GOMS AND PREDICTIVE MODELS

Task Performance

Task performance is critical in
- airline and automobile displays
- emergency management systems
- process control systems
- customer service systems and more

Should measure task performance early to
- maximize performance of high-frequency tasks
- select among alternative designs
- cost justify replacement of an existing systems
Real-World Examples

• For every second saved in operator support, a company could save 3 million dollars per year
  – NYNEX estimate for its operator support, [Gray et al., GOMS Meets the Phone Company, Interact, 1990]

• Replace old workstations with new workstations
  – Promised to reduce operator support time by 2.5s
  – Weigh against investment of the new systems (about 1000 workstations at $5,000 each)

User Models

If we can build a model of how a user “works”, then we can predict how s/he will interact with the interface/system

Predictive model → predictive evaluation

• No mock-ups or prototypes!
  Kinda sort of…
Two Types of User Models

• Stimulus-Response
  – Practice law
  – Hick’s law
  – Fitt’s law

• Cognitive – human as interpreter/predictor – based on Model Human Processor cognitive model
  – Key-stroke Level Model
    • Low-level, simple
  – GOMS (and similar) Models
    • Higher-level (Goals, Operations, Methods, Selections)
**Power law of practice**

How much more efficient does someone get at doing a task with n trials?

\[ T_n = T_1 n^{-a} \]

- \( T \) on the nth trial is \( T \) on the first trial times \( n \) to the power \(-a\); \( a \) is between .2 and .6
- Skilled behavior - Stimulus-Response and routine
  * But NOT learning

- How can we use this law?

**Hick’s law**

How is decision time affected when choosing among \( n \) equally likely alternatives?

\[ T = I_c \log_2(n+1) \]

\((I_c \sim 150 \text{ msec})\)

- How can we use this law?
**Fitts’ Law**

How long does it take to move a pointer to a given target?

$$MT = k_1 + k_2 \times \log_2 \left( \frac{d}{w} + 1.0 \right)$$

- **Distance to move**
- **Width (tolerance) of target**

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

**Questions**

- What do you do in 2D?
  - Rectangles, or circles?
  - For rectangles $ID = \log_2(d/min(w, l) + 1)$

- How can we use this law?
Cognitive models

The “Model Human Processor”

A true classic - Card, Moran and Newell, *The Psychology of Human-Computer Interaction*

- Cognitive model based on results from experimental psychology and modeled on computer model
  - Fits much experimental data
  - But is a partial model

- Focus is on a single user interacting with some entity (computer, environment, tool)
  - Neglects effect of other people
GOMS

- **Goals**: what a user wants to accomplish
- **Operators**: mental or physical actions that change the state of the user or system
- **Methods**: groups of goals and operators
- **Selection rules**: determine which method to apply, if more than one available
How To Use GOMS

• Analyze hierarchical structure of a task
  – coarse analysis focuses more on the cognitive structure of a task
  – fine analysis focuses more on the structure imposed by the specific interface design

• Analyze alternative methods
• Assign operators to base level goals
• Assign times to operators
• Sum the operator times
• Write selection rules for the different methods

GOMS Example

Retrieve the article entitled “Why Goms?” written by Bonnie John in 1995 from the ACM Digital Library
Goal Structure

- Goal: Retrieve article from ACM DL
  - Goal: Go to ACM
    - Goal: Enter ACM URL
    - Goal: Submit URL
  - Goal: Go to DL
    - Goal: Locate DL link
    - Goal: Select the link
  - Goal: Select method
    - [Method: Search method
      - Goal: Search for article
        - Goal: Enter search parameters
        - Goal: Submit search
        - Goal: Identify article from results
      - Goal: Select the article]
    - [Method: Browse method - <take home exercise>]
  - Goal: Save article to disk
    - Goal: Initiate save action
    - Goal: Select location
    - Goal: save article to that location

GOMS: Pros and Cons

- Pros
  - predict human performance before committing to a specific design in code or running empirical studies
  - no special sills required
  - many studies have validated the model (it works)

- Cons
  - assumes error-free, skilled behavior
  - no formal recipe for how to perform decomposition
  - may require significant time investment
**Keystroke-Level Model (KSLM)**

- KSLM - developed by Card, Moran & Newell
- Skilled users performing routine tasks
- Assigns times to basic human operations - experimentally verified

**KSLM Model Includes**

- Keystroking \(T_K\)
- Mouse button press \(T_B\)
- Pointing (typically with mouse) \(T_P\)
- Movement between keyboard and mouse \(T_H\)
- Drawing straight line segments \(T_D\)
- “Mental preparation” \(T_M\)
- System response time \(T_R\)
KSLM

• Decompose task into sequence of operations
• Place M operators
  – Before selecting an option
  – In front of all K’s that are NOT part of argument strings (ie, not part of text or numbers)
• Example – Replace a word in a text document
  – Home on mouse H(mouse)
  – Point to word MP(word)
  – Select word BB(mouse button)
  – Home on keyboard H
  – Type new 5-letter word M5K

KSLM

• Now remove M’s according to the rules
  – Anticipated by prior operation
    • PMB -> PB
  – If string of MKs is a single cognitive unit (such as a command name), delete all but first
    • MKMKMK -> MKKK (same as M3K)
  – If K terminates a constant string, such as command-rtn, then delete M
    • M2K(1s)MK(rtn) -> M2K(1s)K(rtn)
KSLM

- Apply rules to example
  - Home on mouse \( H(\text{mouse}) \)
  - Point to word \( P(\text{word}) \)
  - Select word \( \times \text{MBB}(\text{mouse button}) \)
  - Home on keyboard \( H \)
  - Type new 5-letter word \( M5K \)

\[
T = 5T_K + 2T_B + T_P + 2T_H + T_M + T_R
\]

KSLM

- Plug in real numbers from experiments
  - \( K \): .08 sec for best typists, .28 average, 1.2 if unfamiliar with keyboard
  - \( B \): down or up - 0.1 secs; click - 0.2 secs
  - \( P \): 1.1 secs
  - \( H \): 0.4 secs
  - \( M \): 1.35 secs
  - \( R \): depends on system; often less than .05 secs

\[
T = 5T_K + 4T_B + T_P + 2T_H + T_M + T_R
\]

\[
= 5(.28)+4(.2)+1.1+2(.4)+1.35+.05 = 5.4 \text{ secs , 4.4}
\]
KSLM

- What is the optimal path?
  Contr-H, Steve, <tab>, Julie <enter>
  KK5KK5KK = 20(0.28) = 6.95, 2.95