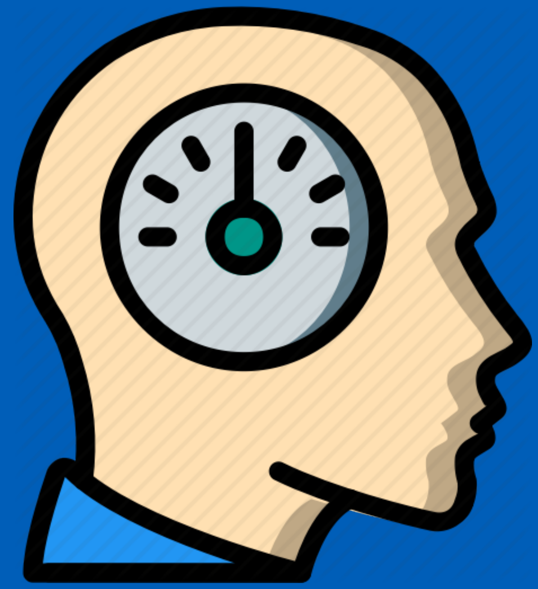
Aalto University

ELEC-E7851
COMPUTATIONAL
USER INTERFACE DESIGN

Lecture 3: Human Performance Models



PII/2020
Antti Oulasvirta
Aalto University
userinterfaces.aalto.fi

Empirical research on human performance

Human performance models

What: Mathematical functions that connect three types of variables:

1. **Those describing initial task conditions**
2. **Predictions for performance (e.g. speed, accuracy)**
3. **Free parameters** (free = determined empirically)

Why: Accurate-but-practical models to inform design

Modeling workflow:

- **Model construction**
- **Parameter recovery**
- **Model fitting**
- **Model validation**
- **Model selection**

Engineering models of human performance try to find a pragmatic trade-off between model-complexity and predictive accuracy

Why is this topic important?

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

Basic capabilities and limitations of humans in interaction

Models allow you to find optimal tradeoffs among design decisions

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

Warm-up exercise 10 mins

Done in pairs (breakout rooms)

We use this page: http://simonwallner.at/ext/fitts/

One student does the task to produce sufficient data. The other studies Fitts' law (on the page) in order describe what the results mean. Report back in text format:

"We obtained the following results.., which mean ..."

When time is due, you will report your findings by pasting them to the chat in the main room.

Note: The following materials are for self-study. The Notebook is the main materials in this lecture

Learning objectives in this lecture

1. Response processes
Human performance with spatial, temporal, and choice properties

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

Fitts' law

Hick-Hyman law

KLM

Jupyter (Python) Notebooks

Lecture slides

Aalto University

Introduction

Response processes

Overview of response processes

Response set; a transducer; feedback

Response tasks are defined by response requirements

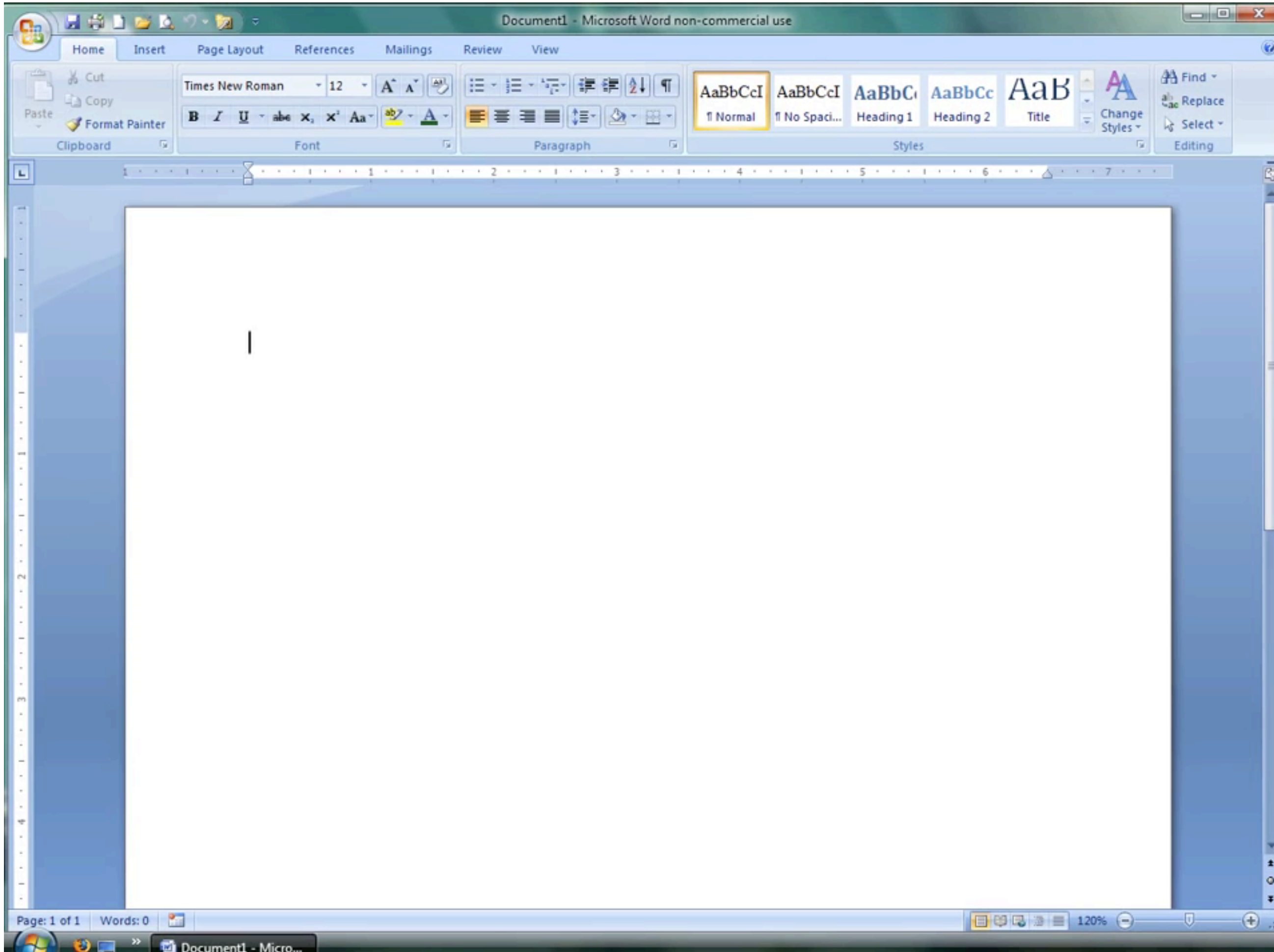
Spatial demand

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

Example: Subway Surf

Example: Microsoft Word



Patient monitoring


RESTROOMS

What’s assumed in this design?

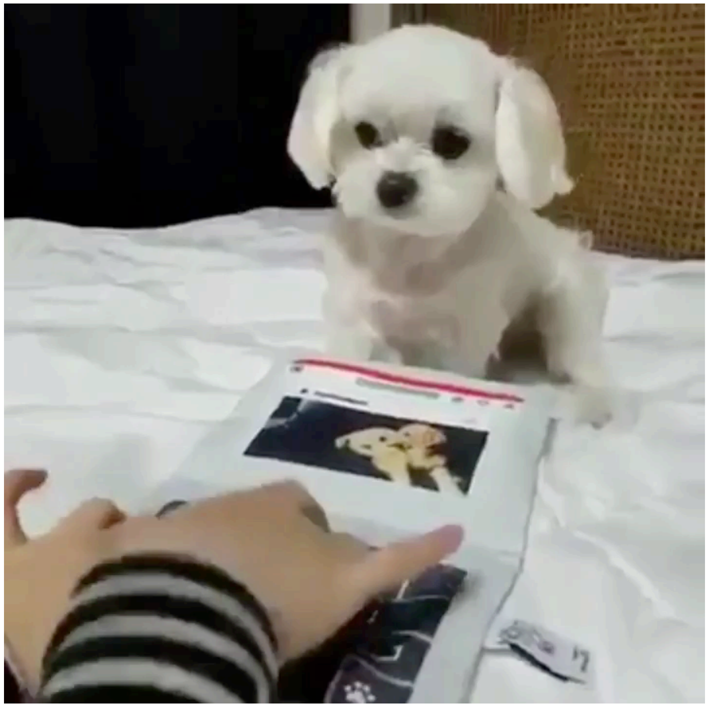
Definitions

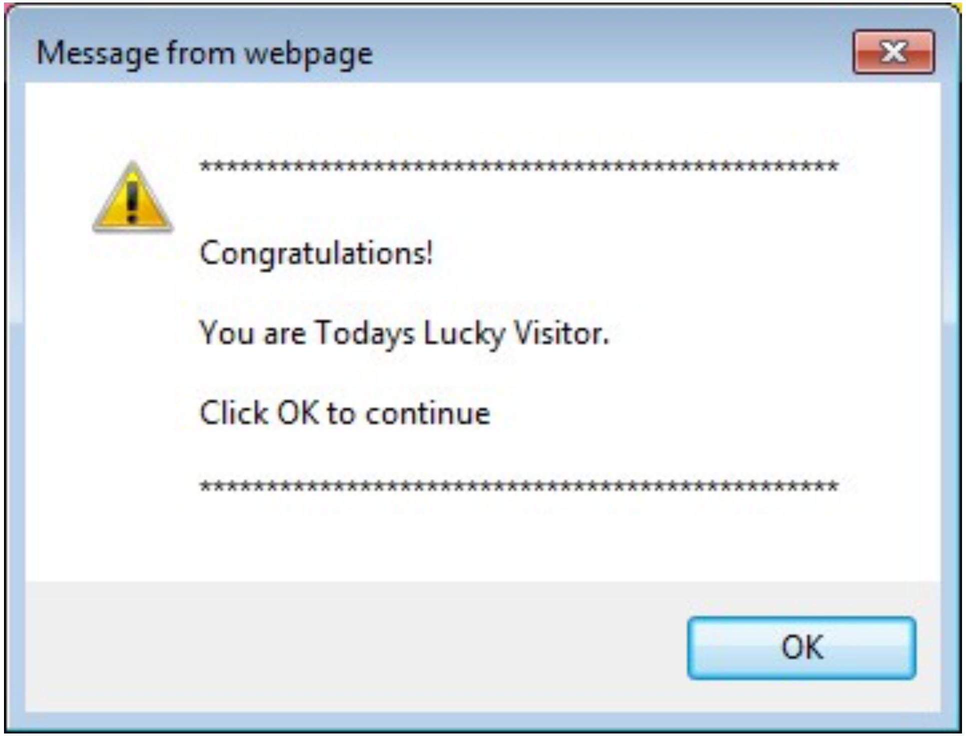
A response is action taken by user within a constrained set of options defined by the computer.

- **The computer is in some state with a limited set of options, which is transformed according to the user's input.**
- **The set of options is defined by 1) set size (number of options), 2) spatial and 3) temporal constraints.**

A response process is a model of the temporal events shaping human performance in responses (speed and accuracy)

Why is this NOT a response?

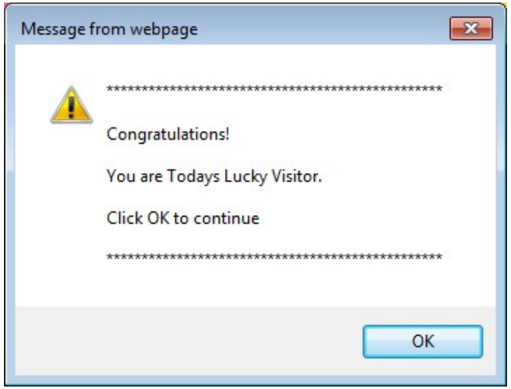



Message from webpage
****
Congratulations!
You are Todays Lucky Visitor.
Click OK to continue
****
OK

“Response demand”

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

1. A response set (e.g., “OK”)
2. A transducing mechanism (e.g., keyboard)
3. Feedback (e.g., dialogue disappears)



Describe the response demand in emergency braking

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

Spatial and temporal characteristics

Emergency braking: Push the right pedal immediately

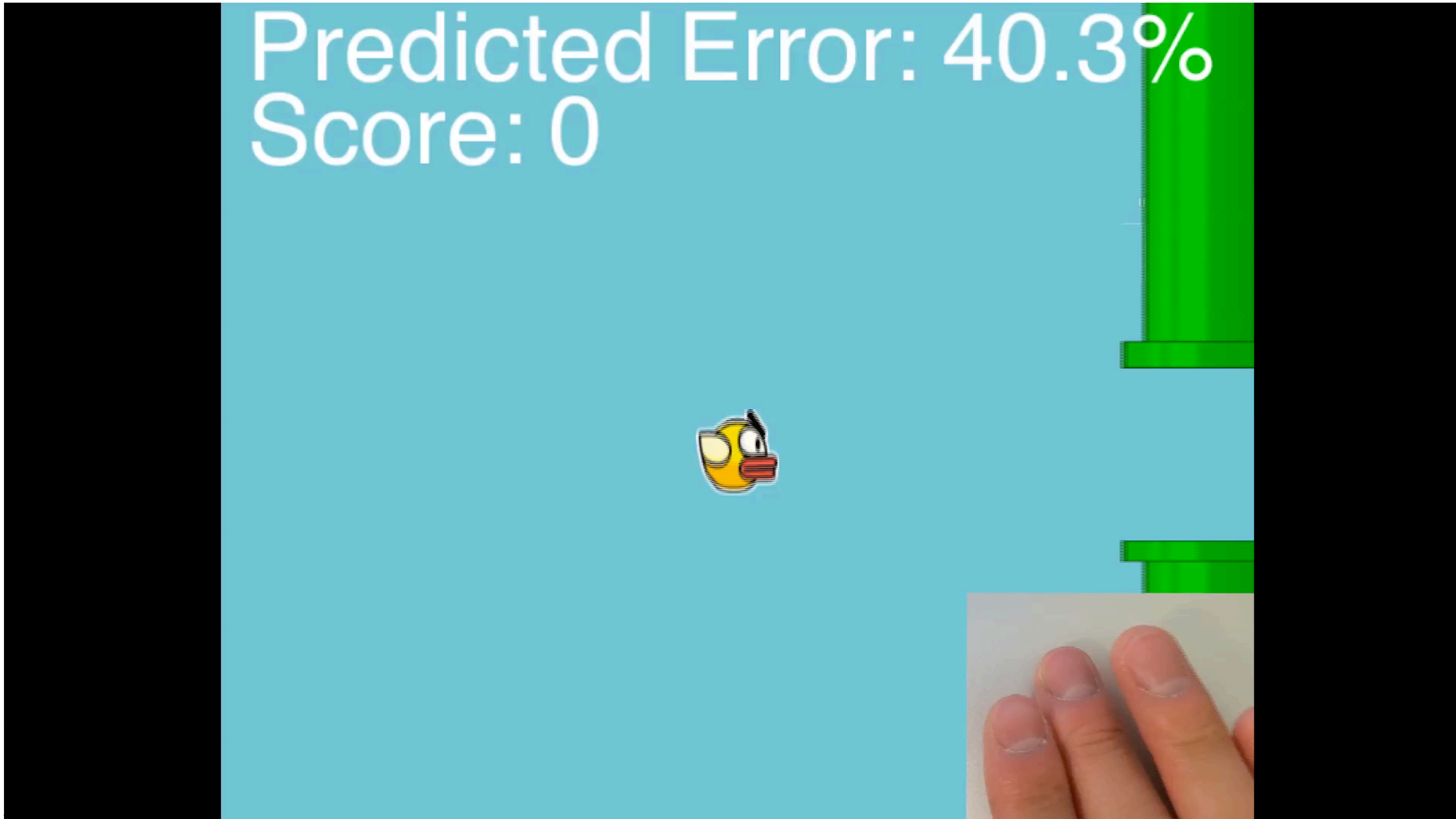
Calling an elevator: Hit the right button and get it activated; no hurry

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

…

Spatial demands

Temporal demands



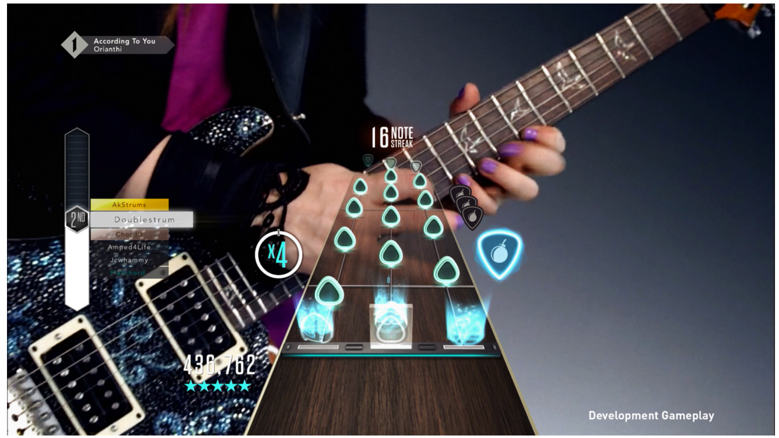
Taxonomy of response processes

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

Which response type?



Interception

A spatially and temporally bound target

Q: Empirical factors affecting response performance

Distractors

Preview time

Size of response set

Input device

Feedback

Motivation: Uses of response process models

Recognize the right response process in a given HCI task

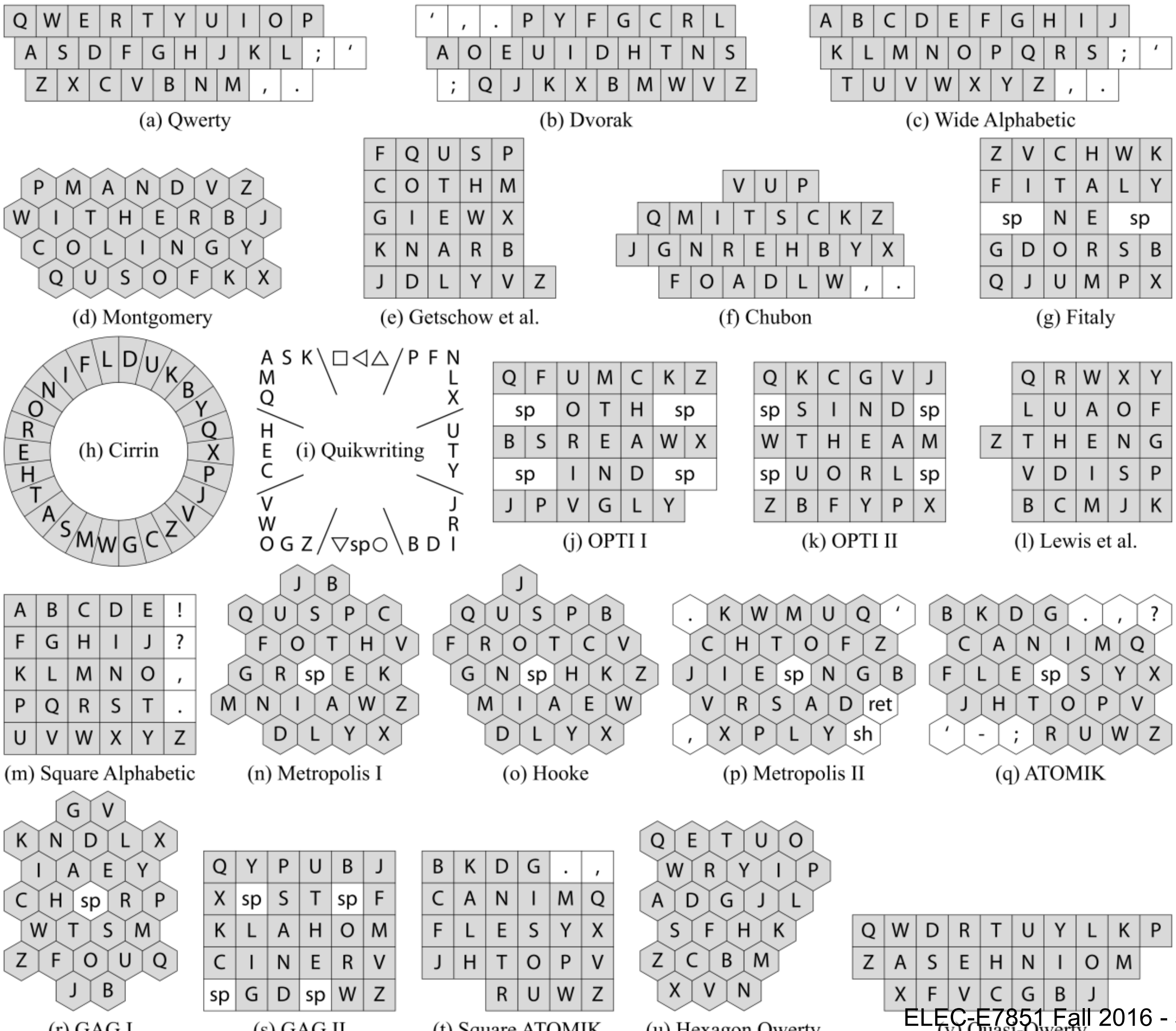
Know the basic models (Fitts’ law and Hick’s law) and understand their position among RP models

Analyze trade-offs using appropriate model mathematically

Use models to enhance designs analytically / computationally

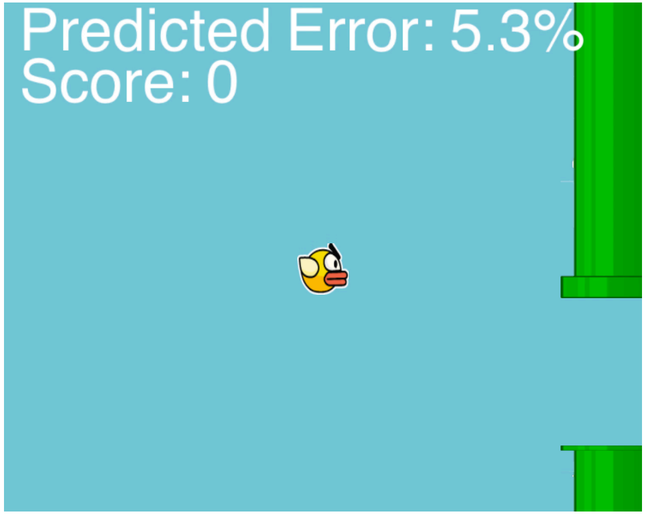
Improve layouts

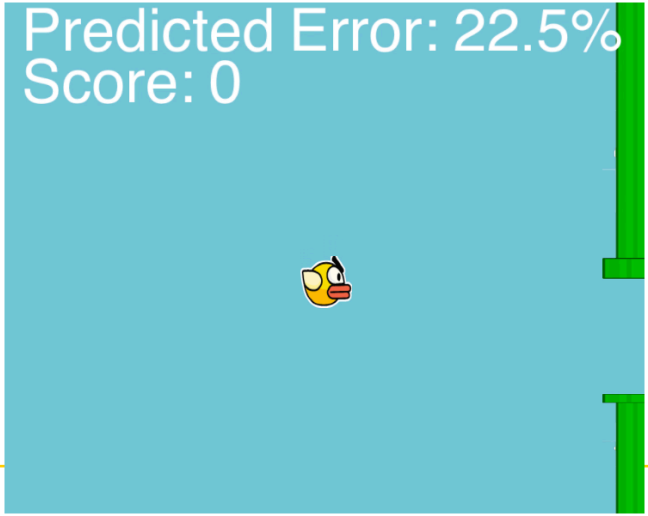
Figure 1. Pareto optimized Arabic keyboard layout

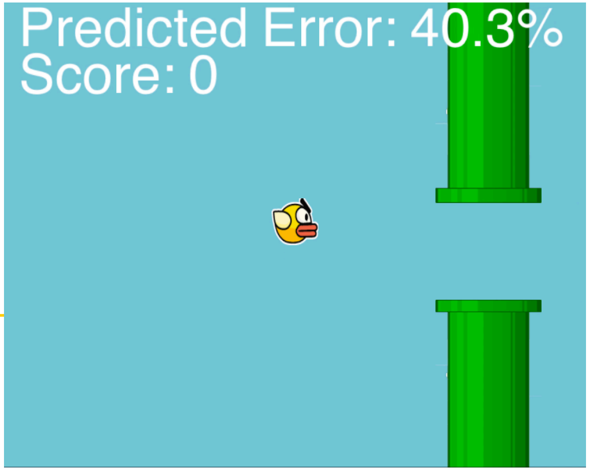

(a) Qwerty
(b) Dvorak
(c) Wide Alphabetic
(d) Montgomery
(e) Getschow et al.
(f) Chubon
(g) Fitaly
(h) Cirrin
(i) Quikwriting
(j) OPTI I
(k) OPTI II
(l) Lewis et al.
(m) Square Alphabetic
(n) Metropolis I
(o) Hooke
(p) Metropolis II
(q) ATOMIK
(r) GAG I
(s) GAG II
(t) Square ATOMIK
(u) Hexagon Qwerty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

Control level of difficulty in user responses

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







Goals today:
Fitts' law
Hick-Hyman law

Aalto University

Response Process Models

From simple reaction to aimed movement

Taxonomy of response processes

Spatial demand

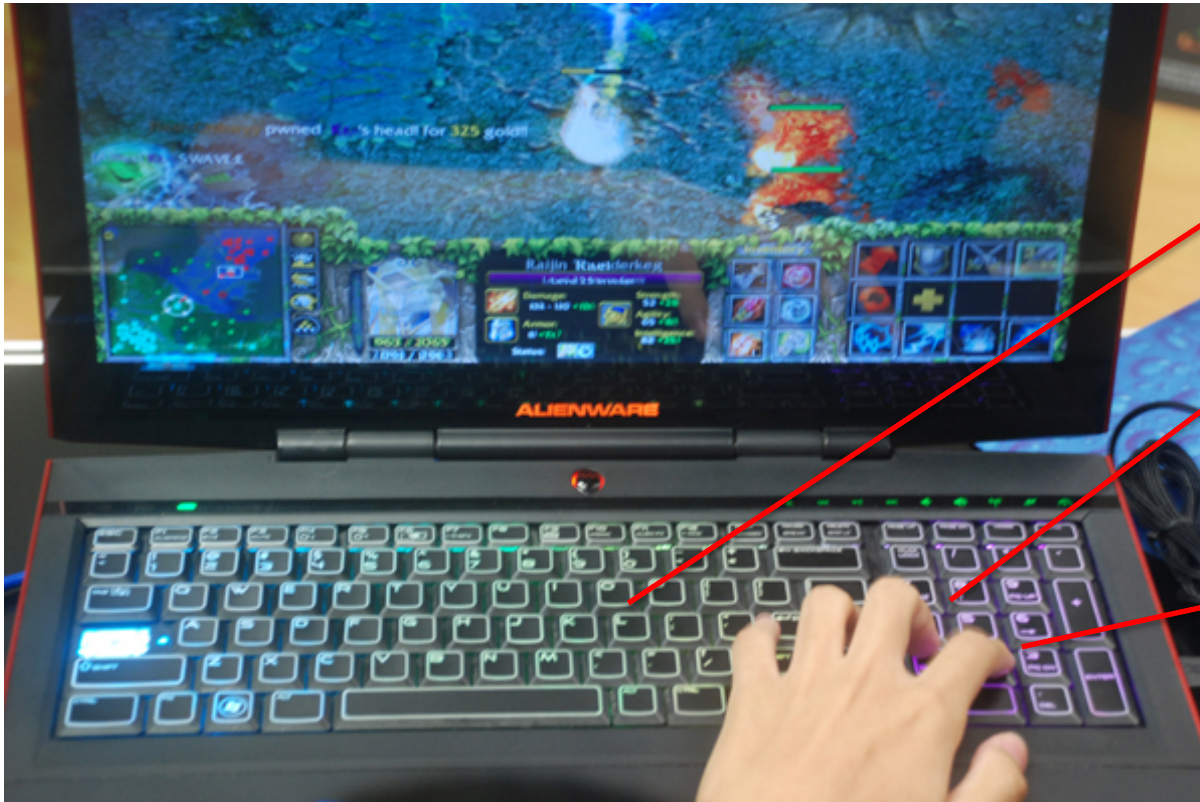
Temporal demand

	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

Aalto University

Lee & Oulasvirta CHI'16

Overview



Aiming

Choice reaction

Simple reaction

Overview

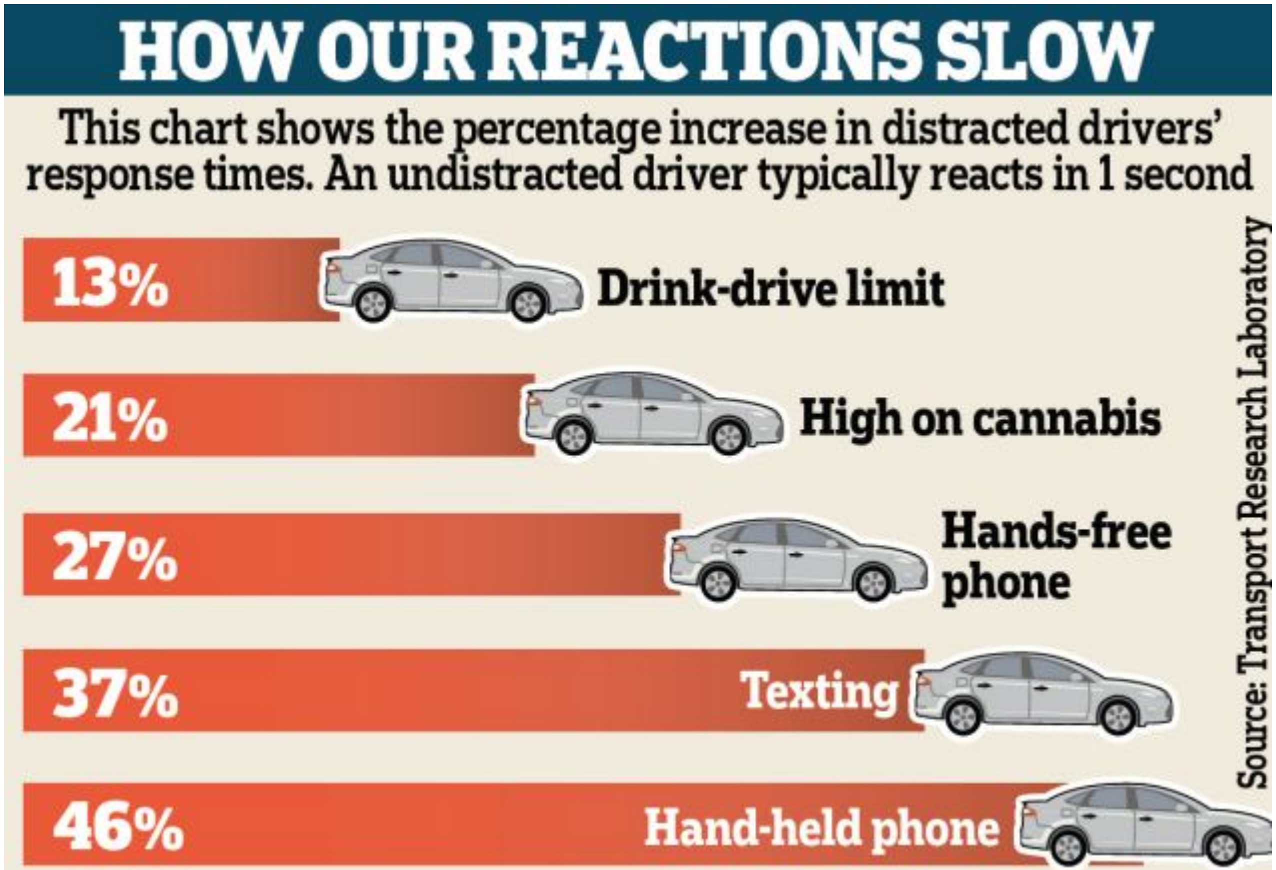
The mathematical formula will be given also in Assignments, we here focus on the main ideas!

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

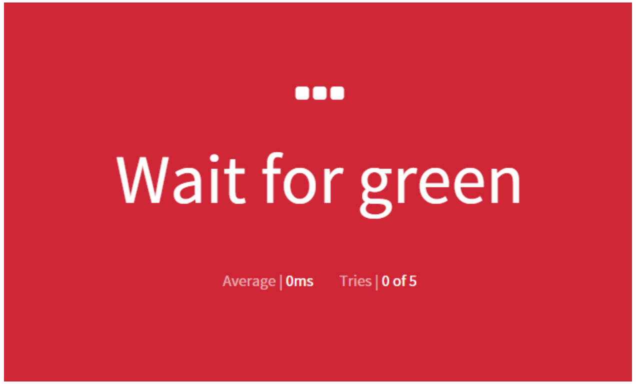
Aalto University

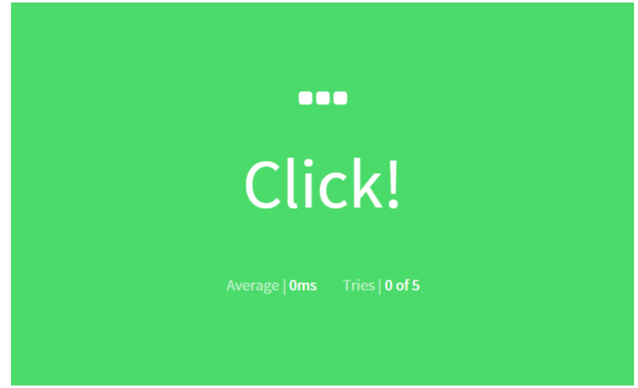
Simple reaction

Reaction times

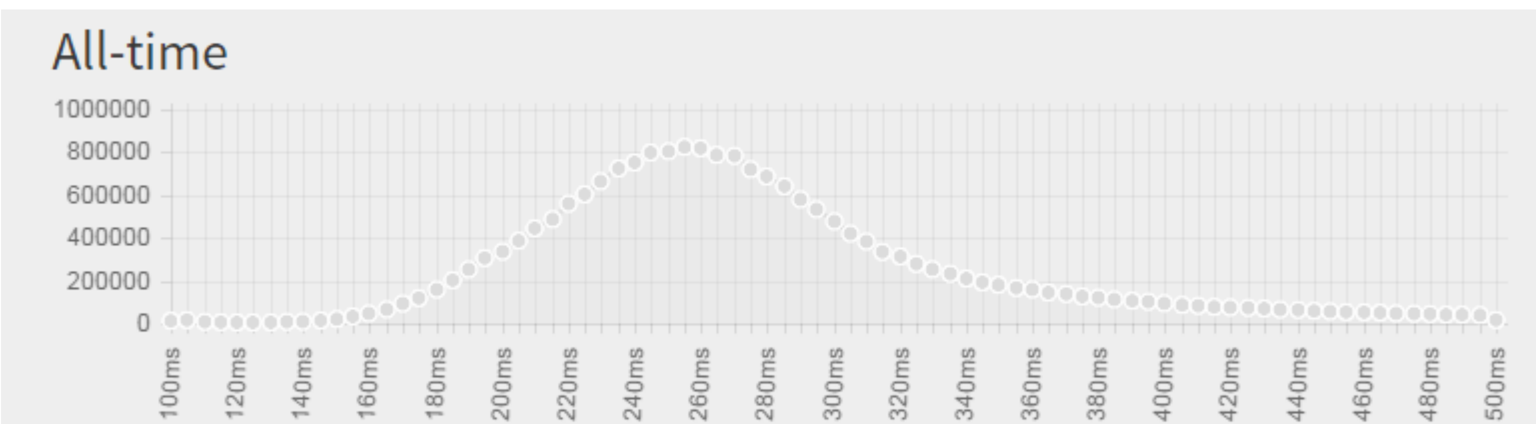


Reaction times “in the wild”



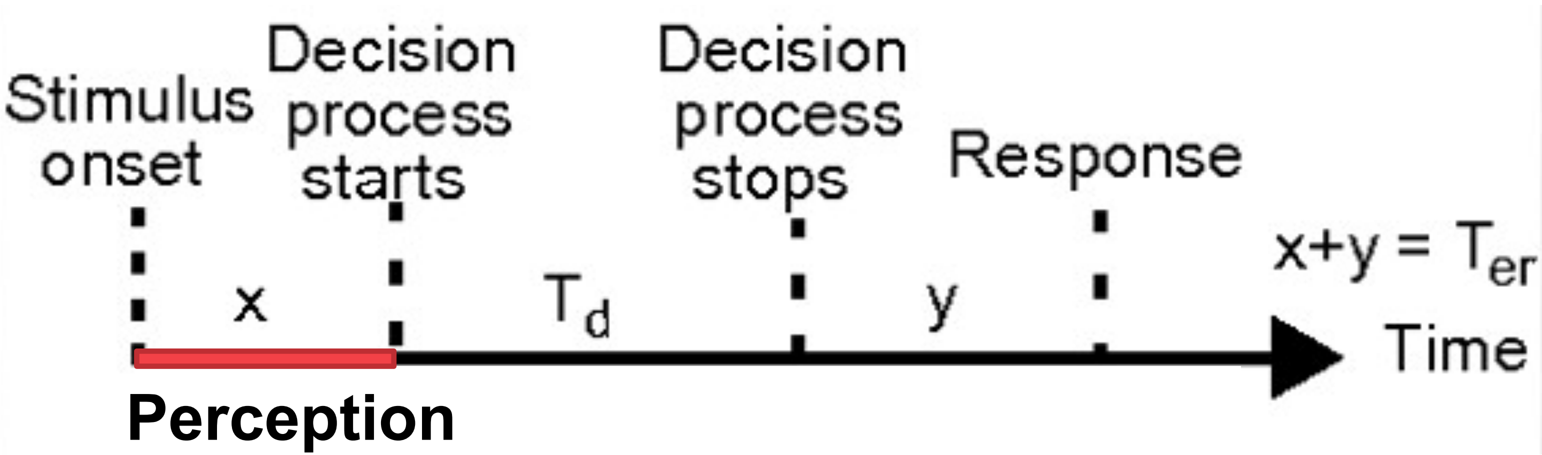


Over 22 million responses
Mean: 268 ms



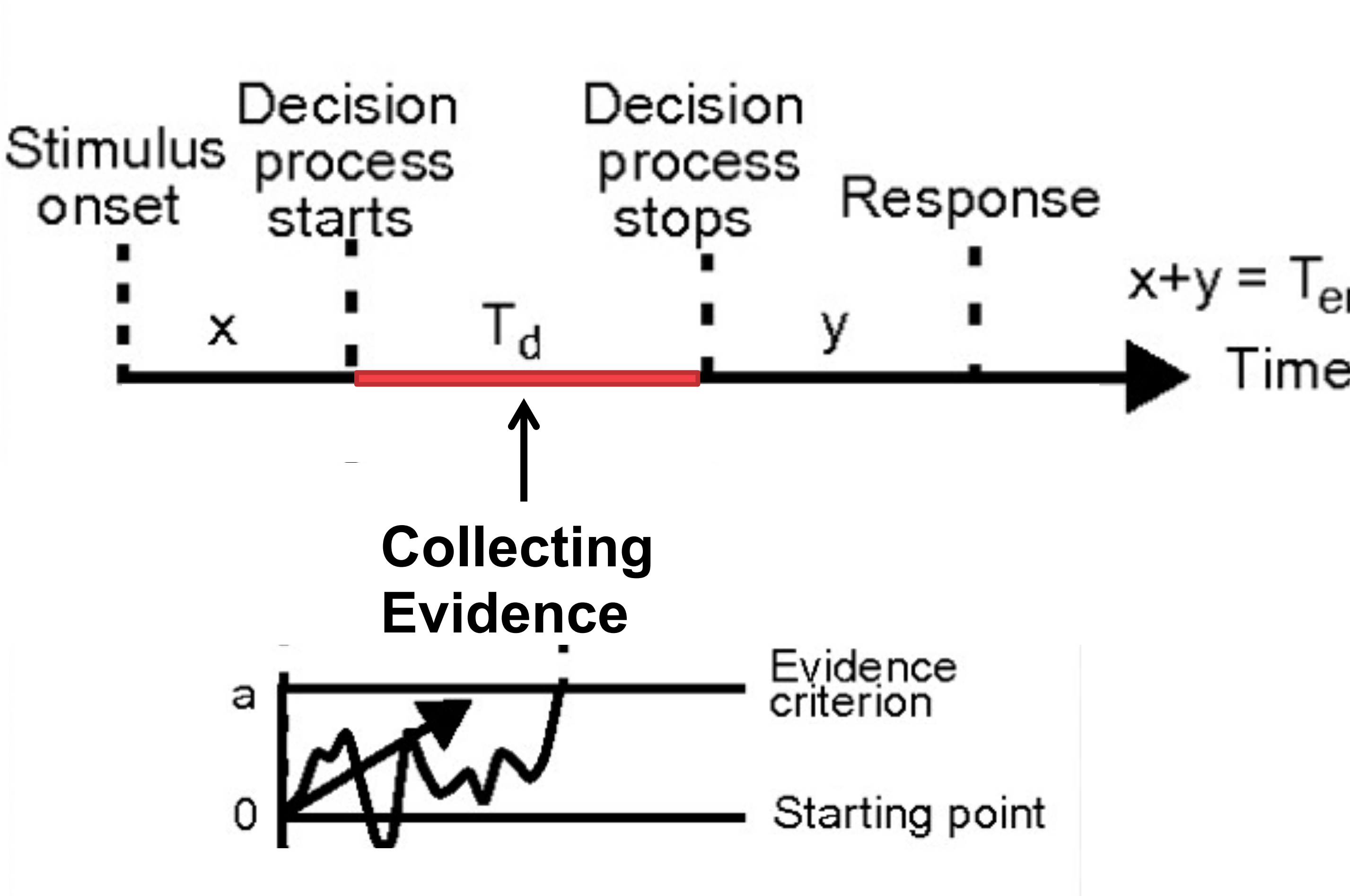
humanbenchmark.com

Ratcliff model

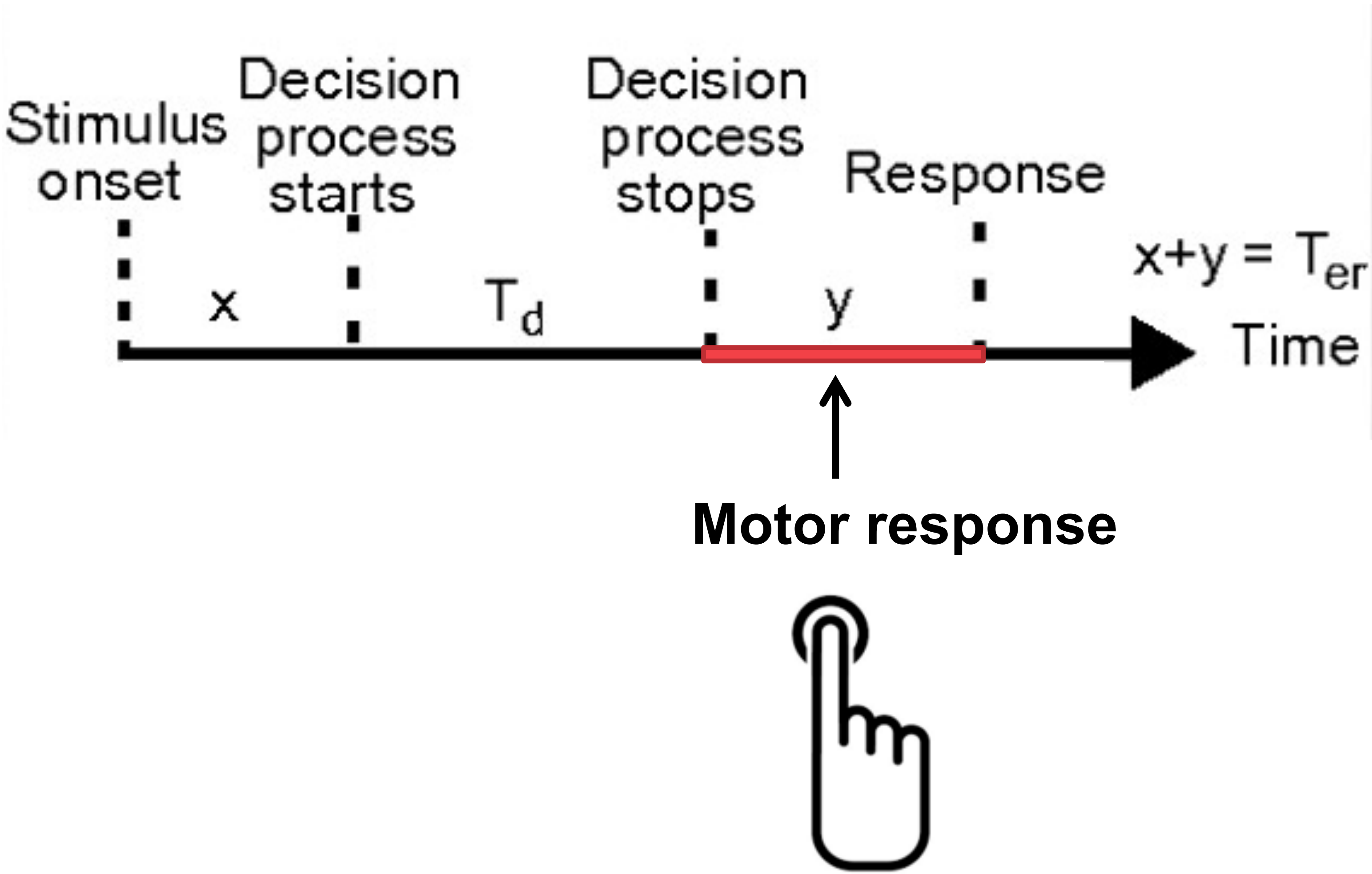


Perception and Encoding

Ratcliff model



Ratcliff model



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

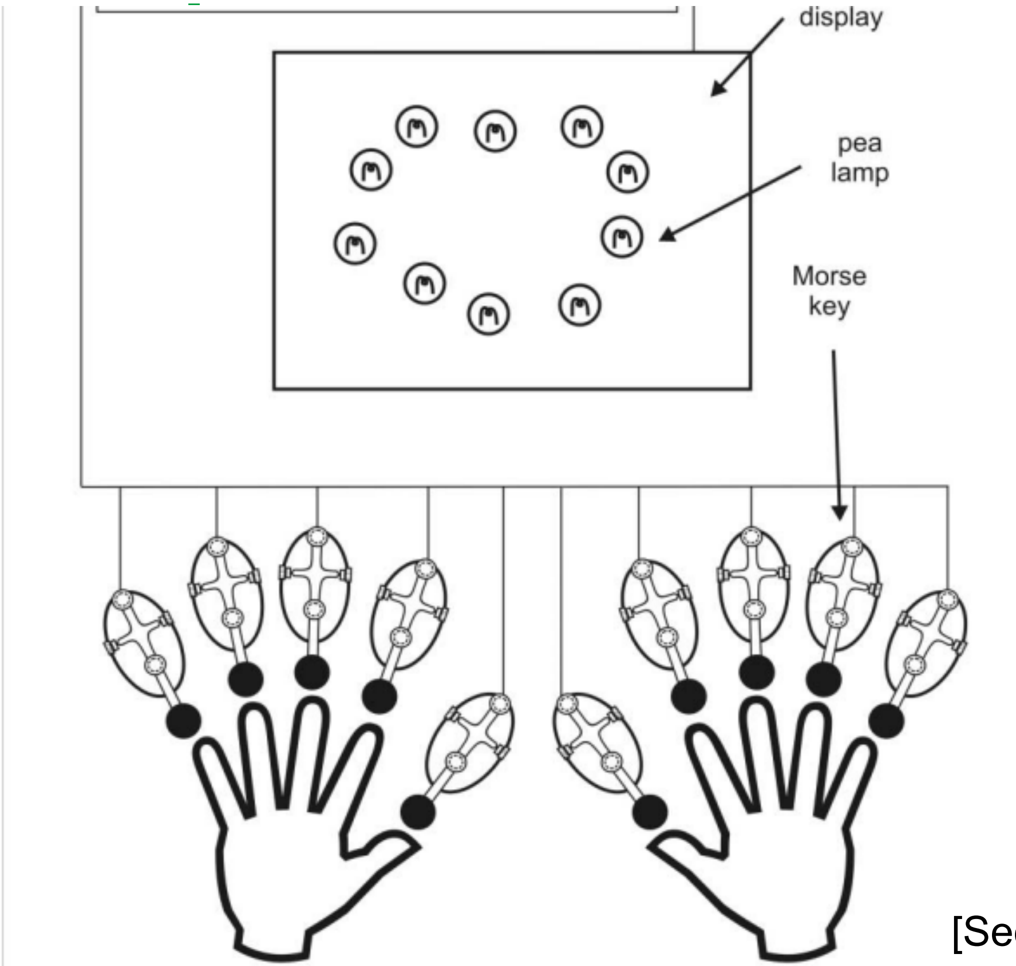
Aalto University

Choice reaction

Examples of choice reaction

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

Hick’s experiment

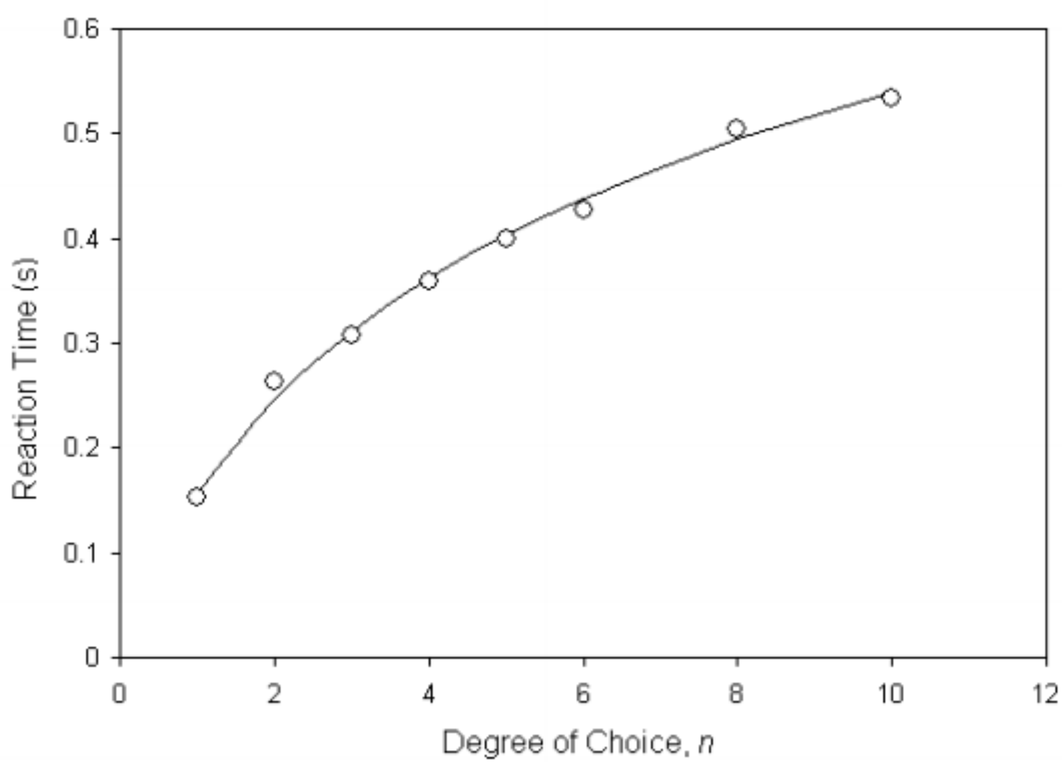


[Seow 2005]

CRT as a function of number of options

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

CRT = choice reaction time



Information-theoretical interpretation

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

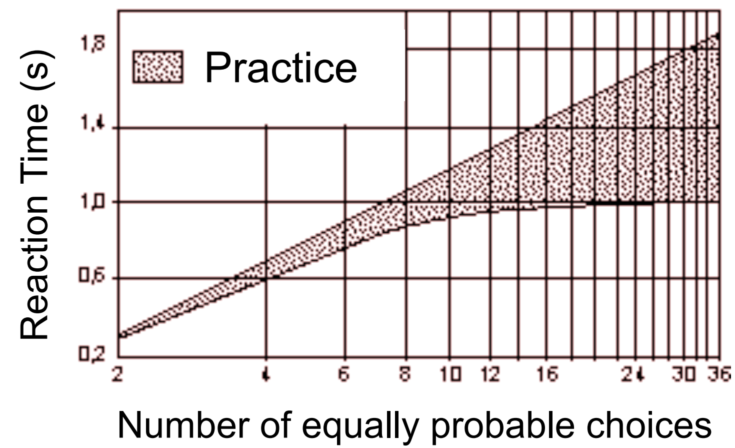
All choices have equal probability:

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

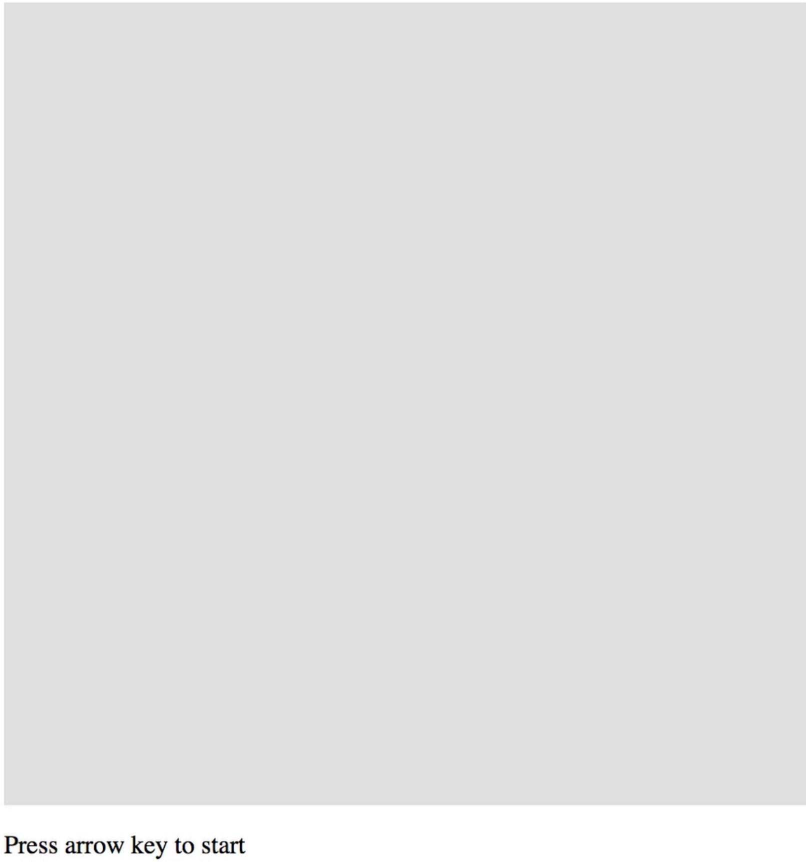
Choices have different probabilites:

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

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



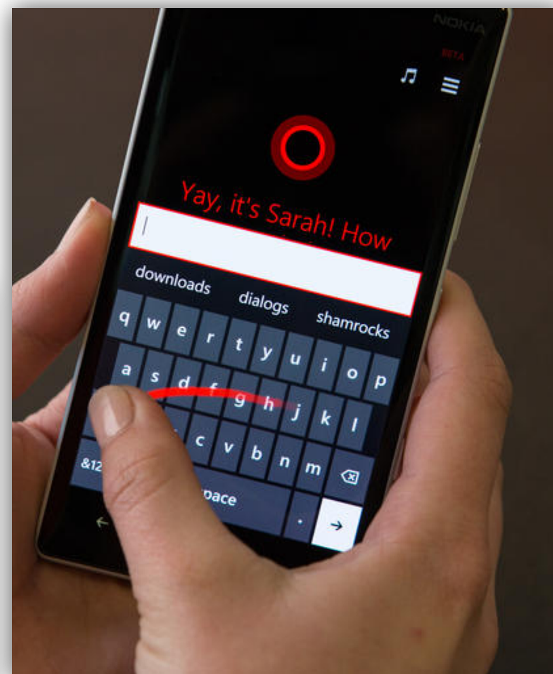
Example: Game



Aalto University

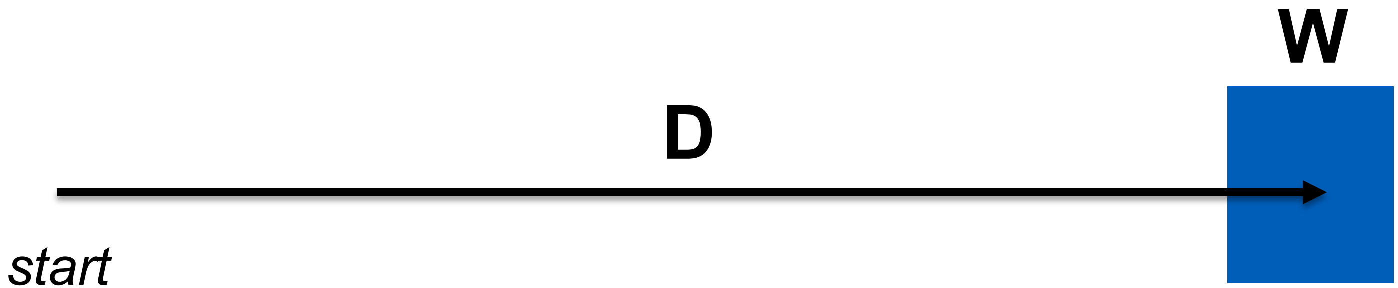
Aimed movements: Fitts’ law

Aimed Movements



Response demands in pointing

“Move as quickly as you can”



Pair work (7 mins)

Explain Fitts' law (the formula) to the class

1. **Intuition**
2. **Diagram**

You can use computers & Internet

Example: Game

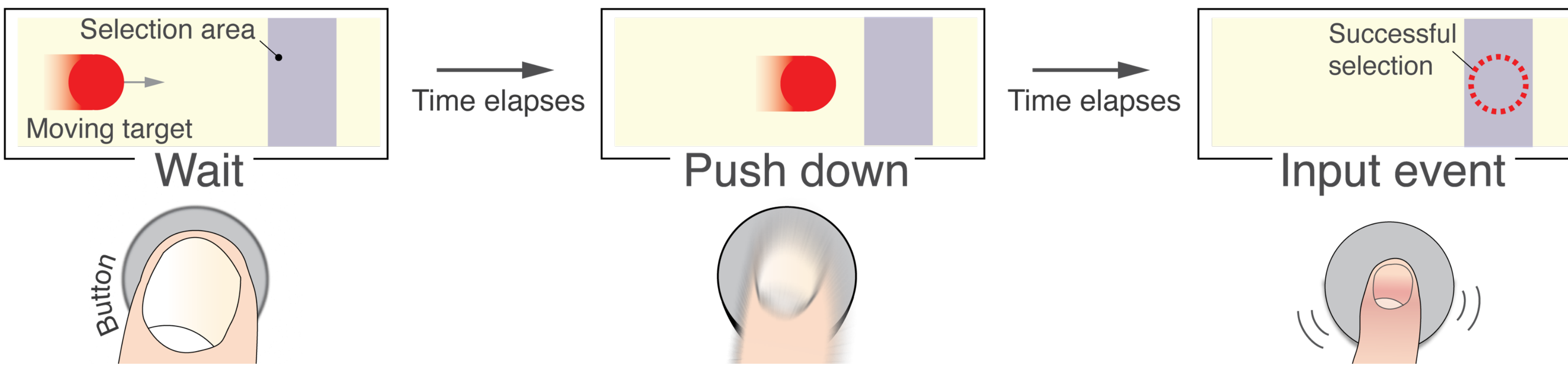
Notebook exercise

Temporal Pointing Model

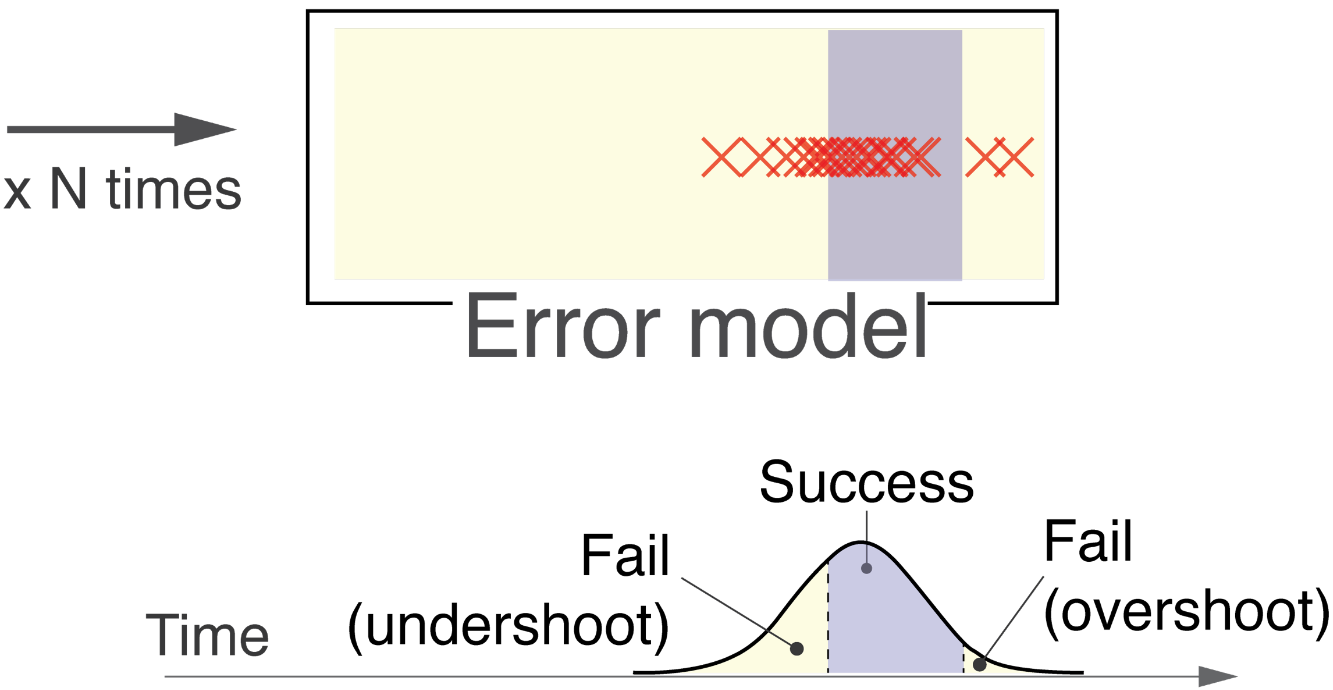
Temporal pointing task

"Press the button when the target appears under selection area"

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



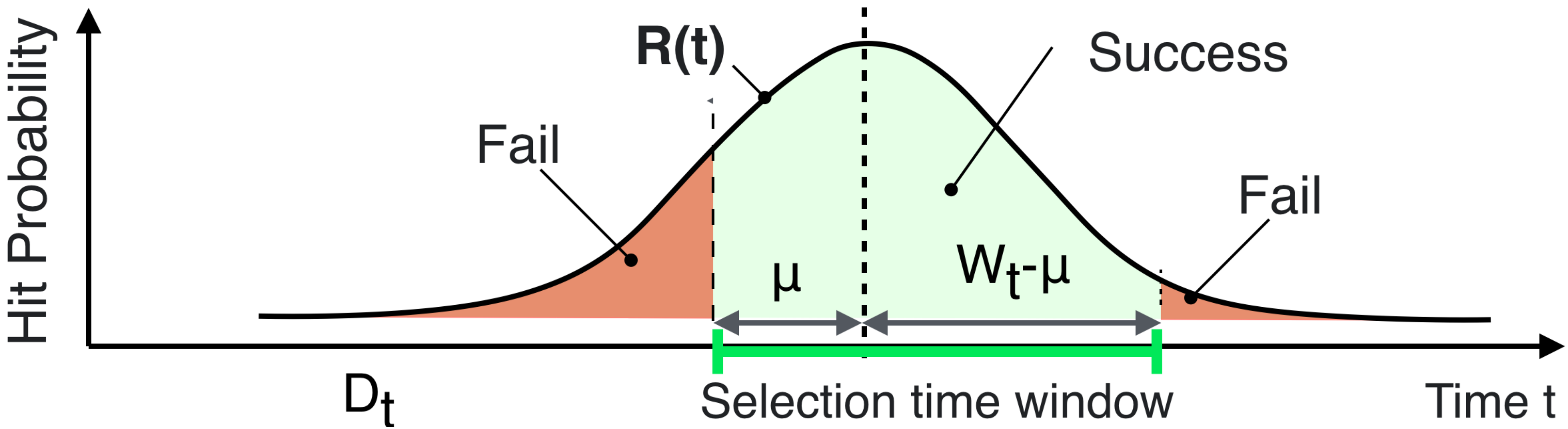
Temporal pointing model



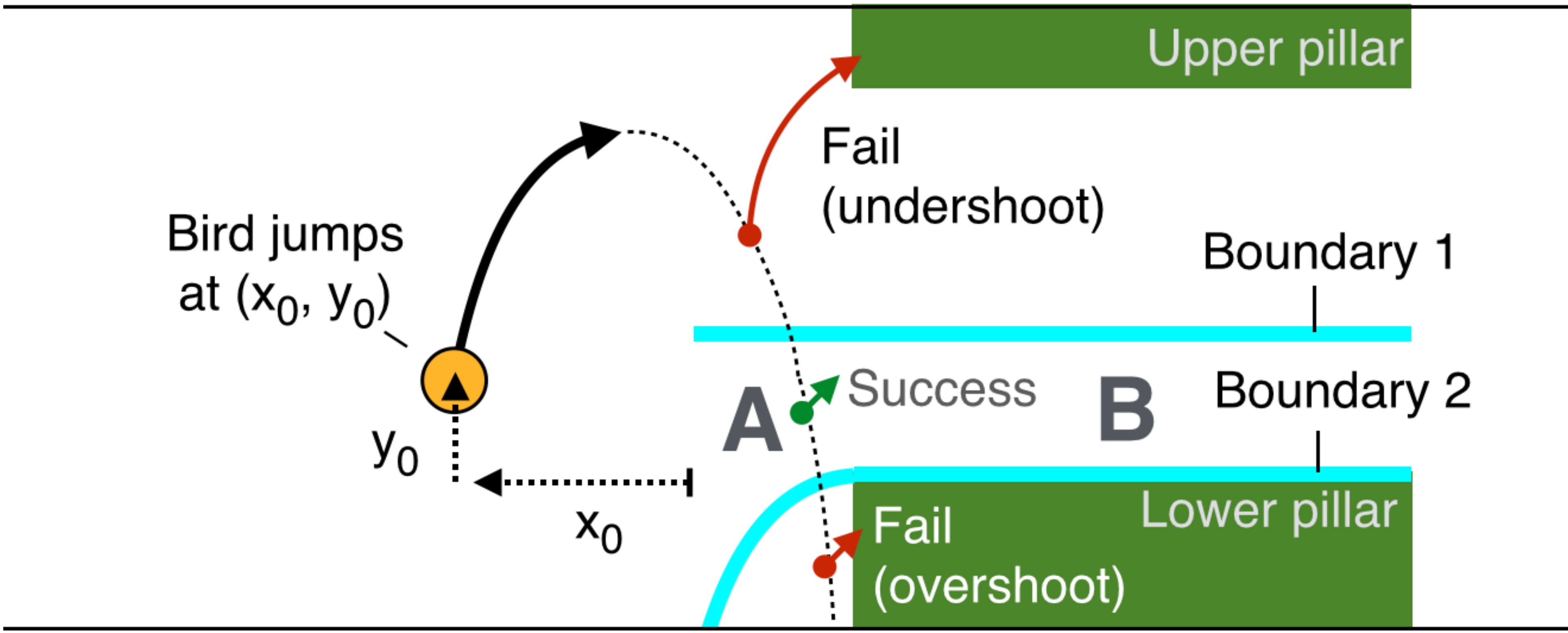
Formula for predicting error rate

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

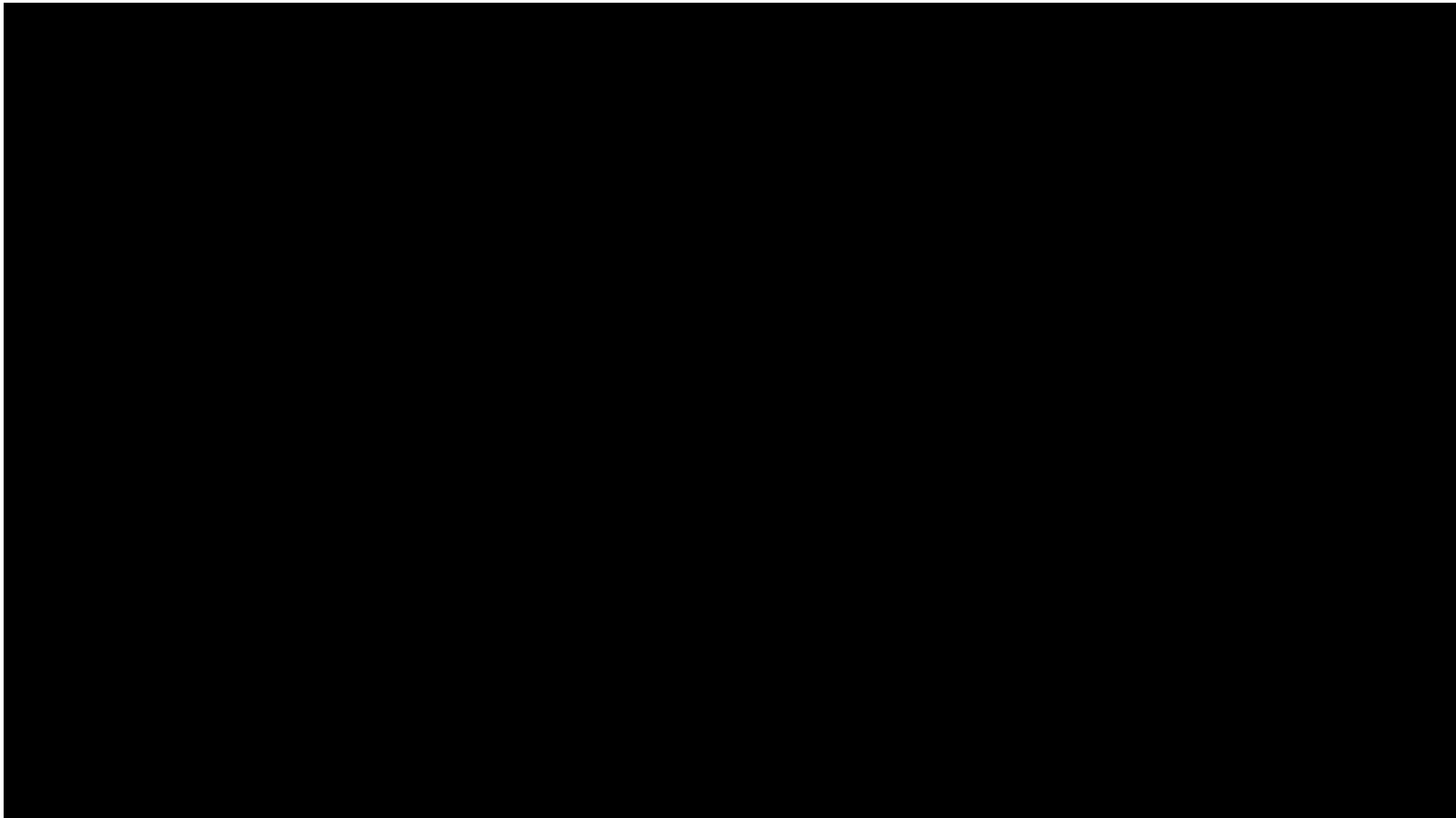
Illustrated



Example application: Flappy bird



Example application: Blinking target



2. KLM

KLM, a task-level predictive model

Pros

- Predict 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 sequential model (no multitasking or interleaving of tasks)
- Focus is on task performance, other aspects of behavior and experience are ignored
  - *Lack a notion of "semantics" and "contexts"*
  - *Overlook individual and cultural differences*
  - *Only rough notion of learning*
- Validity depends on pre-defined task specifications and model assumptions

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]

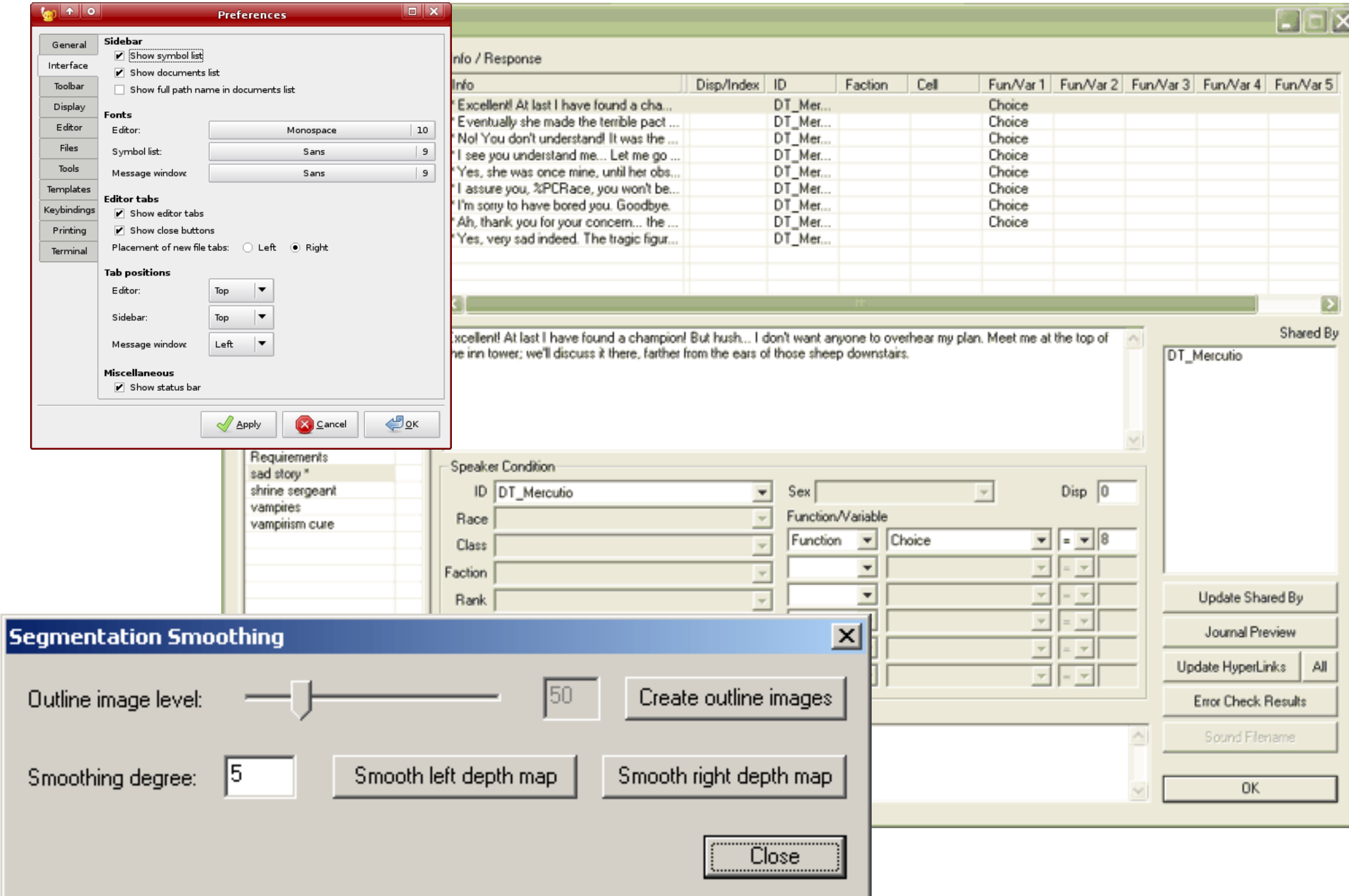
+ t_H [homing]

+ t_D [drawing]

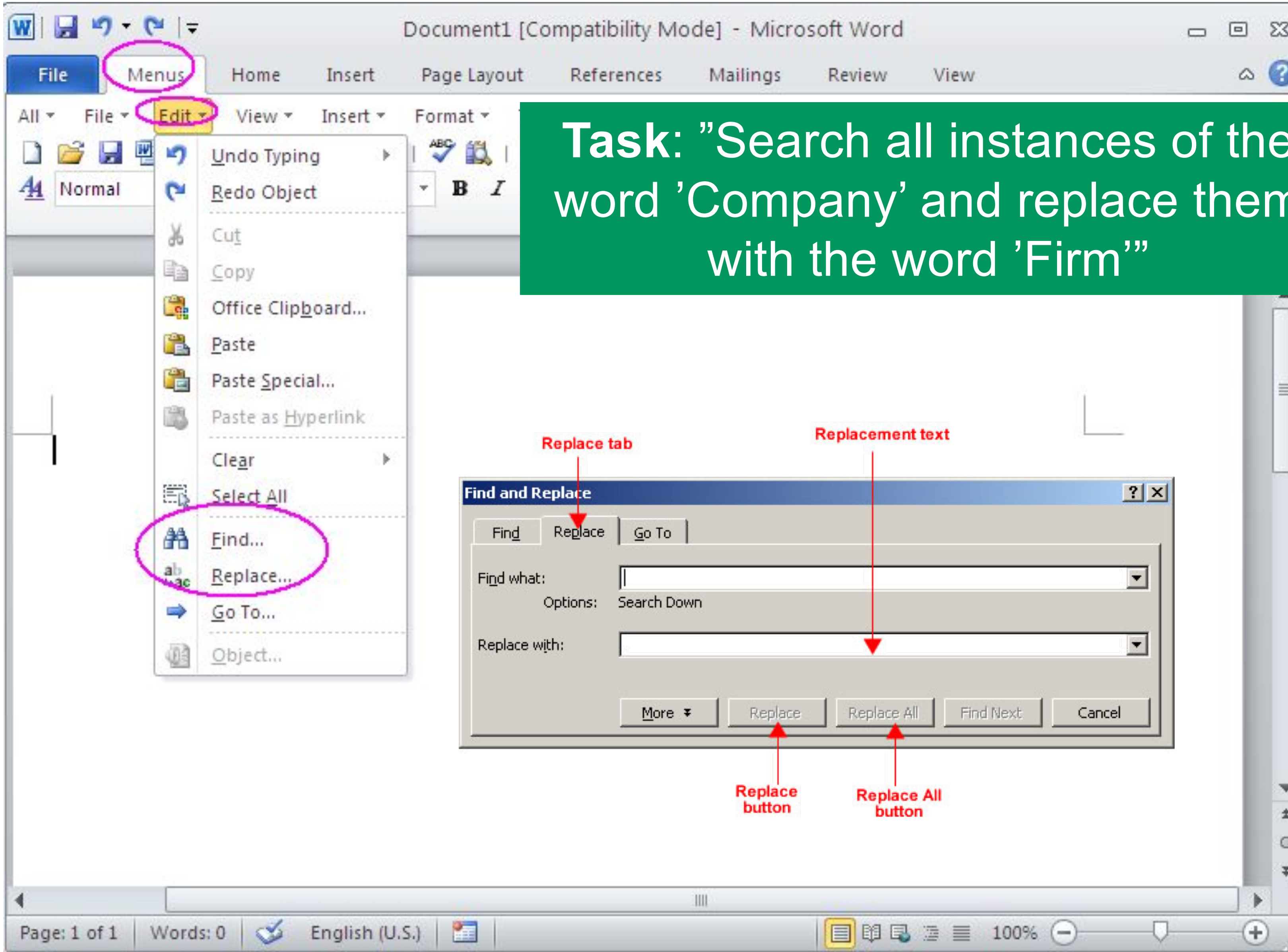
+ t_M [mental operation]

+ t_R [system response]

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**

In-class task:

1. Choose a pair
2. Take pen and paper
3. 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…

In-class 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-free (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 overlooked

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!