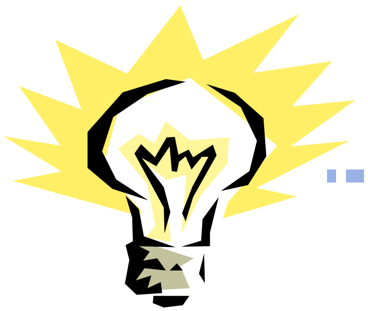


Models of Users: Motor, Cognitive, Description

Predicting thoughts and actions

Agenda

- Cognitive models
- Physical models

User Modeling

- Build a model of how a user works, then predict how she will interact with the interface.
- Goals (Salvendy, 1997):
  1. Predict performance of design alternatives
  2. Evaluate suitability of designs to support and enhance human abilities and limitations
  3. Generate design guidelines that enhance performance and overcome human limitations

Note: Not even a mockup is required

Differing Approaches

- Human as Information Processing machine
  - “Procedural” models
  - Many subfamilies and related models
- Human as a biomechanical machine
- Human as a social actor
  - Situated action
  - Activity theory
  - Distributed cognition

Some Cognitive Models

1. Model Human Processor
2. GOMS
3. Production Systems
4. Grammars

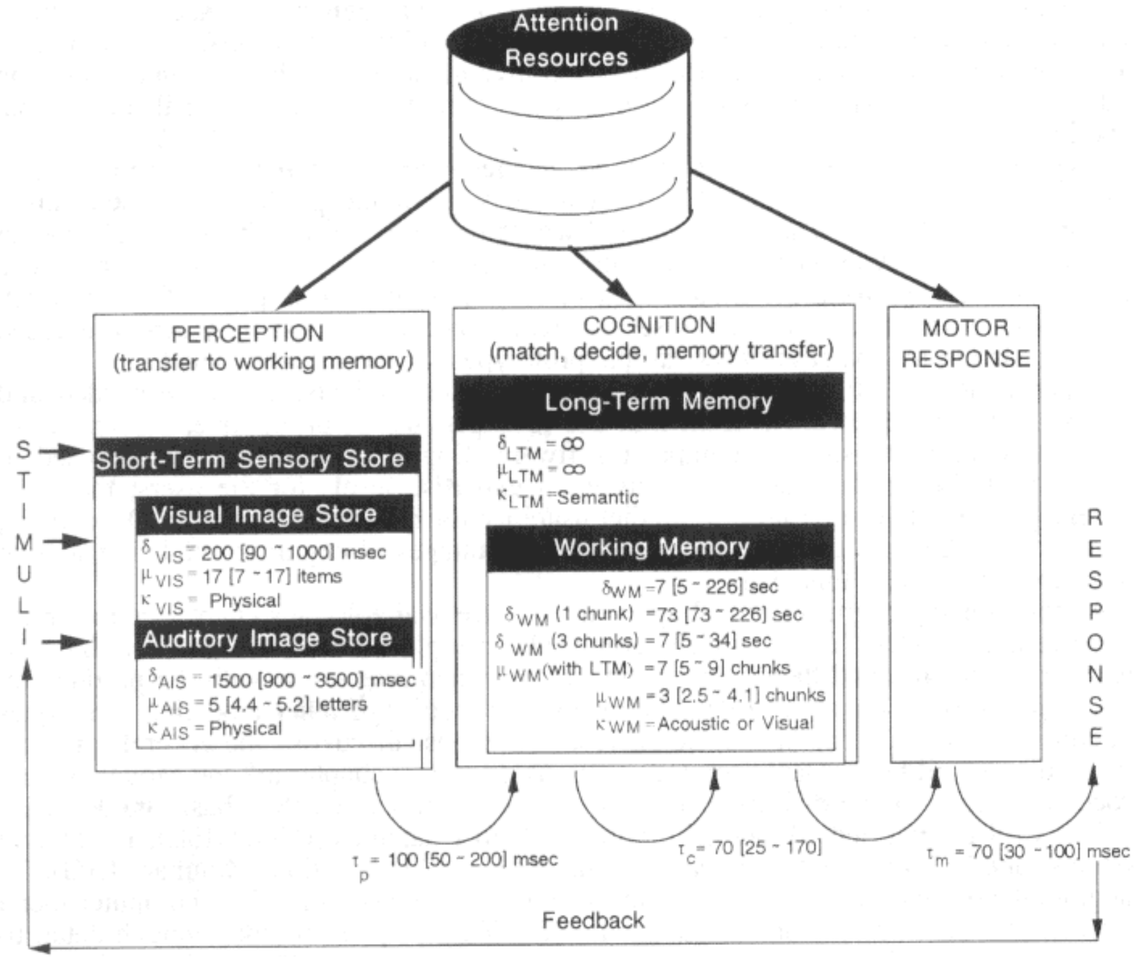
1. Model Human Processor

- Card, Moran, & Newell (1983, 1986)
- "Procedural" models:
  - People learn to use products by generating rules for their use and "running" their mental model while interacting with system
- Components
  - Set of memories and processors
  - Set of "principles of operation"
  - Discrete, sequential model
  - Each stage has timing characteristics (add the stage times to get overall performance times)

MHP: Three Systems in Model

- Perceptual, cognitive, motor systems
  - Timing parameters for each stage/system
  - Cycle times (τ):
    - $\tau_p \approx 100$ ms (“middle man” values)
    - $\tau_c \approx 70$ ms
    - $\tau_m \approx 70$ ms
  - Perception & Cognition have memories
  - Memory parameters
    - Code, decay time, capacity

MHP: Model and Parameters

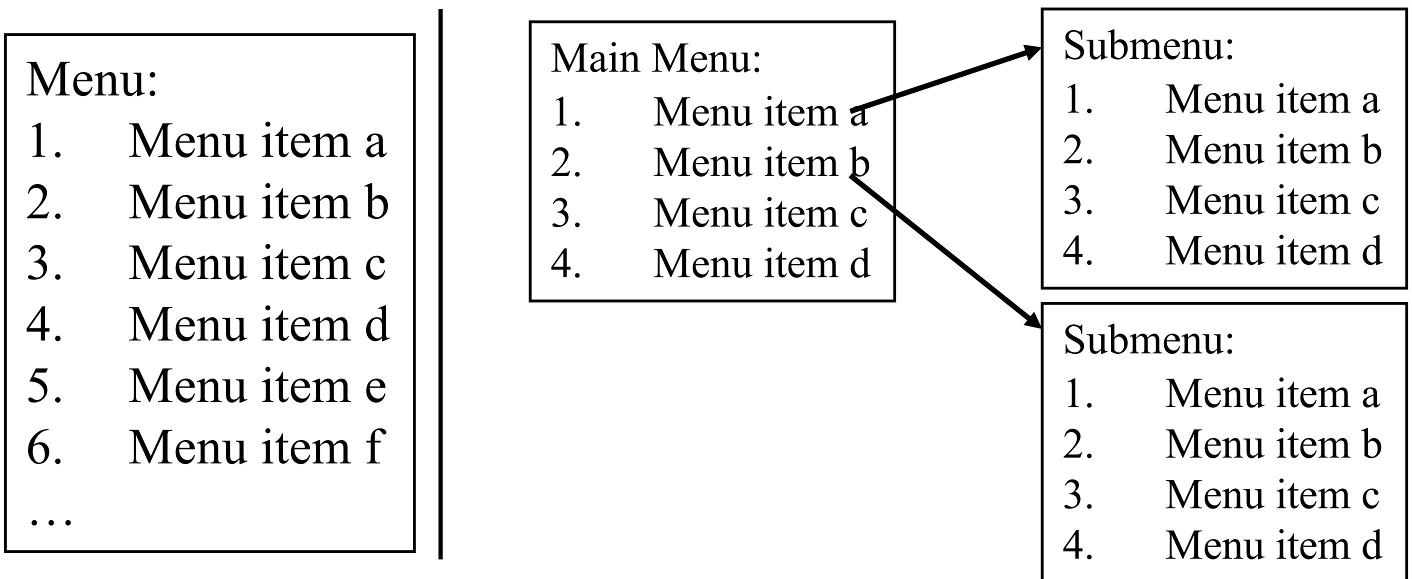


MHP: Principles of Operation

- Set up rules for how the components respond.
- Can be based on experimental findings about humans.
  - Recognize-act cycle, variable perceptual processor rate, encoding specificity, discrimination, variable cognitive processor rate, Fitts' law, Power law of practice, uncertainty, rationality, problem space
  - Note: *caveat emptor*

Applying the MHP

- Example: Designing menu displays
  - 16 menu items in total
  - Breadth (1x16) vs. Depth (4x4) ?



MHP: Calculations

Breadth (1x16):

τ_p perceive item, transfer to WM

τ_c retrieve meaning of item, transfer to WM

τ_c Match code from displayed to needed item

τ_c Decide on match

τ_m Execute eye mvmt to (a) menu item number (go to step 6) or (b) to next item (go to step 1)

τ_p Perceive menu item number, transfer to WM

τ_c Decide on key

τ_m Execute key response

Time = $[((16+1)/2)\ (\tau_p + 3\tau_c + \tau_m)] + \tau_p+\tau_c+\tau_m$

Time = 3470 msec

Serial terminating search over 16 items

Depth (4x4):

Same as for breadth, but with 4 choices, and done up to four times (twice, on average):

Time = $2 \times [((4+1)/2)\ (\tau_p + 3\tau_c + \tau_m)] + \tau_p+\tau_c+\tau_m$

Time = 2380 msec

Therefore, in this case, 4x4 menu is predicted to be faster than 1x16.

Related Modeling Techniques

- Many techniques fall within this "human as info processor" model
- Common thread - hierarchical decomposition
  - Divide behaviors into smaller chunks
  - Questions:
    - What is unit chunk?
    - When to start/stop?

2. GOMS

- **G**oals, **O**perators, **M**ethods, **S**election Rules
  - Card, Moran, & Newell (1983)
- Assumptions
  - Interacting with system is problem solving
  - Decompose into subproblems
  - Determine goals to “attack” problem
  - Know sequence of operations used to achieve the goals
  - Timing values for each operation

GOMS: Example

- Menu structure (breadth vs. depth, again)
- Breadth (1x16):

Goal: perform command sequence

    Goal: perform unit task of the command

Loops

        Goal: determine which unit task to do

            Operator: Look at screen, determine next command

        Goal: Execute unit task

            Select: Which method to enter number of command

            e.g. IF item # between 1 & 9 THEN use 1-KEY METHOD

            Operator: Use 1-Key Method

            Operator: Verify Entry… etc.

Result: Average Number of Steps = 33

GOMS: Example, cont'd

- Depth (4x4):
- Similar steps, in slightly different order and looping conditions
  - Result: Average Number of Steps = 24

- Comparison: Depth is ~25% faster in this case
  - Card et al. did not specify step length (in time)
  - Assume 100msec/step, then depth is 0.9 sec faster
  - Similar to Model Human Processor results

GOMS: Limitations

- GOMS is not so well suited for:
  - Tasks where steps are not well understood
  - Inexperienced users
- Why?

GOMS: Application

- NYNEX telephone operation system
  - GOMS analysis used to determine critical path, time to complete typical task
  - Determined that new system would actually be slower
  - Abandoned, saving millions of dollars

GOMS: Variants

- Keystroke Level Model (KLM)
  - Analyze only observable behaviors
    - Keystrokes, mouse movements
  - Assume error-free performance
  - Operators:
    - K: keystroke, mouse button push
    - P: point with pointing device
    - D: move mouse to draw line
    - H: move hands to keyboard or mouse
    - M: mental preparation for an operation
    - R: system response time

Example of KLM

- Breadth menu (1x16)
  - M: Search 16 items
  - 1 or 2 K: Enter 1 or 2-digit number
  - K: Press return key

  Time=M + K(first digit) + 0.44K(second digit) + K(Enter)

  (Look up values, and when to apply "M" operator)

  Time=1.35 + 0.20 + 0.44(0.20) + 0.20 = 1.84 seconds

  Note: Many assumptions about typing proficiency, M's, etc.

  Also ignores most of the time spent determining which task to perform, and how to perform it.

- Depth menu (4x4)
  - M: Search 4 items
  - K: Enter 1-digit number (no M, since expert user)
  - K: Press return key

  Time=M + K(Digit) + K(Enter)

  Time=1.35 + 0.20 + 0.20 = 1.75 seconds

- Compare the various models in terms of times and predictions:
  - Vary in times, but not in performance predictions

3. Production Systems

- IF-THEN decision trees (Kieras & Polson, 1985)
  - e.g. Cognitive Complexity Theory
  - Uses goal decomposition from GOMS and provides more predictive power
  - Goal-like hierarchy expressed using if-then production rules
- Very long series of decisions
  - Note: In practice, very similar to NGOMSL
    - Bovair et al (1990) claim they are identical
  - NGOMSL model easier to develop
  - Production systems easier to program

4. Grammars

- To describe the interaction, a formalized set of productions rules (a language) can be assembled.
- "Grammar" defines what is a valid or correct sequence in the language.
- Reisner (1981) "Cognitive grammar" describes human-computer interaction in Backus-Naur Form (BNF) like linguistics
- Used to determine the consistency of a system design

Task Action Grammars (TAG)

- Payne & Green (1986, 1989)
- Concentrates on overall structure of language rather than separate rules
- Designed to predict relative complexity of designs
- Not for quantitative measures of performance or reaction times.
- Consistency & learnability determined by similarity of rules

Summary of Cognitive Models

1. Model Human Processor (MHP)
2. GOMS
   - Basic model
   - Keystroke-level models (KLM)
   - NGOMSL
3. Production systems
   - Cognitive Complexity Theory
4. Grammars

Modeling Problems

- Terminology - example
  - Experts prefer command language
  - Infrequent novices prefer menus
  - What's "frequent", "novice"?
- Dependent on "grain of analysis" used
  - Can break down getting a cup of coffee into 7, 20, or 50 tasks
  - That affects number of rules and their types
  - (Same issues as Task Analysis)

Modeling Problems (contd.)

- Does not involve user per se
  - Doesn't inform designer of what user wants
- Time-consuming and lengthy, (but…)
- One user, one computer issue
  (lack of social context)
  - i.e., non-situated
  - Can use Socially-Centered Design

Physical/Movement Models

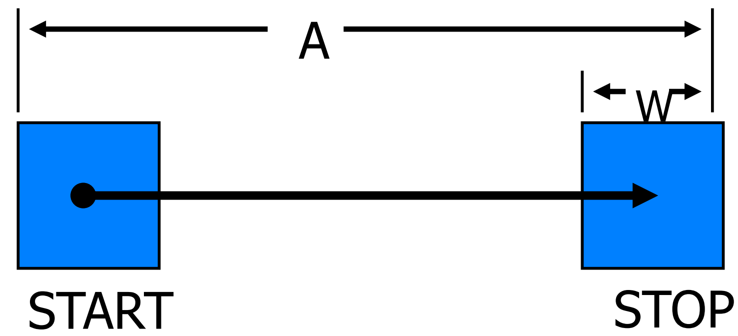
- Fitts' Law
- Simulations

Fitts' Law

- Fitts' Law
  - Models movement times for selection tasks
  - Paul Fitts: war-time human factors pioneer

- Basic idea: Movement time for a well-rehearsed selection task
  - Increases as the distance to the target increases
  - Decreases as the size of the target increases

- Move from START to STOP



Index of Difficulty:

$ID = \log_2 (2A/W)$ (in unitless bits)

distance

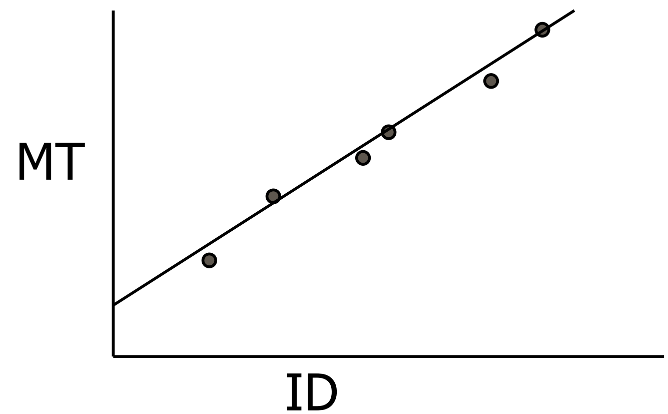
width of target

Movement Time

$MT = a + b*ID$

or

$MT = a + b \log_2 (A/W + 1)$



Different devices/sizes have different movement times--use this in the design

- What do you do in 2D?
- Where can this be applied in interface design?

Extending to 2D, 3D

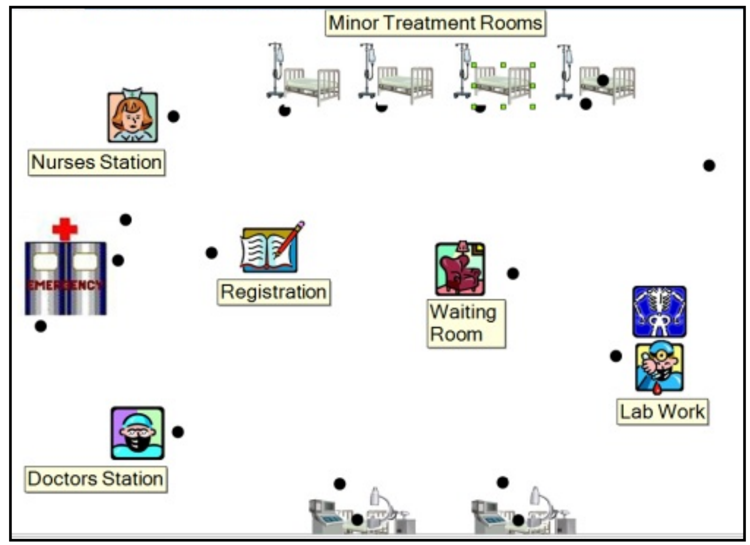
- What is W in 2D? In 3D?
- Larger movements?
  - Short, straight movements replaced by biomechanical arcs

Simulations

- Higher-level, view humans as components of a human-machine system
- E.g., MicroAnalysis and Design (maad.com)
  - Micro Saint - any type of models!
  - WinCrew - workload models
  - Supply Solver - supply chain

e.g., Micro Saint Sim Tools

http://www.microsaintsharp.com



(Social) Context

- Human information processor models all involve unaided individual
- In reality, people work with other people and other artifacts
- Other models of human cognition
  - Situation action
  - Activity theory
  - Distributed cognition

Situated Action

- Studies situated activity or practice
  - Activity grows out of the particulars of a situation
  - Improvisation is important

- Basic unit of analysis is "the **activity** of persons acting in a **setting**"

Example

- Need 3/4 of a cup of cottage cheese
  - Just has a 1-cup measuring cup available
- Person solves problem by
  - Measuring 1 cup
  - Pouring out into a circle
  - Divide into quadrants
  - Take away one quarter
- One time solution to one time problem

Situated Action Principles

- Structuring of an activity grows out of immediacy of the situation
- People engage in opportunistic, flexible ways to solve problems

- NOT Formulaic plans
- NOT Rational problem solving

Activity Theory

- Unit of analysis is an activity
- Components:

subject, object, actions, operations

- subject → Noun
- object → Held by subject, motivates activity "object of game"
- actions → Goal-directed processes "tasks"
- operations → How action is carried out

Activity Theory Principles

- Key idea: Notion of *mediation* by artifacts (objects)
- Our work is a computer-mediated activity
  - Starring role goes to activity
  - In "regular" HCI, stars are person and machine
- Context is not "out there". It is generated by people in activities

Distributed Cognition

- Unit of analysis is cognitive system composed of individuals and the artifacts they use
- Studies the coordination and cooperation between people and artifacts in a distributed process

Distributed Cog. Principles

- NOT Individual agents
- Distributed collection of interacting people and artifacts
- Functional system is what matters, not individual thoughts in people's heads

Simpler User Modeling

- How do attributes of users (in their context) influence the design of user interfaces?

- Are there some design guidelines that we can derive from different attributes?

User Profiles

- Attributes:
  - attitude, motivation, reading level, typing skill, education, system experience, task experience, computer literacy, frequency of use, training, color-blindness, handedness, gender,…

- Novice, intermediate, expert

Motivation

- <u>User</u>
  - Low motivation, discretionary use
  - Low motivation, mandatory
  - High motivation, due to fear
  - High motivation, due to interest
- <u>Design goal</u>
  - Ease of learning
  - Control, power
  - Ease of learning, robustness, control
  - Power, ease of use

Knowledge & Experience

- Experience
- task system
  - lowlow
  - high high
  - lowhigh
  - high low
- Design goals
  - Many syntactic and semantic prompts
  - Efficient commands, concise syntax
  - Semantic help facilities
  - Lots of syntactic prompting

Job & Task Implications

- Frequency of use
  - High - Ease of use
  - Low - Ease of learning & remembering
- Task implications
  - High - Ease of use
  - Low - Ease of learning
- System use
  - Mandatory - Ease of using
  - Discretionary - Ease of learning

Upcoming

- Evaluation