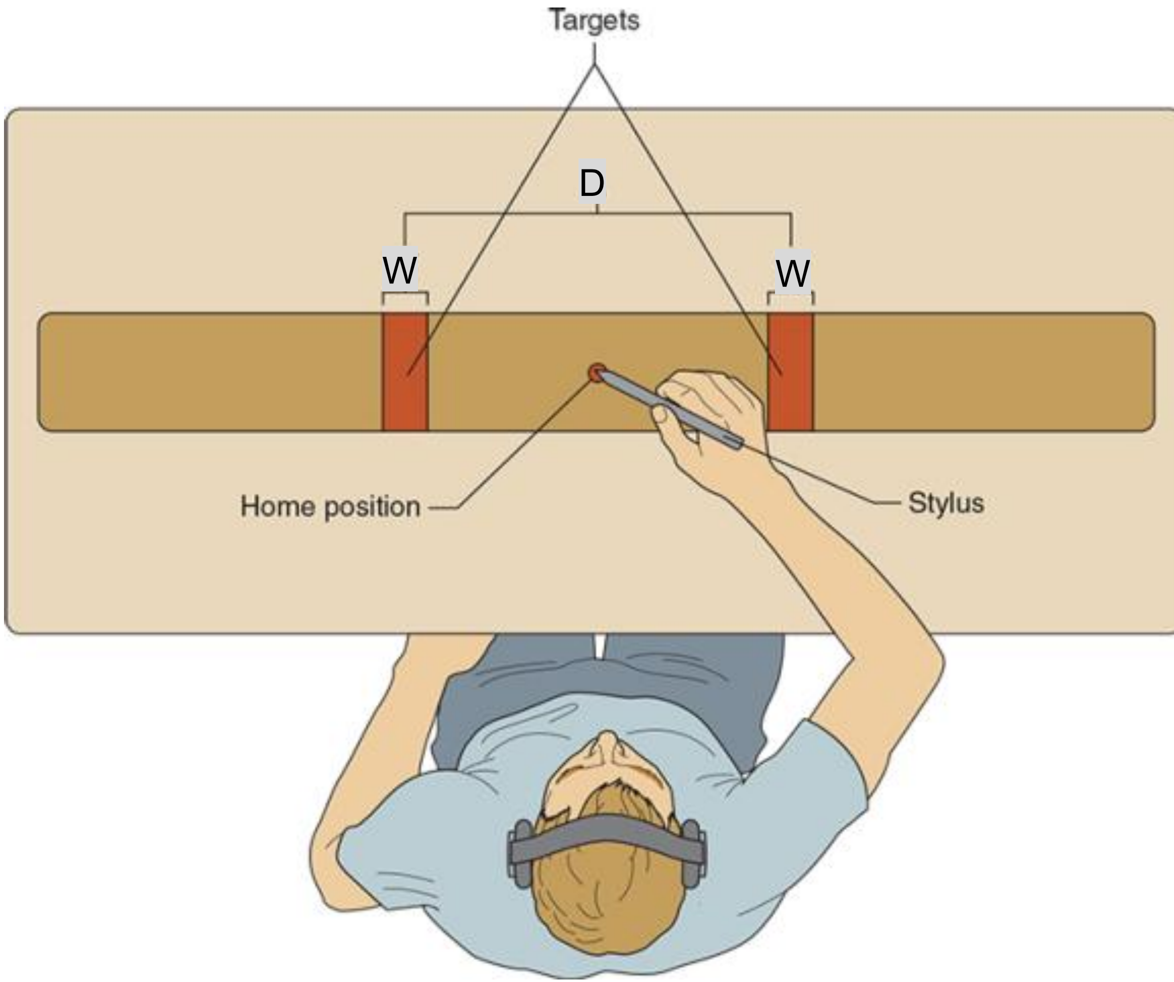


Response demands in pointing

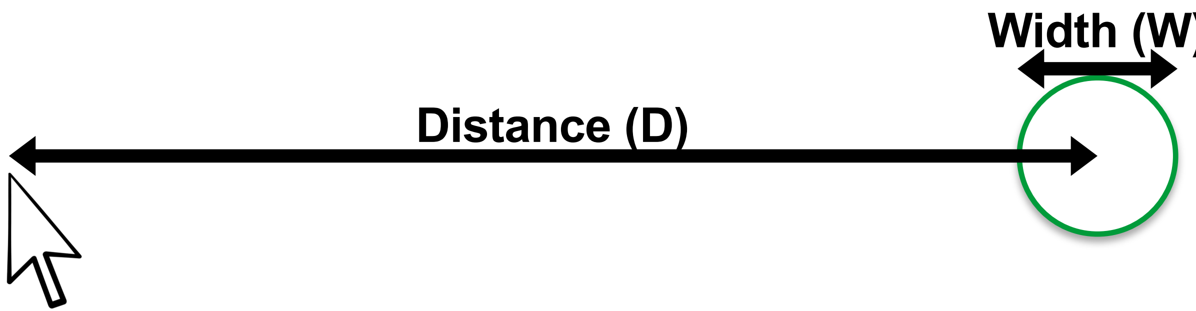
"Select the target as quickly as you can"



Reciprocal pointing experiment



Fitts' law: Idea



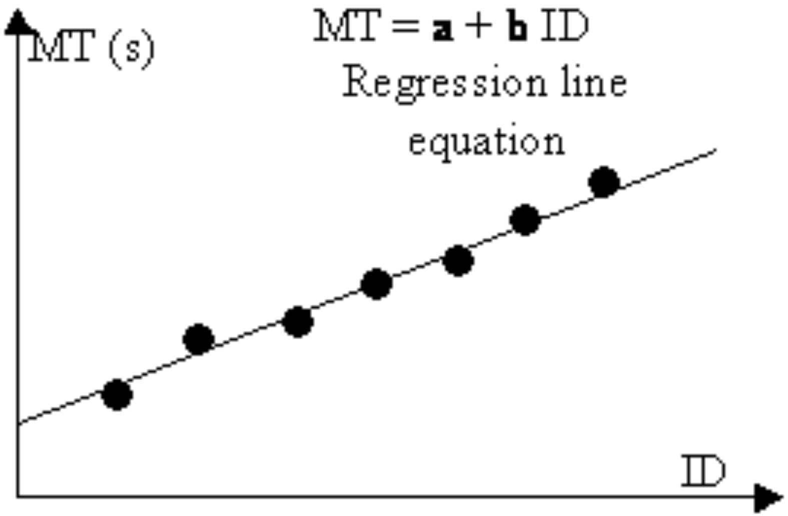
Fitts' law

$$MT = a + b\,\text{ID} = a + b\log_2(D/W + 1)$$

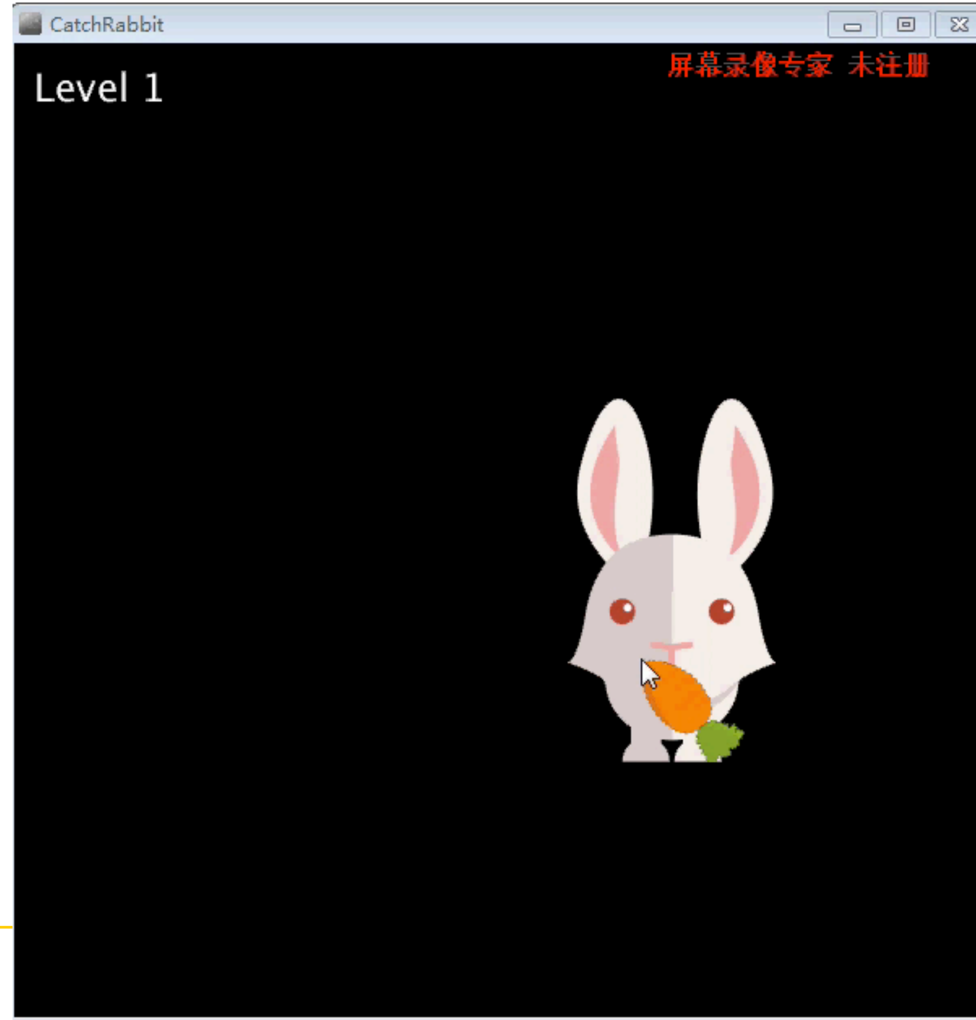
Index of Difficulty

Q: Draw “the Fitts’ diagram”

X-axis: ”Index of difficulty”
Y-axis: Movevement time



Example: Game



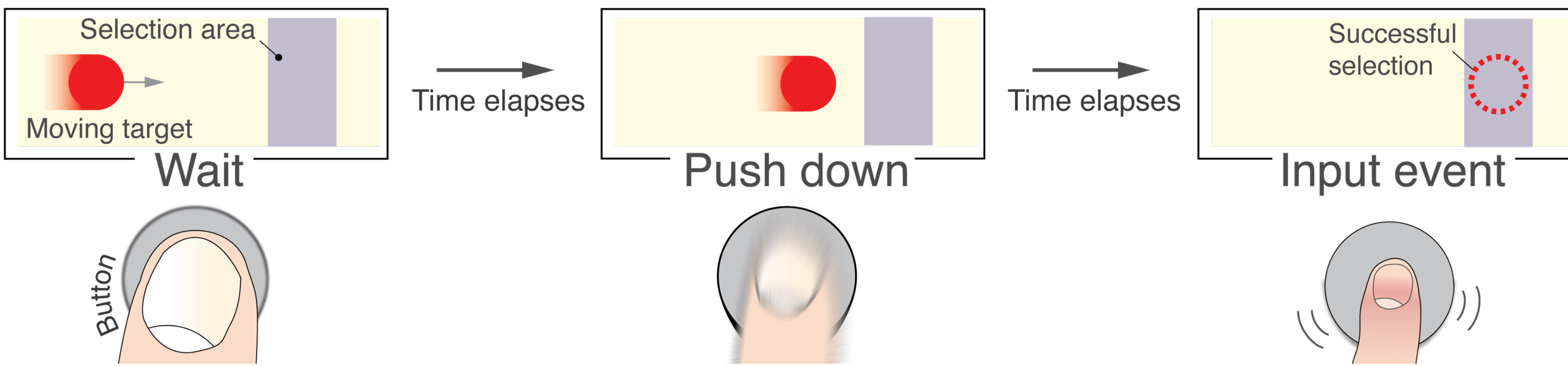
Aalto University

Temporal Pointing Model

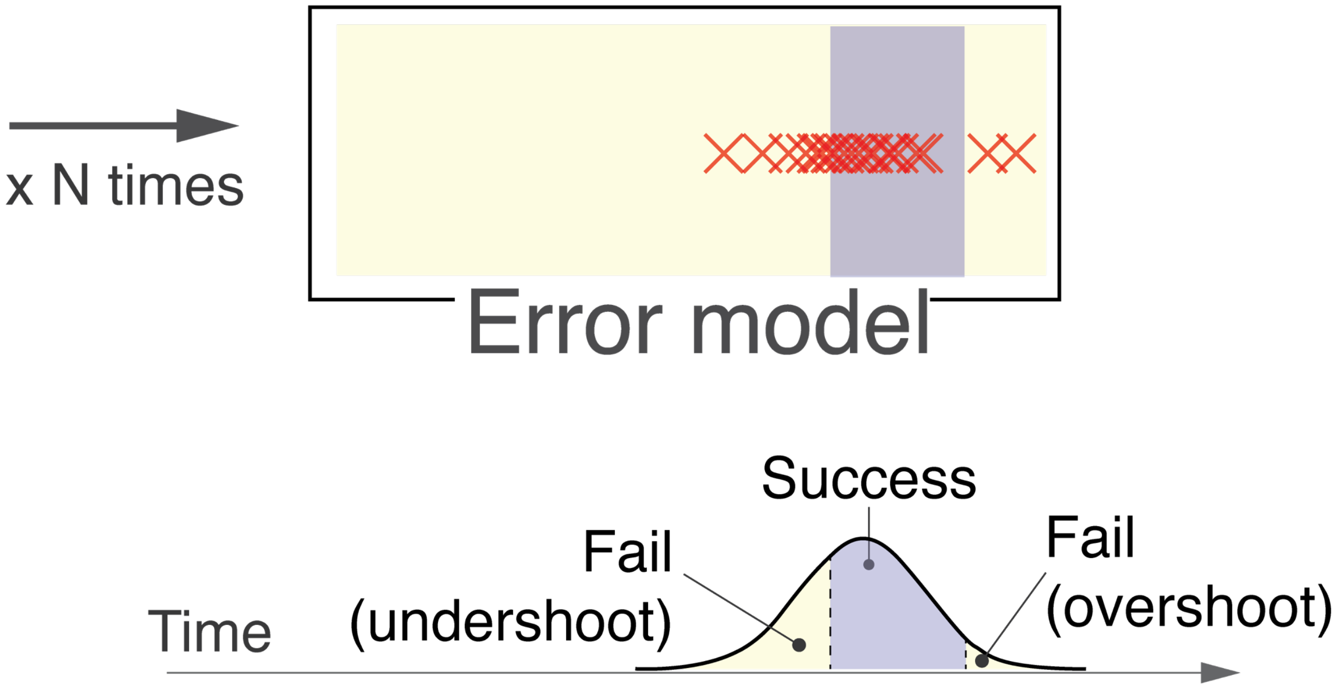
Temporal pointing task

“Press the button when the target appears under selection area”

Model applies when time to target is larger than 600 ms (some anticipation needed)



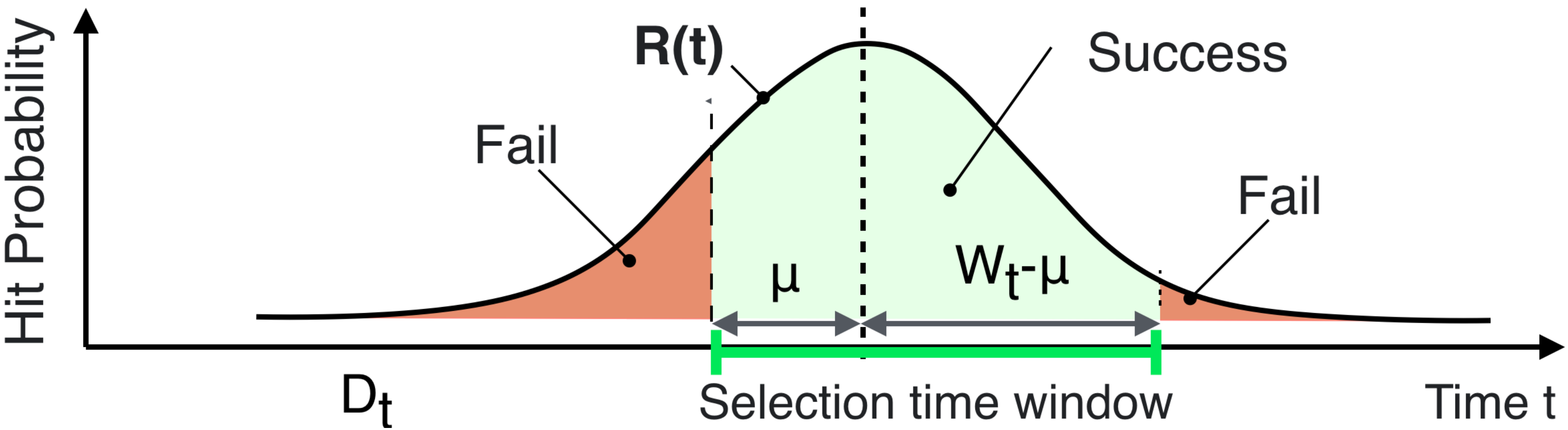
Temporal pointing model



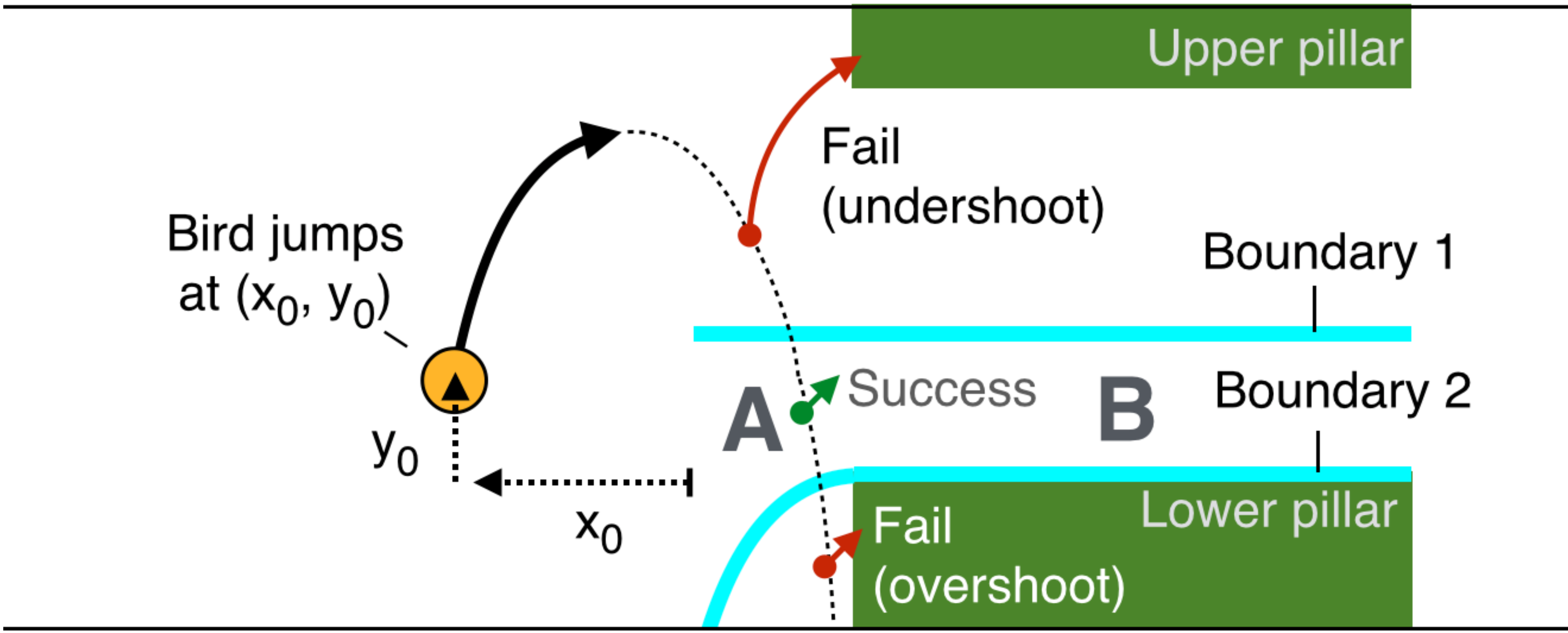
Formula for predicting error rate

$$E(ID_t) = 1 - \frac{1}{2}\left[erf\left(\frac{(1-c_\mu)}{c_\sigma 2^{(ID_t+0.5)}}\right) + erf\left(\frac{c_\mu}{c_\sigma 2^{(ID_t+0.5)}}\right)\right] \quad (7)$$

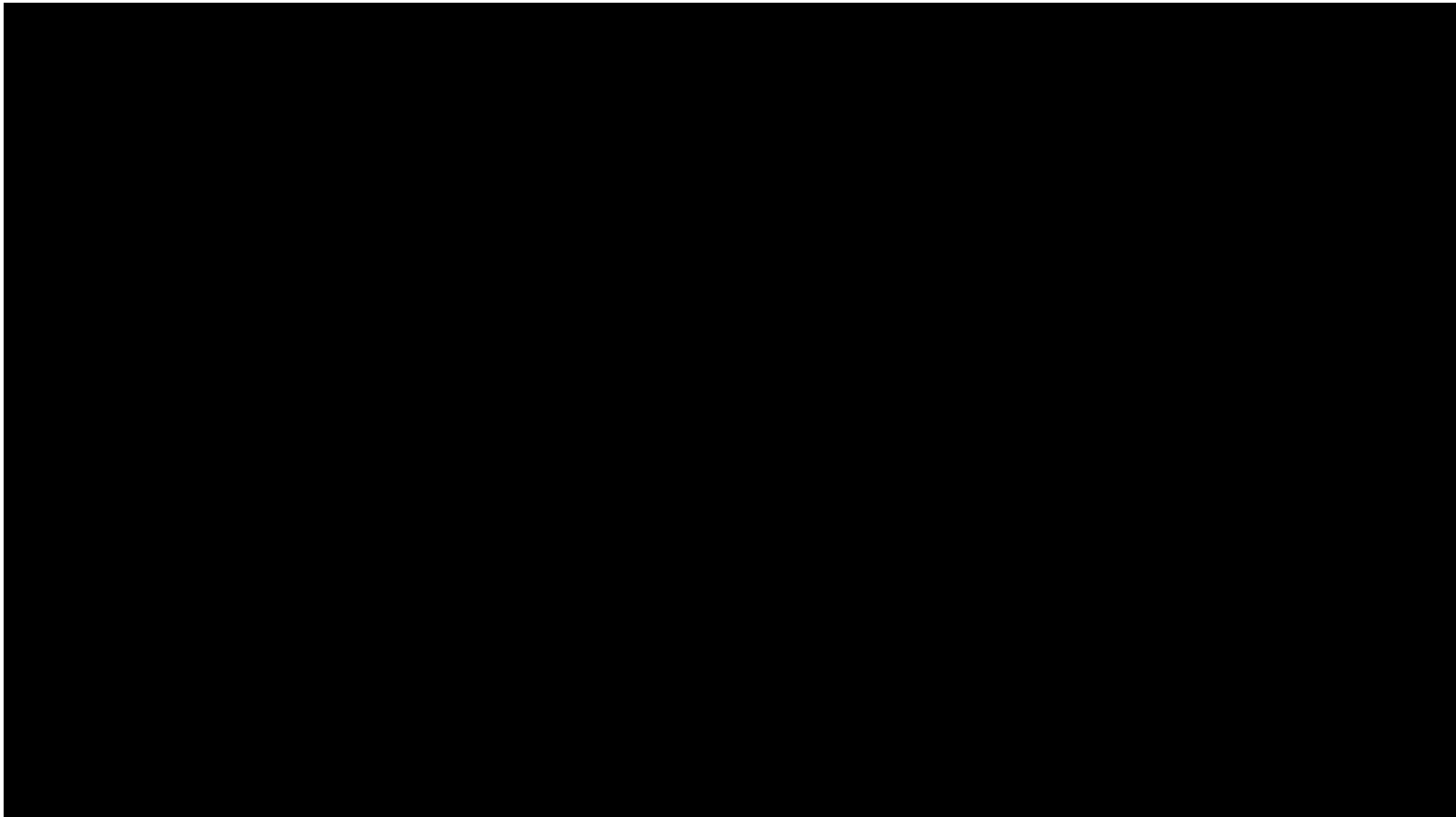
Illustrated



Example application: Flappy bird



Example application: Blinking target



Aalto University

2. Keystroke Level Modeling

Keystroke-level model

A model of task completion time in sequentially performed tasks consisting of simple actions. A memoryless model

Input: Operation sequence, UI elements and layout

Output:

Task completion time **=**

t_K [key stroking]

+ t_P [pointing]

+ t_H [homing]

+ t_D [drawing]

+ t_M [mental operation]

+ t_R [system response]

KLM, a task-level predictive model

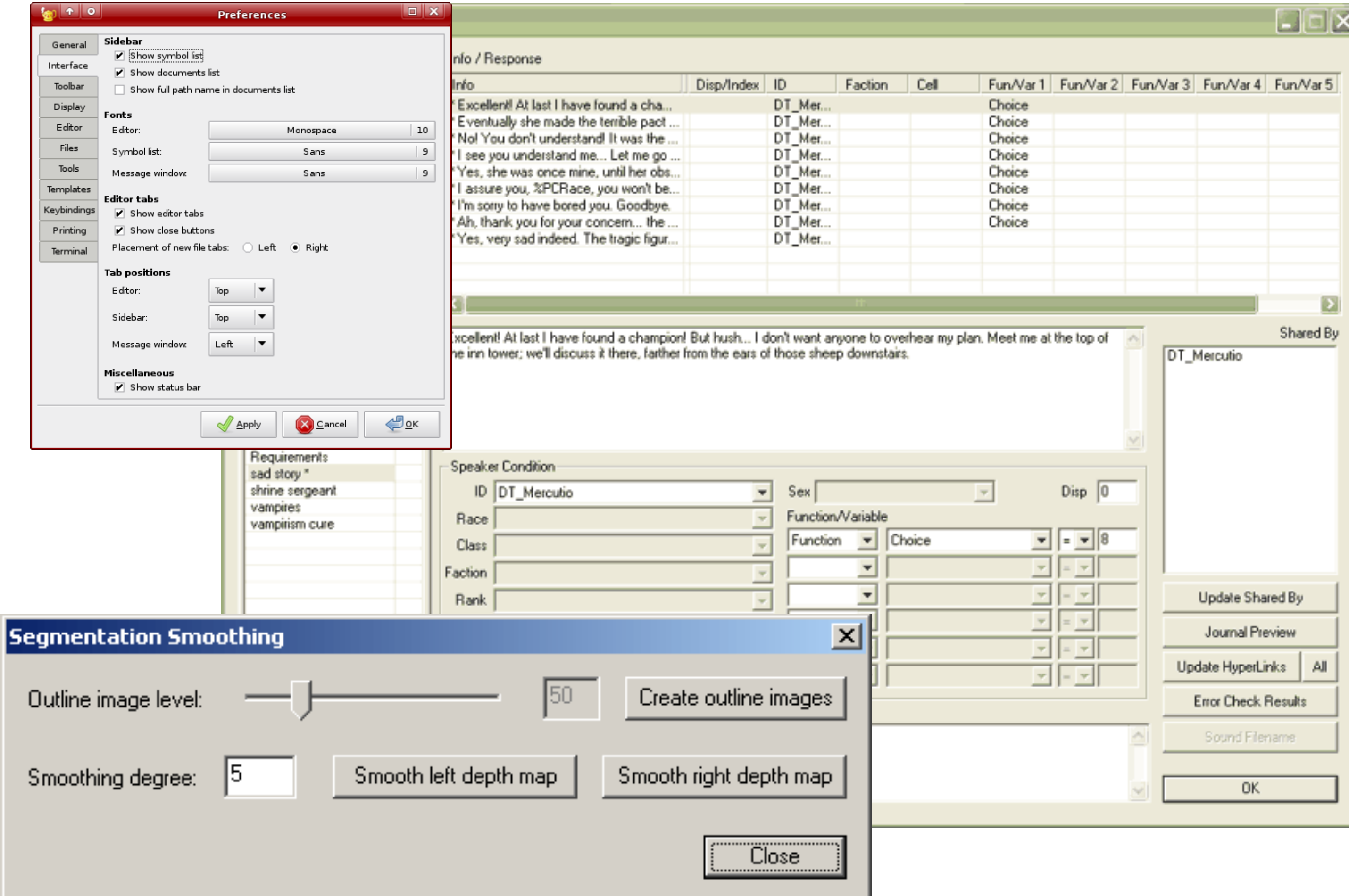
Pros

- Predicts total task completion time (TCT) for UIs operated by discrete commands
    - Some GUIs, web pages, forms, widgets, dialogues, panels, toolbars etc
- Informs design and evaluation

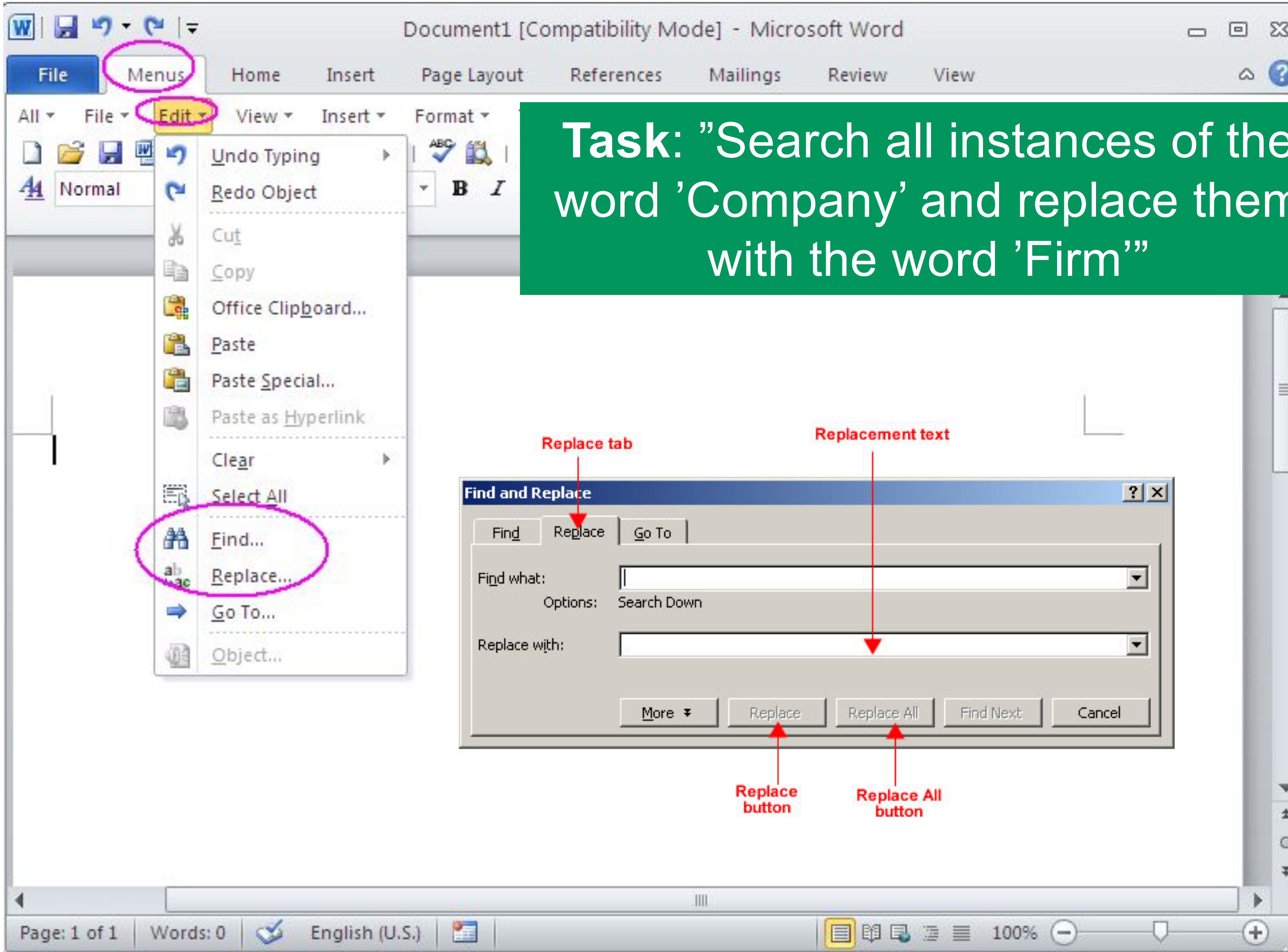
Cons

- A strictly sequential model; no multitasking
- Memory-less
- Focus is on task performance, other aspects of behavior and experience are ignored
    - *Lacks a notion of "semantics" and "contexts"*
    - *Overlooks individual and cultural differences*
    - *Only rough notion of learning (i.e., parameters can be updated)*
- Validity depends on task specifications and model assumptions

Scope: sequentially operated UIs



Task: ”Search all instances of the word ’Company’ and replace them with the word ’Firm’”



Keystroke-level model (KLM)

Task completion time =

t_K [key stroking]

$+\ t_P$ [pointing]

$+\ t_H$ [homing]

$+\ t_D$ [drawing]

$+\ t_M$ [mental operation]

$+\ t_R$ [system response]

Expert typist (90 wpm): .12 sec
Average skilled typist (55 wpm): .20 sec
Average nonsecretarial typist (40 wpm): .28 sec
Worst typist (unfamiliar with keyboard): 1.2 sec

Fitts' law

Time it takes to move hand from one input device to another. Typically constant, e.g. 0.4s

Estimated by the researcher or looked up from a lookup table

Observed response time

Example: replacing a word

Reach for mouse	H[mouse]	0.40
Move pointer to "Replace" button	P[menu item]	1.10
Click on "Replace" command	K[mouse]	0.20
Home on keyboard	H[keyboard]	0.40
Specify word to be replaced	M[word]	2.15
Reach for mouse	H[mouse]	0.40
Point to correct field	P[field]	1.10
Click on field	K[mouse]	0.20
Home on keyboard	H[keyboard]	0.40
Type new word	M[word]	2.15
Reach for mouse	H[mouse]	0.40
Move pointer on Replace-all	P[replace-all]	1.10
Click on field	K[mouse]	0.20
Total		**10.2**

Task:

1. Take pen and paper
2. Open browser: Wikipedia page on “Keystroke-level model”

Task: Estimate TCT (task completion time) for a task done with the UI shown on the following page…

Task: 10 minutes

Task: Do a KLM model for the task of entering "Aalto" and pressing "Google Search"

http://mattbors.tumblr.com/post/100671142990/the-ultimatum-game-more-comics-at-the-nib

Reliability of KLM-based estimates

Normally obtained via empirical measurements carried out on representative users and devices

- When these conditions change, estimates change, too

Point estimates lose information about variability

Memory-less (prior states do not affect estimates)

Limitations of KLM

Limited behavior: “Script-like” task performance: Do this, then that, then that, …

- No perception, choice, decision-making…

Limited scope of UIs: Selection & data entry mostly; Forms, settings, panels, menus etc

Parameter acquisition: KLM values may not be available

Simple error analysis with KLM

We assume that an error occurs with probability of *p*

Now, the new expected $TCT_{average}$ is

$$TCT_{average} = (1 - p) * TCT_{no\ error} + p * TCT_{error\ occurred}$$

Instructions:

- Identify the most costly & probable error
- Estimate *p*
- Compute the new TCT

Transition matrix for error analysis

	1a	1b	2a	2b	2c	2d	3a	3b	3c	3d	4a	4b	4c	4d	5a	5b	5c	5d	6
1a	—	L	L	L	L	L	I	I	I	I	I	I	I	I	—	—	—	—	L
1b	L	—	I	I	I	I	I	I	I	I	I	I	I	I	—	—	—	—	I
2a	—	—	L	—	—	—	L	—	—	—	I	—	—	—	—	—	—	—	L
2b	—	—	—	L	—	—	—	L	—	—	—	I	—	—	—	—	—	—	L
2c	—	—	—	—	L	—	—	—	L	—	—	—	I	—	—	—	—	—	L
2d	—	—	—	—	—	L	—	—	—	L	—	—	—	I	—	—	—	—	L
3a	—	—	I	—	—	—	L	—	—	—	L	—	—	—	—	—	—	—	L
3b	—	—	—	I	—	—	—	L	—	—	—	L	—	—	—	—	—	—	L
3c	—	—	—	—	I	—	—	—	L	—	—	—	L	—	—	—	—	—	L
3d	—	—	—	—	—	I	—	—	—	L	—	—	—	L	—	—	—	—	L
4a	—	—	I	—	—	—	I	—	—	—	L	—	—	—	L	—	—	—	L
4b	—	—	—	I	—	—	—	I	—	—	—	L	—	—	—	L	—	—	L
4c	—	—	—	—	I	—	—	—	I	—	—	—	L	—	—	—	L	—	L
4d	—	—	—	—	—	I	—	—	—	I	—	—	—	L	—	—	—	L	L
5a	L	—	I	—	—	—	I	—	—	—	I	—	—	—	L	—	—	—	L
5b	L	—	—	I	—	—	—	I	—	—	—	I	—	—	—	L	—	—	L
5c	L	—	—	—	I	—	—	—	I	—	—	—	I	—	—	—	L	—	L
5d	L	—	—	—	—	I	—	—	—	I	—	—	—	I	—	—	—	L	L
6	—	L	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	L

Figure 6 Transition matrix

Many common causes of errors are ignored

Motor execution variability

Misperception of display and change blindness

Level of skill (e.g., novices vs. experts)

Wrong or partial beliefs about the system

Spatial memory and inference (getting lost)

Cognitive load

Multitasking

Decision-making fallacies

Idiosyncratic differences (e.g., age groups)

Aalto University

Conclusion

Response demand

Response demands characterizes the response the user must give. It consists of:

1. **A response set (options for responding)**
2. **A transducing mechanism**
3. **Spatial and temporal constraints**
4. **Feedback**

→ Determine the model that should be used

KLM

Predicts skilled user's performance in *sequentially operated* tasks

Sum up time spent in 6 elementary operations

Parameter values are terminal and user specific

A handy "back of the envelope" tool for first estimates!