Human Information Processor (Card, Moran, Newell)

Perceptual Processor

Cycle time: **100** ms

Cognitive Processor

Cycle time: **70** ms

Motor Processor

Cycle time: **70** ms

It is a model – understandable by computer scientists

Predictive, but simplistic

Does not describe actual underlying mechanisms

Put it together: Do two letters have same name?

a A

Perceive first letter (representations @ VIS and WM)

Start clock (how long it takes to respond to the second symbol)

Perceive second letter (τ_P)

Recognize letter (τ_C)

Match (τ_C)

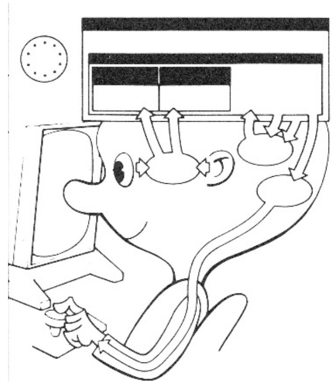
Initiate response (τ_C)

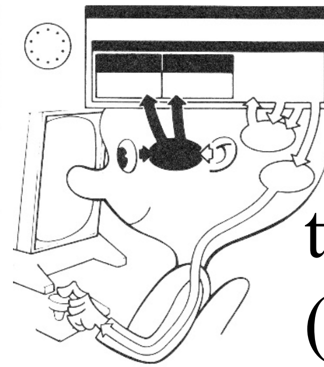
Respond (τ_M)

=> $\tau_P + 3\tau_C + \tau_M$

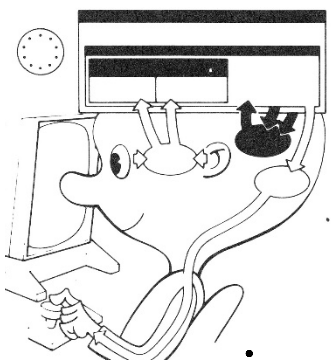
Put it together: Are A and B letters? (class match)

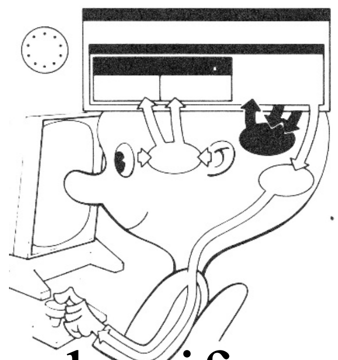
appear

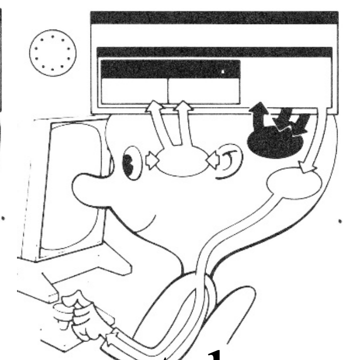


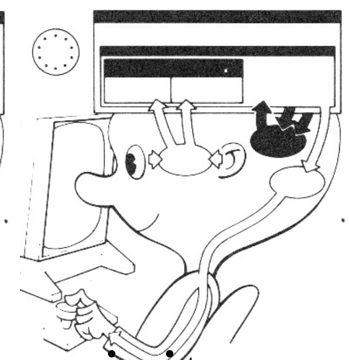


transmit to WM
(perceive it)

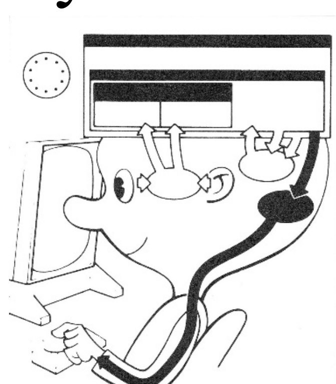








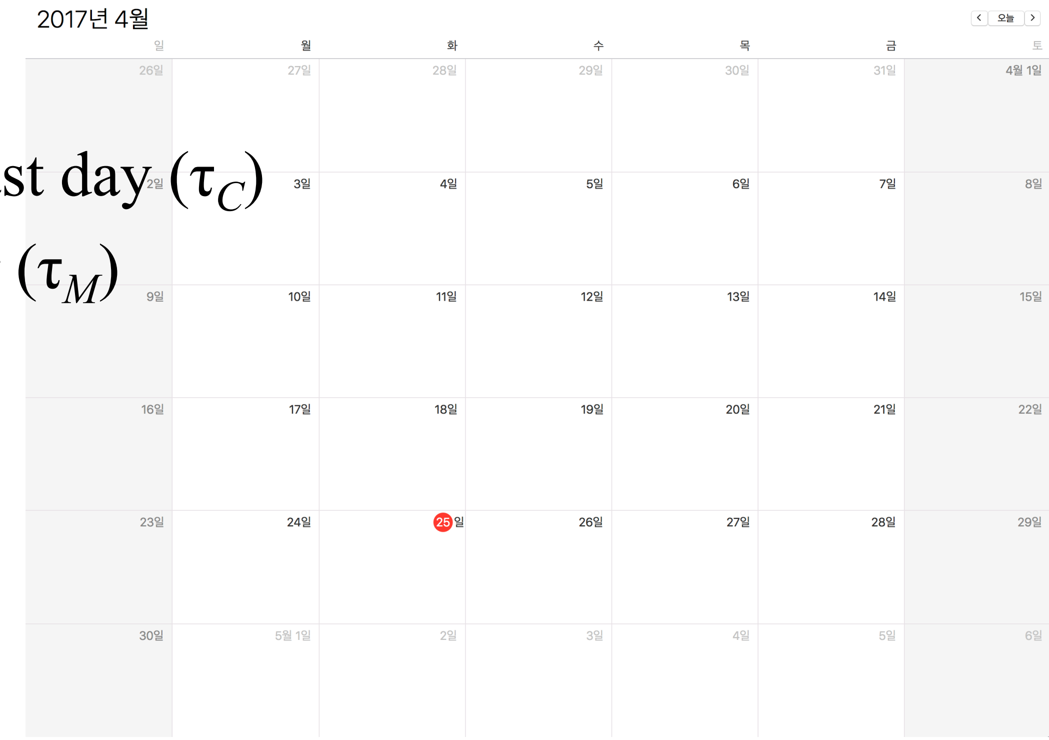
recognize classify match init response



Step		Display	VIS	WM	Hand	Elapsed Time
Example 9. Simple reaction						
	State at start of clock:					
1.	Symbol appears	α				0
2.	Transmitted to VIS		α'	α''		τ_P
3.	Initiate response		α'	α'', PUSH-YES		$\tau_P + \tau_C$
4.	Process motor command		α'	α'', PUSH-YES	PUSH-YES	$\tau_P + \tau_C + \tau_M$
Example 10. Physical match						
	State at start of clock:		α'	α''		
1.	Second symbol appears	α	α'	α''		0
2.	Transmitted to VIS		α', α'	α'', α''		τ_P
2.1.	*Match*		α', α'	α'', α'', MATCH = TRUE		$\tau_P + \tau_C$
3.	Initiate response		α'	α'', α'', PUSH-YES		$\tau_P + 2\tau_C$
4.	Process motor command			α'', α'', PUSH-YES	PUSH-YES	$\tau_P + 2\tau_C + \tau_M$
Example 11. Name match						
	State at start of clock:		α_1'	α_1'':A		
1.	Second symbol appears	α_2	α_1'	α_1'':A		0
2.	Transmitted to VIS		α_1', α_2'	α_2'', α_1'':A		τ_P
2.01.	*Recognize*		α_1', α_2'	α_2'':A, α_1'':A		$\tau_P + \tau_C$
2.1.	Match		α_1', α_2'	MATCH = TRUE		$\tau_P + 2\tau_C$
3.	Initiate response		α_2'	PUSH-YES		$\tau_P + 3\tau_C$
4.	Process motor command			PUSH-YES	PUSH-YES	$\tau_P + 3\tau_C + \tau_M$
Example 12. Class match						
	State at start of clock:		α'	α'':A:LETTER		
1.	Second symbol appears	β	α'	α'':A:LETTER		0
2.	Transmitted to WM		α', β'	β'', α'':A:LETTER		τ_P
2.01.	Recognize		α', β'	β'':B, α'':A:LETTER		$\tau_P + \tau_C$
2.02.	*Classify*		α', β'	β'':B:LETTER, α'':A:LETTER		$\tau_P + 2\tau_C$
2.1.	Match		β'	MATCH = TRUE		$\tau_P + 3\tau_C$
3.	Initiate response			PUSH-YES		$\tau_P + 4\tau_C$
4.	Process motor command			PUSH-YES	PUSH-YES	$\tau_P + 4\tau_C + \tau_M$

How many days are in April, 2017?

- Perceive calendar (τ_P)
- Recognize last day (τ_C)
- Decide to move eye to last day (τ_C)
- Move eye to last number (τ_M)
- Perceive number (τ_P)
- Recognize number(τ_C)
- Initiate response (τ_C)
- Respond (τ_M)



$$=> 2\tau_P + 4\tau_C + 2\tau_M = 2*100 + 4*70 + 2*70 = 620ms$$

Eye Movement

- Eye movement time (saccade + fixation)
  - 230 ms [70 –700ms]
  - Saccade takes 120ms

- If you move **one eye movement per phrase** while reading, and one phrase is 2.5 words, then

(60 sec/min)/(.230 sec/EM x 1/2.5 EM/word)

= 652 words / minute

So, speed readers (2,500 words / minute) don't see all words → they skim!

Putting it together: Reading Speed

- Reading speed
- What are limits?
- Which limits could be removed?

$\Rightarrow$ RSVP reader (see Firefox plugin demo)

$Tp + Tc = 100ms + 70ms$

(2.5 words / phrase) x (60 sec / min) / (.170 sec/phrase)
= 882 words / minute

Predictive Models

Predictive Models

- A predictive model is an equation
- Predicts the outcome on a *criterion variable* (aka *dependent variable* or *human response*) based on the value of one or more *predictor variables* (aka *independent variables*)
- Note: the predictor variables must be ratio-scale attributes (See **HCI:ERP** for discussion)
- Predictive models, like descriptive models, allow a problem space to be explored
- However, predictive models deal with numbers, not concepts

Why Use Predictive Models

- Card et al. presented perhaps the first predictive model in HCI.[1] In many respects, their work was straight-forward experimental research; but they went further:
  - "While these empirical results are of direct use in selecting a pointing device, it would obviously be of greater benefit if a theoretical account of the results could be made. For one thing, <u>the need for some experiments might be obviated</u>; for another, <u>ways of improving pointing performance might be suggested</u>."
- This is a call for the use of predictive models in HCI
- They went on to present predictive models using Fitts' law (which we meet shortly)

[1] Card, S. K., English, W. K., & Burr, B. J. (1978). Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics, 21*, 601-613.

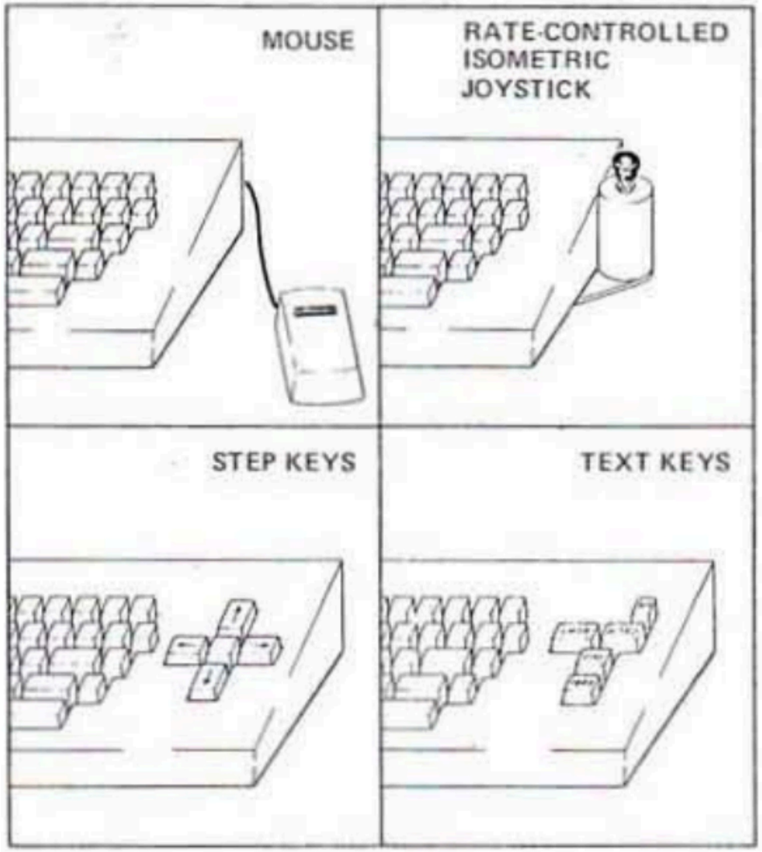


Figure 1. Pointing devices tested.

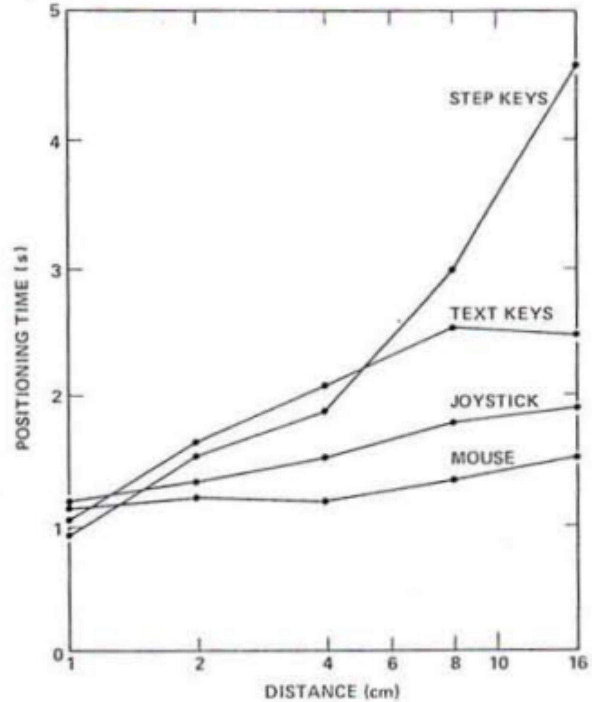
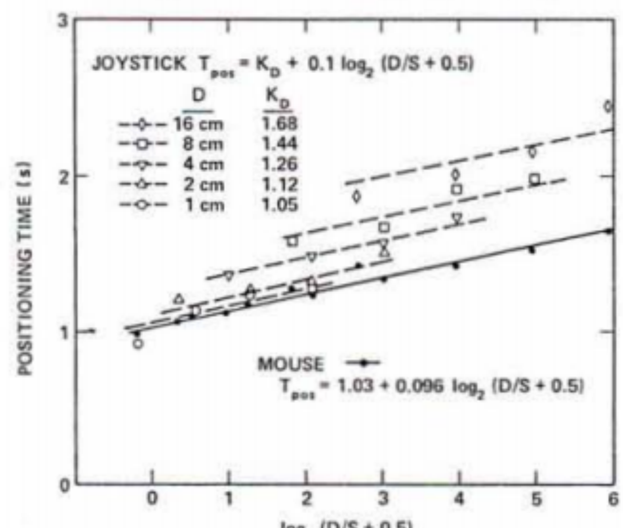


Figure 3. Effect of target distance on positioning time.



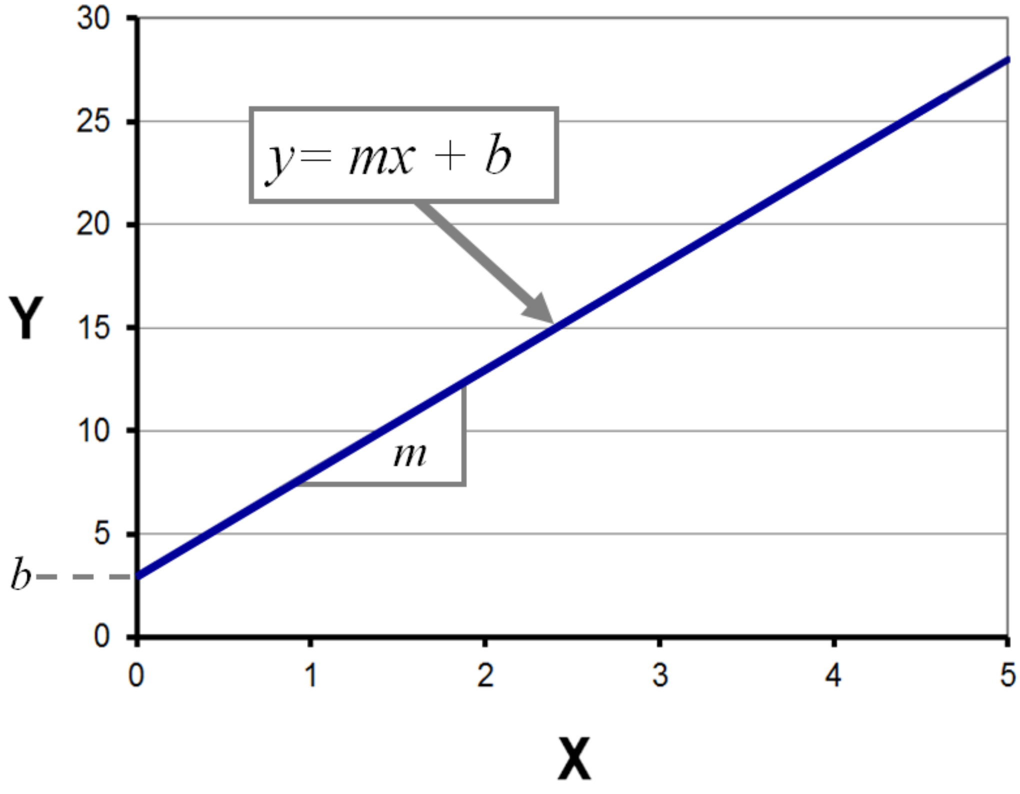
[1] Card, S. K., English, W. K., & Burr, B. J. (1978). Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics, 21*, 601-613.

Predictive Model Examples

- Linear prediction equation
- Fitts' law
- Choice reaction time
- Keystroke-level model (KLM)
- Skill acquisition
- More than one predictor

Linear Prediction Equation

- The basic prediction equation expresses a linear relationship between a predictor variable (*x*) and a criterion variable (*y*):



Linear Regression

- A linear prediction equation is built using a statistical procedure know as *linear regression*
- Goal:
  - Given a set of x-y sample points, find the coefficients m and b (previous slide) for the line that minimizes the squared distances (*least squares*) of the points from the line
- The result is a prediction equation that gives the best estimate of y in terms of x
- The assumption, of course, is that the relationship is linear
- Want the details? Just enter "linear regression" or "least squares" into Google or Wikipedia

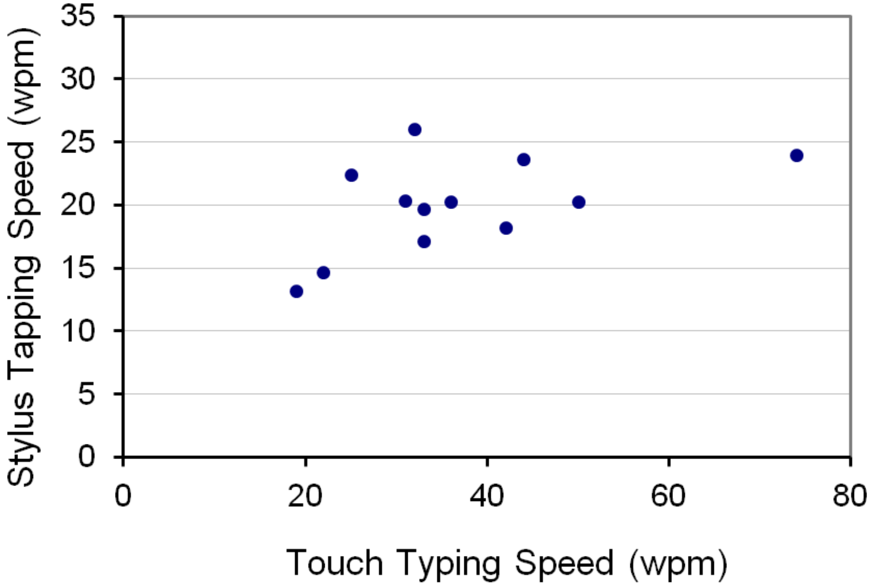
Example

- A research project investigated text entry on soft keyboards[1]
- The research also asked…
  - *Can stylus **tapping** entry speed be predicted from touch **typing** entry speed?*
- Touch typing speed is the predictor variable (*x* - measured in a pre-test)
- Stylus typing speed is the criterion variable (*y* - measured experimentally)
- Data and scatter plot

[1] MacKenzie, I. S., & Zhang, S. X. (2001). An empirical investigation of the novice experience with soft keyboards. *Behaviour & Information Technology, 20*, 411-418.

Data and Scatter Plot

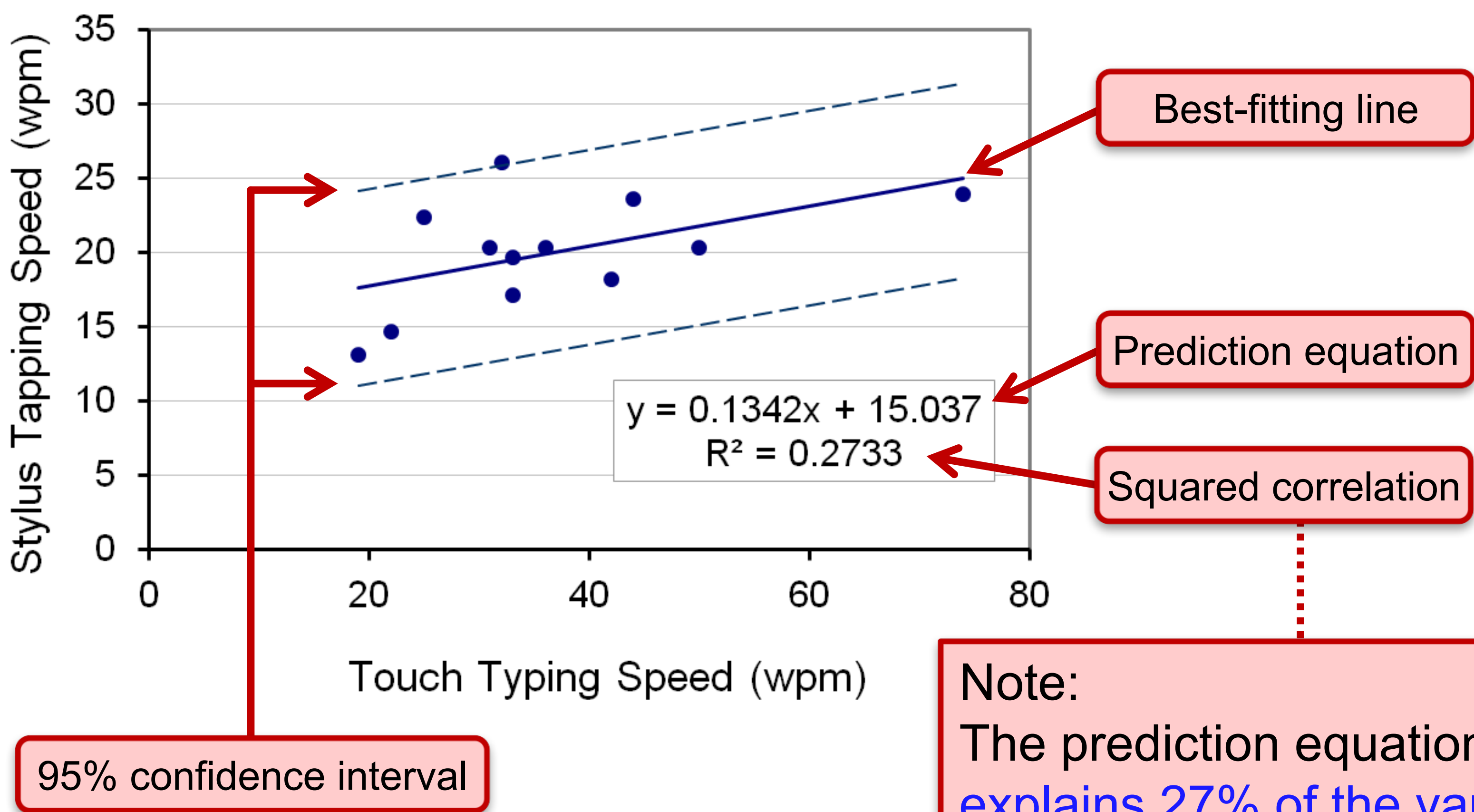
Participant	Stylus Tapping Speed (wpm)	Touch Typing Speed (wpm)
P1	18.2	42
P2	23.6	44
P3	26.0	32
P4	20.3	50
P5	20.3	36
P6	17.1	33
P7	24.0	74
P8	14.7	22
P9	20.3	31
P10	19.7	33
P11	22.4	25
P12	13.1	19



There seems to be a relationship: Faster touch typists seem to be faster at stylus tapping.

Questions:
What is the prediction equation?
How strong is the relationship?

Prediction Equation



Note:
The prediction equation explains 27% of the variation in the data – a modest predictor, at best.

Predictive Model Examples

- Linear prediction equation
- Fitts' law
- Choice reaction time
- Keystroke-level model (KLM)
- Skill acquisition
- More than one predictor

Fitts' Law

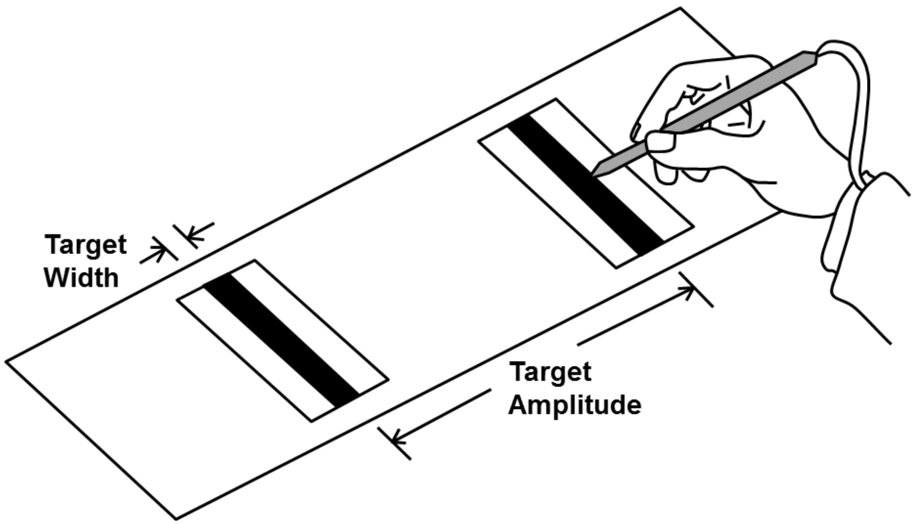
- One of the most widely used models in HCI
- Model for rapid aimed movements (e.g., moving a cursor toward a target and selecting the target)
- Three applications:
  1. Use a Fitts' law prediction equation to analyse and compare design alternatives
  2. Use Fitts' *index of performance* (now *throughput*) as a dependent variable in a comparative evaluation (use IM)
  3. Determine if a device or technique "conforms to Fitts' law"
- Origins: Two highly-cited papers in experimental psychology, one from 1954[1], the other from 1964[2]

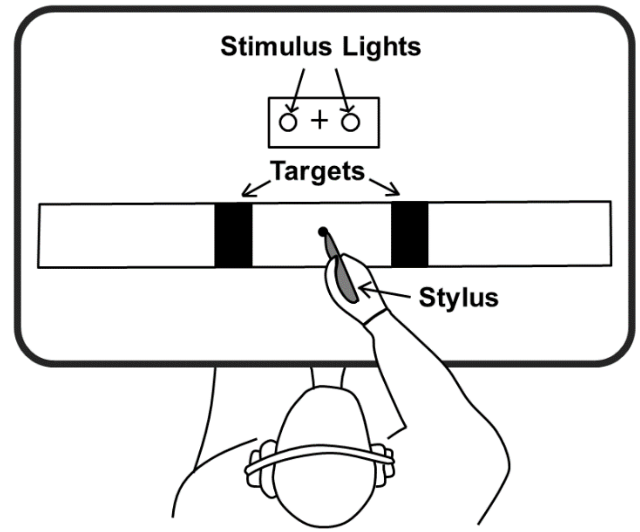
Predictive model

[1] Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology, 47*, 381-391.

[2] Fitts, P. M., & Peterson, J. R. (1964). Information capacity of discrete motor responses. *Journal of Experimental Psychology, 67*, 103-112.

Fitts' Law – Task Paradigms





Serial task **Discrete task**

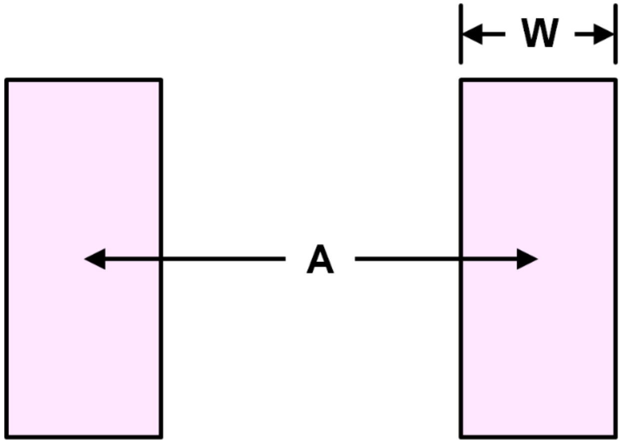
These sketches were adapted from Fitts' 1954 and 1964 papers. It is easy to imagine comparable tasks implemented on computing technology.

Fitts' Index of Difficulty (*ID*)

- Fitts' index of difficulty (*ID*) is a measure of the difficulty of a target selection task:

$$\boxed{ID = \log_2\left(\frac{A}{W} + 1\right)}$$

Units: bits



- Normally the prediction equation is built using the *effective* index of difficulty (ID_e) – includes an "adjustment for accuracy" (see **HCI:ERP** for discussion)
- Fitts hypothesized that the relationship between movement time (*MT*) and *ID* is linear

Fitts' Law Models for Pointing Devices

- A research project compared four pointing devices, including two for remote pointing[1]
- Twelve participants performed a series of serial target selection tasks using the four devices
- For our purpose, we'll look at the data and models for two of the devices:

Interlink *RemotePoint*

Microsoft *Mouse 2.0*

[1] MacKenzie, I. S., & Jusoh, S. (2001). An evaluation of two input devices for remote pointing. *Proceedings - EHCI 2000*, 235-249, Heidelberg, Germany: Springer-Verlag.

Experiment Conditions and Observations

Conditions | Mouse Observations | RemotePoint Observations

A (pixels)	W (pixels)	ID (bits)	Mouse				RemotePoint			
			W_e (pixels)	ID_e (bits)	MT (ms)	TP (bits/s)	W_e (pixels)	ID_e (bits)	MT (ms)	TP (bits/s)
40	10	2.32	11.23	2.19	665	3.29	13.59	1.98	1587	1.25
40	20	1.58	19.46	1.61	501	3.21	21.66	1.51	1293	1.17
40	40	1.00	40.20	1.00	361	2.76	37.92	1.04	1001	1.04
80	10	3.17	10.28	3.13	762	4.11	10.08	3.16	1874	1.69
80	20	2.32	18.72	2.40	604	3.97	25.21	2.06	1442	1.43
80	40	1.58	35.67	1.70	481	3.53	37.75	1.64	1175	1.40
160	10	4.09	10.71	3.99	979	4.08	10.33	4.04	2353	1.72
160	20	3.17	21.04	3.11	823	3.77	19.09	3.23	1788	1.81
160	40	2.32	41.96	2.27	615	3.69	35.97	2.45	1480	1.65
		Mean	23.25	2.38	644	3.60	23.51	2.35	1555	1.46

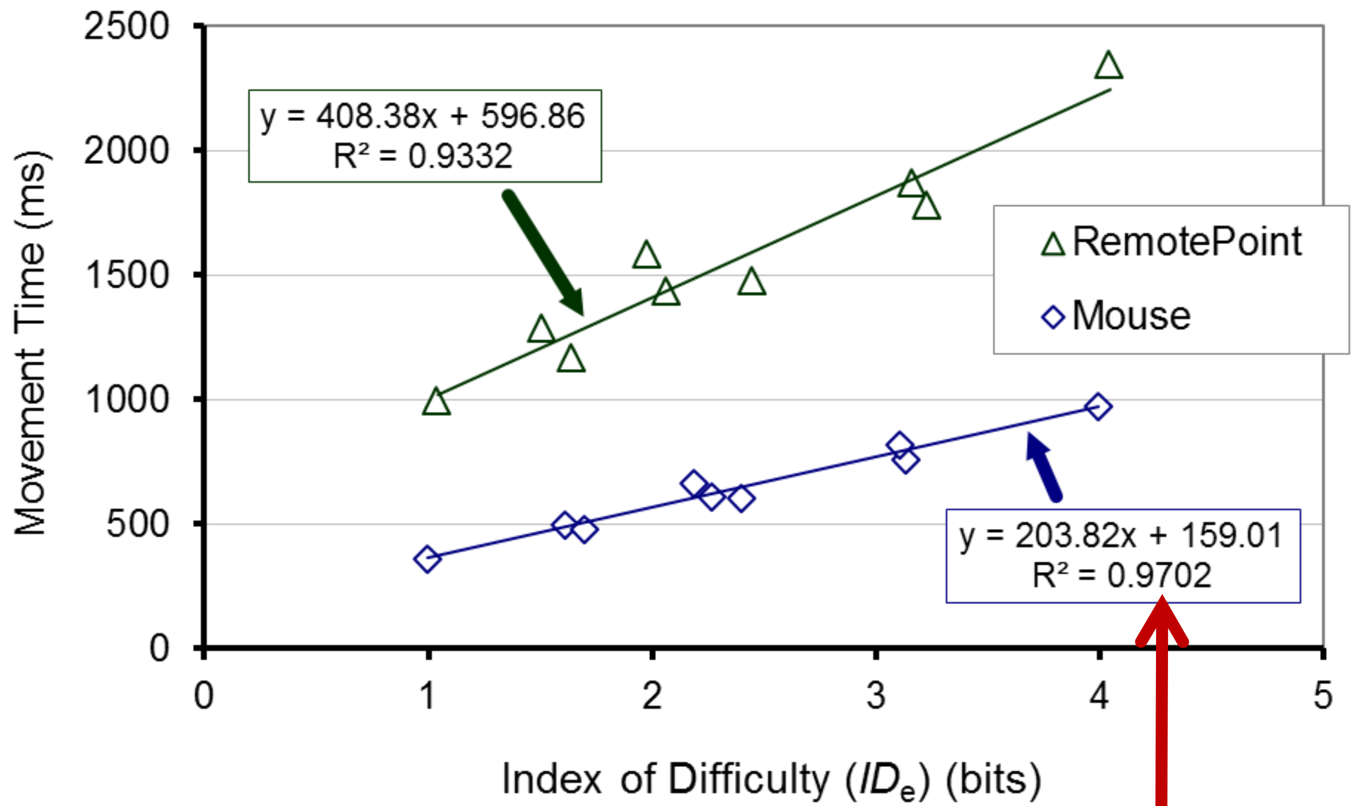
For model building…

Mouse: *x* sample points (ID_e), *y* sample points (MT)

RemotePoint: *x* sample points (ID_e), *y* sample points (MT)

- effective target width: $W_e = 4.133 \times SD_x$
- SD_x : standard deviation in the selection coordinates gathered over a sequence of trials for a particular D-W condition
- TP (or IP) : index pf performance (throughput), ID_e/MT

Fitts' Law Prediction Equations



Squared correlations are very high.
Yes, the *MT-ID* relationship is linear!

Predictive Model Examples

- Linear prediction equation
- Fitts' law
- Choice reaction time
- Keystroke-level model (KLM)
- Skill acquisition
- More than one predictor

Choice Reaction Time

- Given n stimuli, associated one-for-one with n responses, the **time to react to the onset of a stimulus** is the *choice reaction time*
- Modeled by the Hick-Hyman law:[1 2]

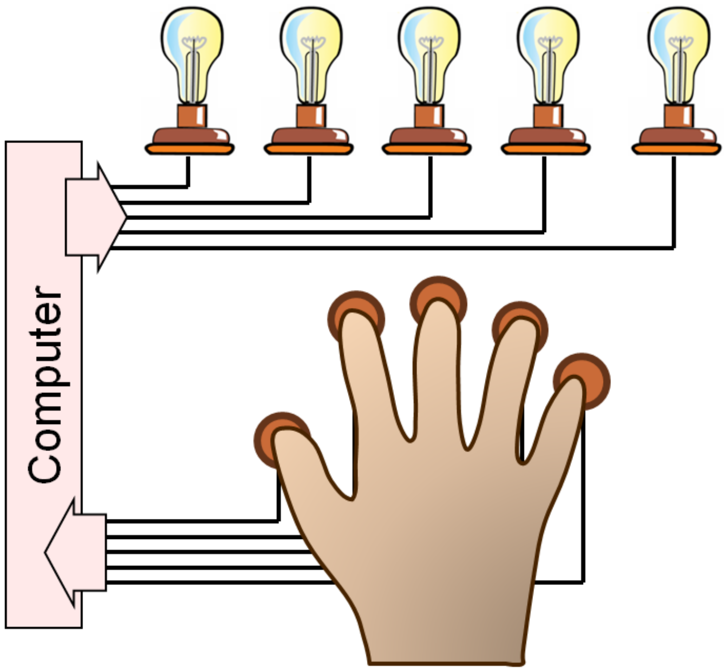
$$\boxed{RT = a + b \log_2(n + 1)}$$

- Coefficients:

  $a \approx 200$ ms

  $b \approx 150$ ms/bit

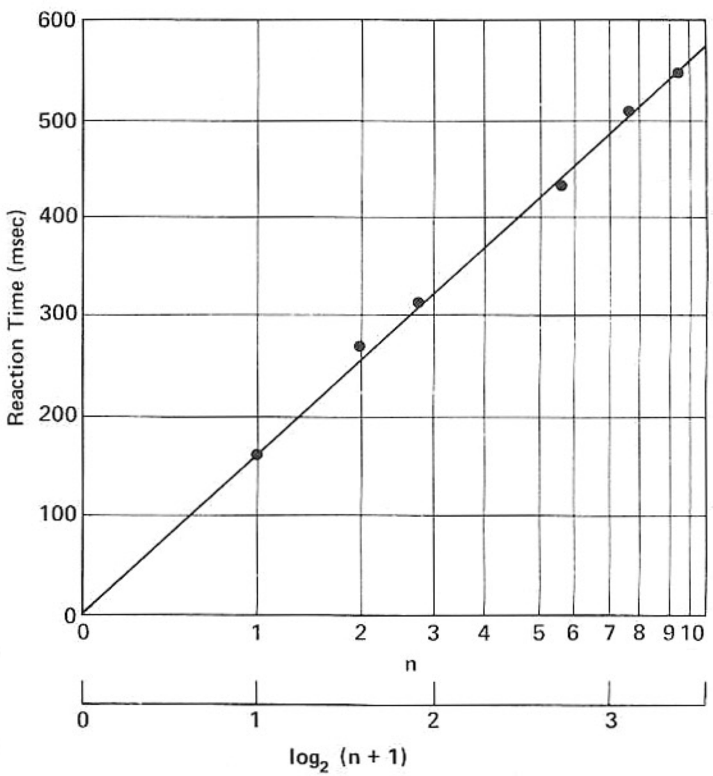
- An Information processing model (like Fitts' law)

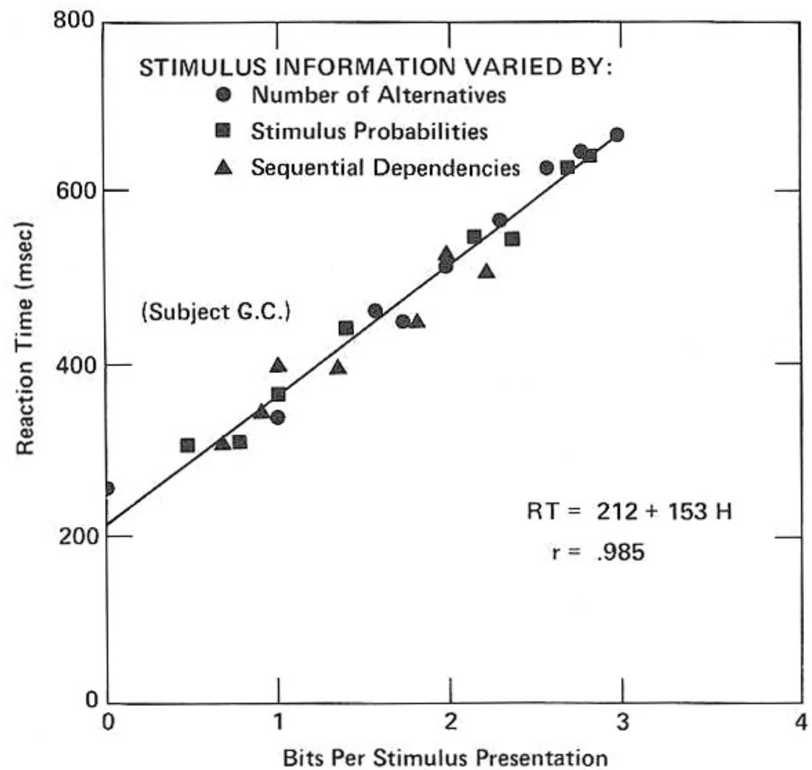


[1] Hick, W. E. (1952). On the rate of gain of information. *Quarterly J Exp Psychol, 4*, 11-36.
[2] Hyman, R. (1953).Stimulus information as a determinant of reaction time. *J Exp Psychol, 45*, 188-196.

Hick-Hyman law

- Choice Reaction time
  - Time to choose one out of **n** actions : RT=a + b H
- log rather than linear





Hick-Hyman law

- An Information Processing Model (like Fitts' law)
- H: information content of a task (# of bits)
  - information-theoretic entropy of the decision
  - uncertainty about whether to respond or not (uncertainty principle)
- H=$\sum_i p_i \log_2\left(\frac{1}{p_i}+1\right)$
  - p_i: probability of occurrence of the i-th item in the set
- When p_i is $\frac{1}{n}$ for all i, H= $\log_2(n+1)$

- Application example
  - one menu of eight item vs. two menus of four items
  - http://www.lap.umd.edu/poms/ (The Psychology of Menu Selection)

Predictive Model Examples

- Linear prediction equation
- Fitts' law
- Choice reaction time
- Keystroke-level model (KLM)
- Skill acquisition
- More than one predictor

Skill Acquisition

- When learning a skill, we begin as *novices*
- Initial performance is poor, but, with practice, we acquire skill
- With continued practice, we become proficient, perhaps *experts*
- The novice to expert transition is well suited to predictive modeling
- Dependent variable → proficiency (typically, the time or speed in doing a task)
- Independent variable → amount of practice (e.g., hours, days, months, blocks, sessions)

Power Law of Learning

- Relationship between proficiency and practice is non-linear:
  - At first, a small amount of practice yields substantial improvement
  - Later, the same "small amount of practice" yields only a slight improvement
- Relationship best expressed by a power function:

$$\boxed{y = b \times x^{a}} \qquad \boxed{T_n = T_1 \times n^{a}}$$

(general form) (power law of learning)

where…
T_n → time to do the task on the n^{th} trial
T_1 → time to do the task on the 1st trial (a constant)
n → trial indicator (e.g., hours, days, blocks, sessions)
a → a constant setting the shape of the curve

Note: *a* is negative since task completion time *decreases* with practice

Power Law of Practice

- The time T_n to perform a task on the n-th trial follows a power law

$$T_n = T_1 n^{-\alpha}$$

- The time to do a task decreases with practice
- The rate of decrease is proportional to a power of the amount of practice

- Typical values for α are [.2~.6]

Speed Variation

- Dependent variable can be speed (*S*), the reciprocal of time
- Model predicts "tasks per unit time" (e.g., words per minute)
- Mathematical form:

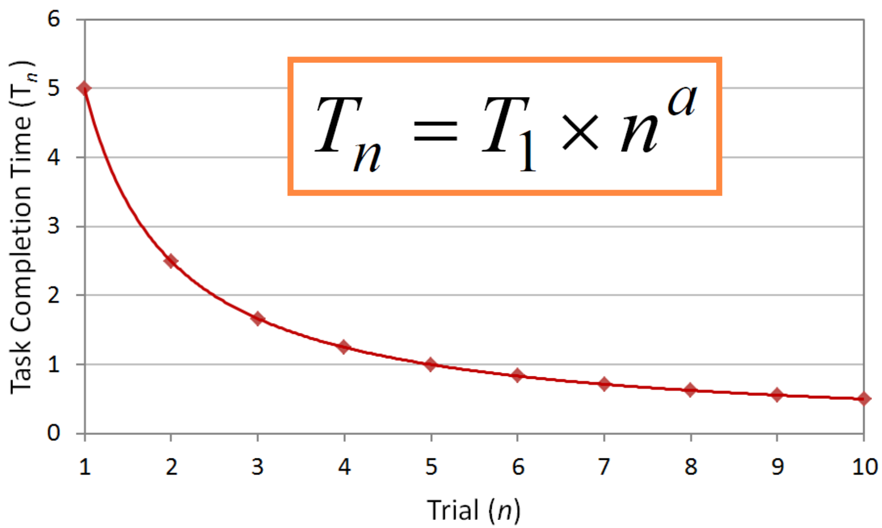
$$\boxed{S_n = S_1 \times n^a}$$

where…
S_n → speed on the n^{th} trial
S_1 → speed on the 1st trial (a constant)
n → trial indicator (e.g., hours, days, blocks, sessions)
a → a constant setting the shape of the curve

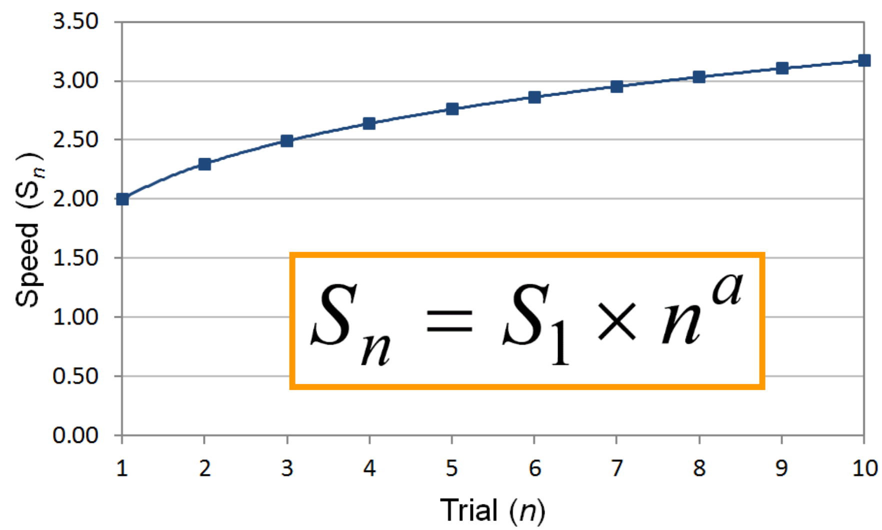
Note: a is positive and <1 reflecting the diminishing return with practice

Curve Shapes

Predicting time



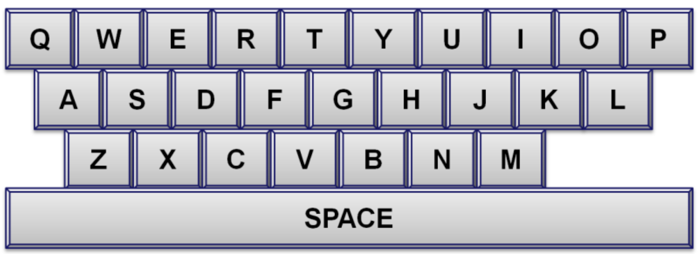
Predicting speed



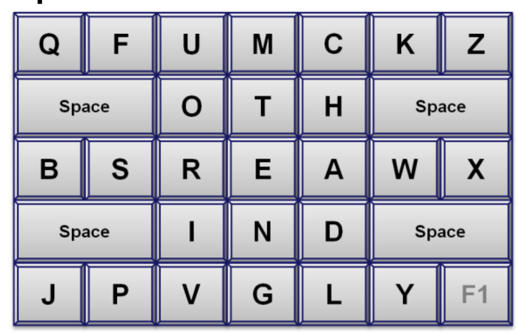
Example

- An experiment compared two soft keyboards for text entry[1]
  - Qwerty → conventional letter arrangement
  - Opti → optimized to minimize finger or stylus movement

Qwerty



Opti



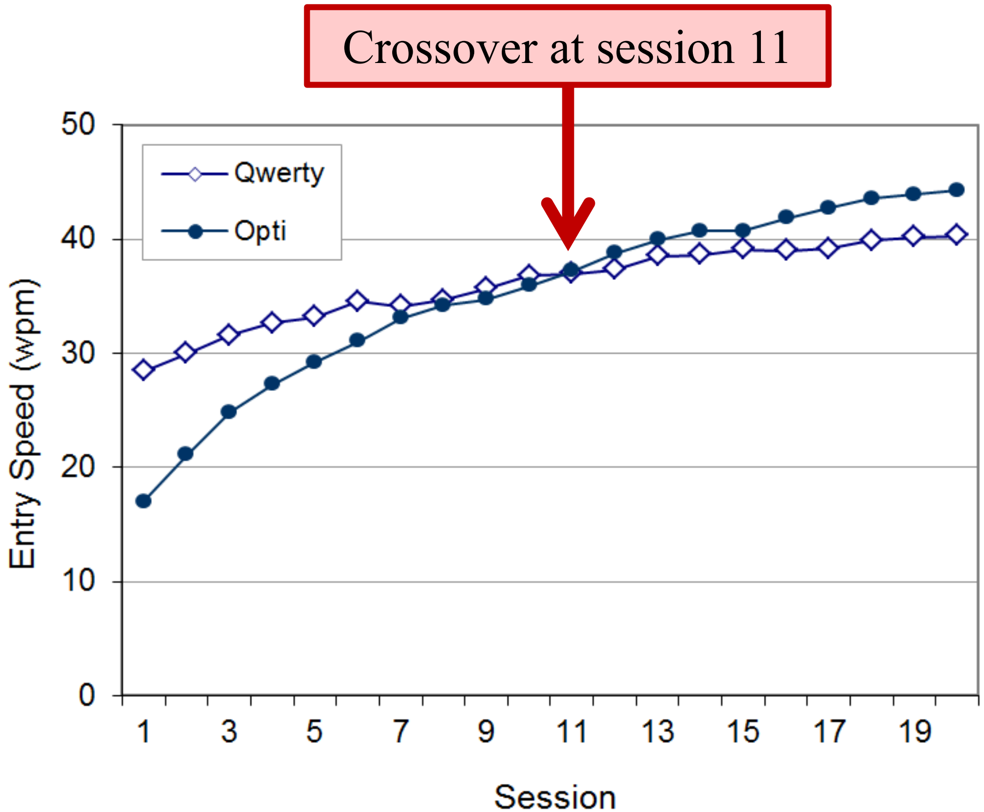
- Qwerty expected to be faster initially
- Opti expected to be faster with practice
- Participants performed 20 sessions of text entry; results →

[1] MacKenzie, I. S., & Zhang, S. X. (1999). The design and evaluation of a high-performance soft keyboard. *Proc CHI '99*, 25-31, New York: ACM.

Results

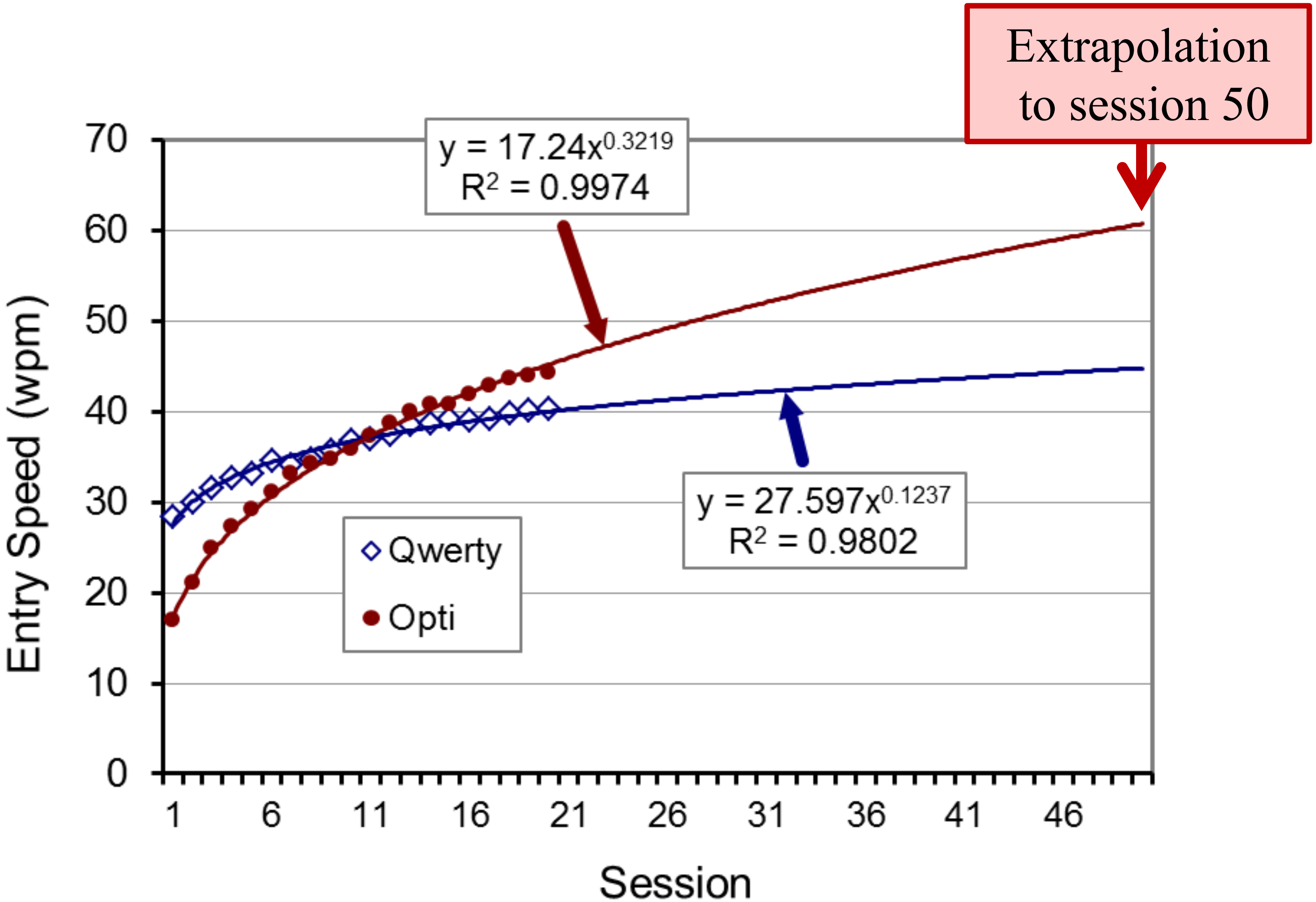
Entry Speed (wpm)

Session	Qwerty	Opti
1	28.44	16.98
2	29.98	21.06
3	31.56	24.80
4	32.62	27.26
5	33.20	29.18
6	34.52	31.04
7	34.14	33.08
8	34.68	34.16
9	35.66	34.72
10	36.77	35.89
11	36.95	37.22
12	37.36	38.75
13	38.50	39.95
14	38.65	40.73
15	39.12	40.68
16	39.00	41.85
17	39.20	42.74
18	39.85	43.55
19	40.16	43.91
20	40.30	44.29



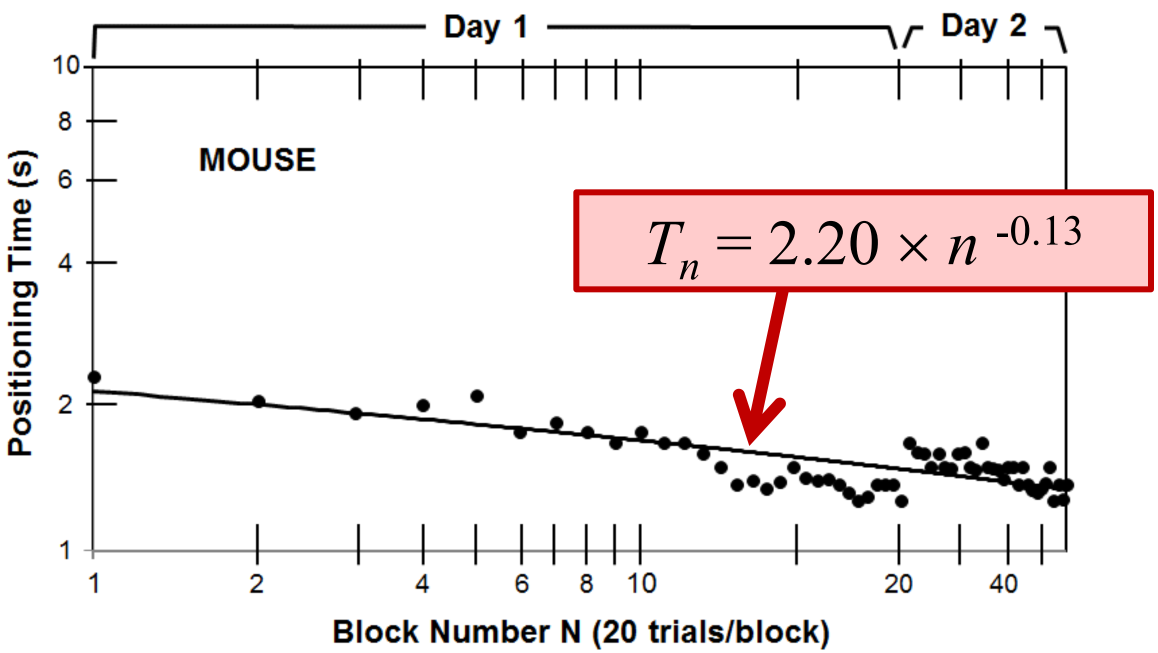
Power law of learning?
(next slide)

Power Law of Learning



Log-log Model

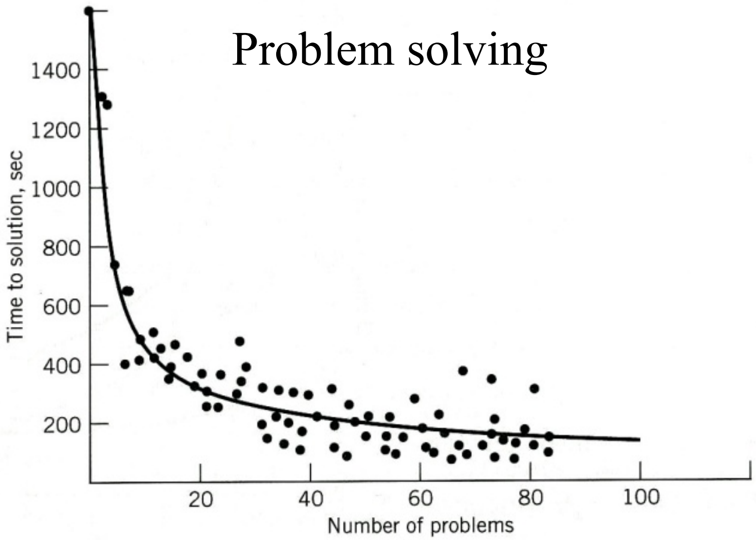
- If the x and y data are transformed to log scales, the relationship is linear
- Example:[1]



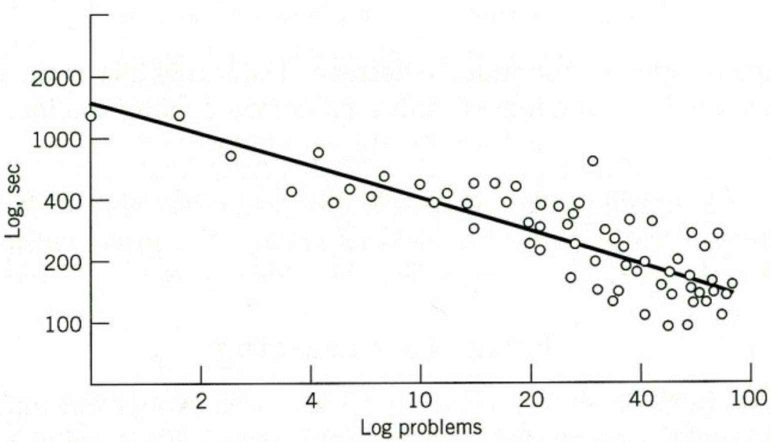
[1] Card, S. K., English, W. K., & Burr, B. J. (1978). Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics, 21*, 601-613.

Learning
("Learning and memory" Anderson)

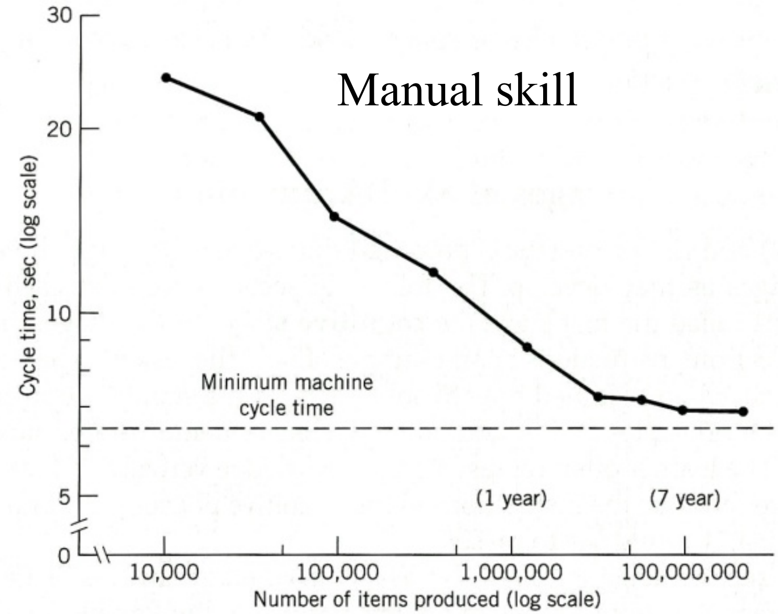
- Power law of learning

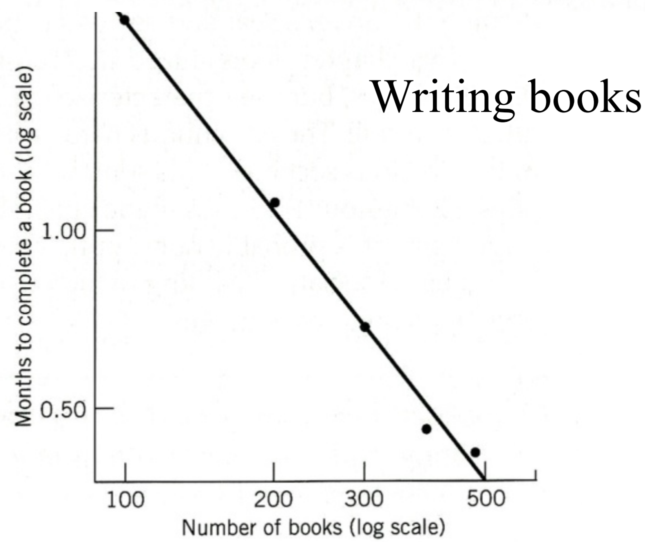


(a)



(b)

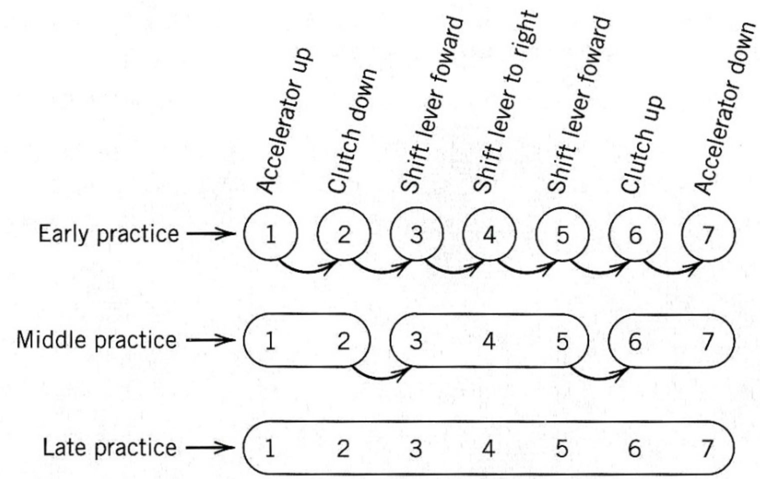




Stages of skill acquisition
("Learning and memory" Anderson)

Example: Using a manual transmission

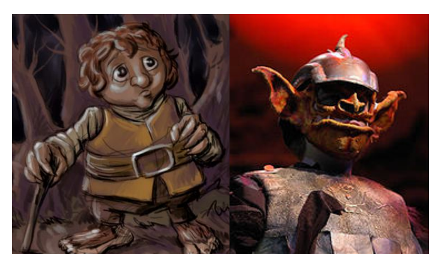
- Cognitive
  - Verbal representation of knowledge
  - Instructions or Examples
  - Learn through problem-solving
- Associative
  - Proceduralization
    - *From rehearsal to recognition*
- Autonomous
  - More and more automated
  - Faster and faster
  - No cognitive involvement
    - *become difficult to verbally describe the skill (or what to do)*
  - **The importance of motor program**

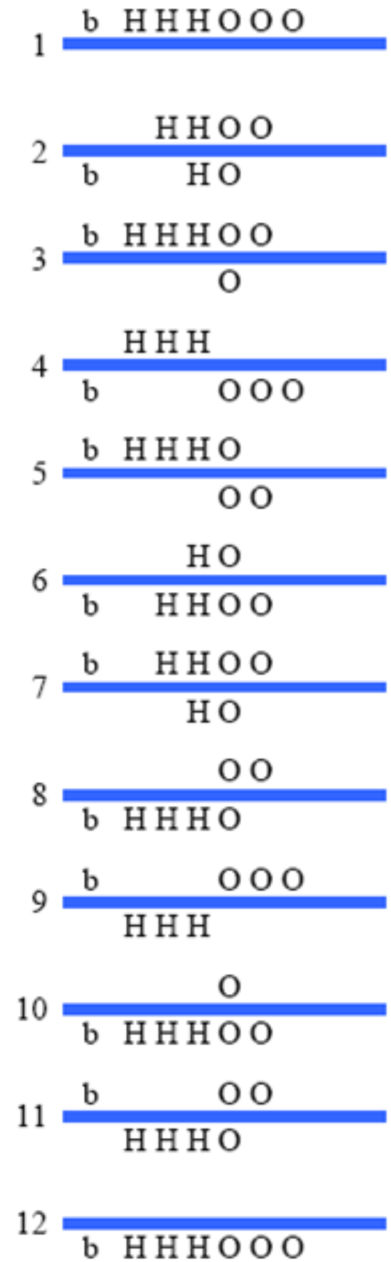


Problem solving

- Hobbits and Orcs crossing a river
  - three hobbits and three orcs arrive at a riverbank
  - they all wish to cross onto the other side
  - A boat can only hold two creatures at one time
  - Whenever orcs outnumber hobbits on any side of the river, the orcs will kill the hobbits

- Problem solving framework
  - States
  - Goals
  - Operators
    - *Mechanisms to select operators*
      - Difference reduction
      - Sub-goaling
  - Search



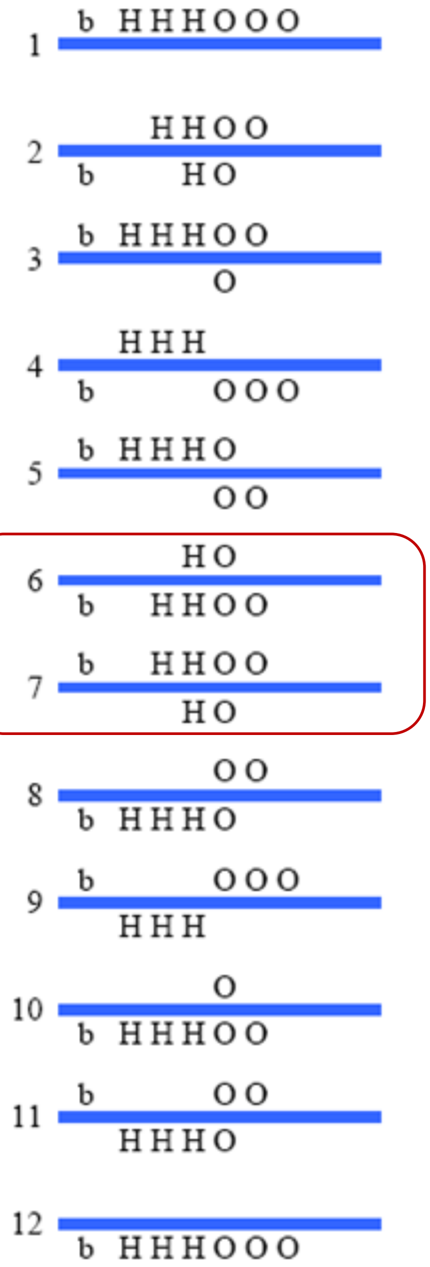


Difference Reduction

- What is the difficult step in the Hobbit-Orcs problem?
  - This transition goes against the gain of difference reduction, but it's critical.

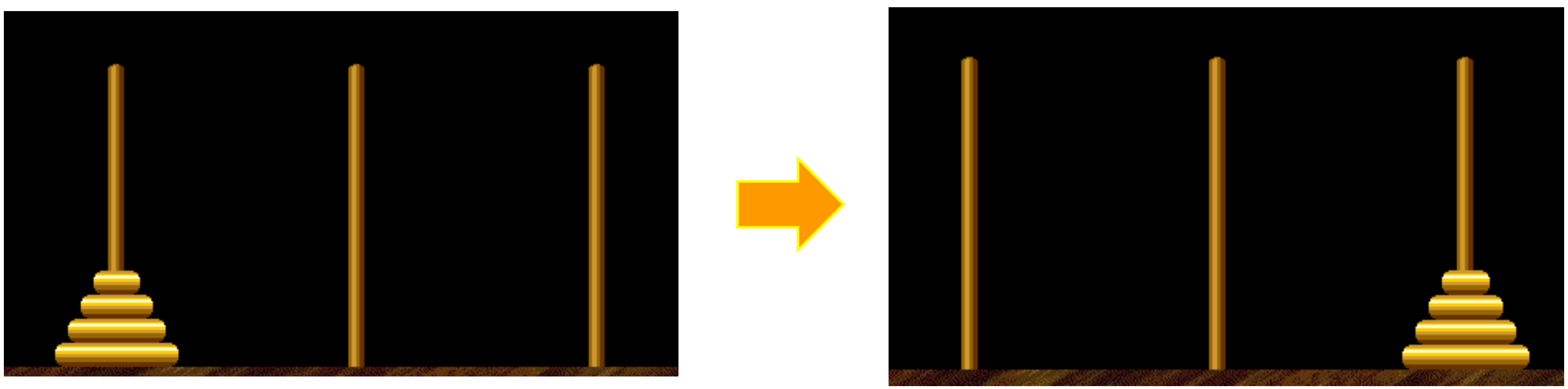
- "Greedy" algorithm
- Optimize simple metric

  => could get stuck in local minima



Operator Subgoaling

- <= Working backward from goals
- Requires memory for state
- Like AI-style “planning”



Production Rules

- Condition-action pairs represent procedural knowledge
- Can be applied with less memory

- Can recognize directly what to do without having to think through all the possibilities.

  (i.e., linear search replaced with immediate recognition)

- How does this apply to counting?
- How does this relate to long-term memory?

Experts

- Practice
- Knowledge
- Rules

=> Automatized Skills

- Doesn’t engage cognitive system
  - No working memory load
  - Not interruptible
  - Even perception goes away as system goes open-loop