

認知モデルに基づく ユーザビリティ評価（Ⅰ）

1. GOMSモデル：ルーチンタスクの実行時間の予測
Card, S. K., T. P. Moran, A. Newell. (1983).
The psychology of human-computer
interaction. Hillsdale, NJ, Lawrence Erlbaum
Associates. 5章
2. 認知ウォークスルー：初めて行うタスクの操作選択過程の予測

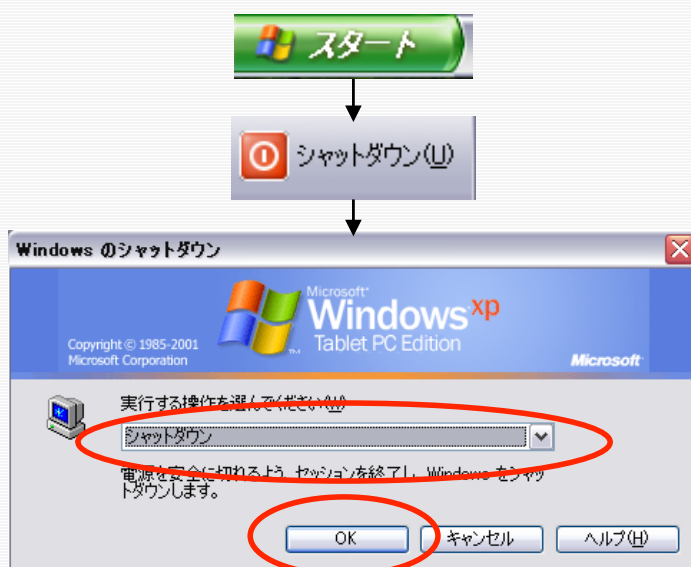
目的志向のルーチンタスクを対象とした認知行動のモデル化

目的志向のルーチンタスクの特徴

- 明確な目的のある行動
 - ▶ 携帯電話で日本語を入力する
 - ▶ エクセルのセルにデータを入力する
 - ▶ かばんから玄関の鍵を取り出し開ける
- 繰り返し実行される行動
 - ▶ 十分に練習が行われ、エキスパートの域に達している
 - ▶ 行動の仕方に再現性がある
 - － 行為の系列
 - － 行為の実行時間

プロダクションシステムによるモデル化

- 目的志向のルーチンタスクの実行過程を、プロダクションシステムでモデル化した場合
 - ▶ 明確なゴールが存在する
 - ▶ 同じ環境条件の場合に、同じプロダクションルールが発火する(成功経験による強化)
 - ▶ その結果、ゴール変換の過程が決定論的になる
- タスク遂行環境(システムデザイン)とゴール(タスク)が特定されたときに、ゴールを達成するために取られる行為の系列(発火するプロダクションルールの系列)は、特定のものになる
- システムデザイン+ゴール
 - 発火するプロダクション系列の予測が可能
 - タスク遂行時間の予測が可能



PCを終了する・・・ゴール

1. タスクバーのスタートをクリックする
2. シャットダウンをクリックする
3. シャットダウンが選択されているのを確認する
4. OKをクリックする

モデルヒューマンプロセッサの 動作原理との関係

プロダクションシステムは、これらの動作原理と整合するように行動をシミュレートする

□ 動作原理 8: 合理性原理

- ▶ タスクの構造、タスク実行者に対する情報入力、および知識の範囲と情報処理能力の限界が与えられたとき、その実行者は、合理的なアクションによってゴールを達成するように行動する
- ▶ 処理能力の限界においては、「限定合理性」と「満足化原理」が重要

□ 動作原理 9: 問題空間原理

- ▶ 人間が問題を解決する為に行う合理的なアクティビティは次の項目によって記述される
 1. 状態に関する知識
 2. 状態を更新するためのオペレータ
 3. オペレータを適用する際の制約条件
 4. 次に適用すべきオペレータを決定する為の制御知識

- 経験を積むことによりゴールを達成するためのオペレータ系列をその度に導き出す(問題空間原理により再計算する)ことは必要なくなり、経験知識からオペレータ系列を思い出して実行することができるようになる

- 合理的(時間最短、労力最小)であろうとするが故に、どのオペレータ系列を実行するかは、システムデザインによって概ね決定される

- システムデザイン+ゴール
→ 実行時間、オペレータ系列の予測が可能

GOMSモデルとは

- プロダクションシステムによるモデル化より次のことが導かれる
- ユーザがデバイスやシステムを利用して目的志向のルーチンのタスクを行う場面を対象としたとき、ユーザが持っている手続き的な知識は、
G:ゴール、
O:オペレータ、
M:メソッド、
S:選択規則 の4要素で表現される
→ GOMSモデル
- ユーザがルーチンのタスクを遂行するプロセスをGOMSで表現することにより、操作時間や学習時間が予測できる

ユーザビリティ評価を行う方法

- ユーザビリティ評価を行う方法には、ユーザテスト、ユーザビリティインスペクション法、認知工学モデルを利用した方法がある
- ユーザテストによる方法
 - ▶ 実際のユーザに開発中のシステムを使わせ、ユーザビリティにおける問題点を見つける
 - ▶ 問題のある箇所を改善する
 - ▶ 繰り返しテストする
 - ▶ 欠点は、時間がかかること。お金がかかる。テストユーザをリクルートするのが難しい場合がある
- 2つの解決策
 - ▶ ユーザビリティインスペクション法(チェックリストなど)
 - ▶ ユーザビリティ評価のための認知工学モデル
- 認知工学モデル(GOMSモデルなど)
 - ▶ 人間が、製品を利用する過程を認知モデルに基づいてシミュレートし、実行時間、操作学習時間、エラーを予測する

認知工学モデルを利用したユーザビリティ評価

- 認知工学モデルによるユーザビリティ評価の特徴
 - ▶ デザインが与えられたとき、ユーザがどのようにタスクを遂行できるかを予測する
 - ▶ 予測は実際のユーザによるテストの代用となる
 - デザインの変更の影響をすばやく評価できる
 - ▶ インタフェースデザインを分析し、成功要因を将来のデザインに利用できる
- 認知工学モデルを利用したインタフェースデザイン開発の流れ
 - ▶ タスク分析を行う
 - ▶ インタフェースデザインを開発する
 - ▶ 認知工学モデルを用いてユーザビリティ評価を行う
 - ユーザビリティの問題を探す
 - 問題がなくなるまでデザインを改善する
 - ▶ ユーザテストを行う
- 認知工学モデルを利用した製品評価
 - ▶ 製品開発時: 製品開発の初期の段階において人間の行動の定量的な予測を行うことにより、プロトタイプの実作を待たずに製品の評価を行える
 - ▶ 製品導入時: 仮に使用される場合を想定して、人間がそれを利用する過程のシミュレーションを行って旧製品との比較検討を行うことができる

GOMSモデルの基本要素

- 目的志向のルーチンタスクを遂行するのに必要な手続き的知識はGOMSで表現できる
- 1. G:ゴール(Goals)
- 2. O:オペレータ(Operators)
- 3. M:メソッド(Methods)
- 4. S:選択規則(Selection Rules)

G:ゴール = 「ユーザが達成したいこと」

□ ゴールの表現

- ▶ 階層的に表現される
 - ゴール、サブゴール、サブサブゴール、...
- ▶ 上位のゴールの達成のためには、全てのサブゴールの達成が必要

□ サブゴールへの分解は、分析の目的に応じて変わってくる

タスク:論文を共同で執筆する

- ・ ゴール:「論文を書く」
 - ・ サブゴール:最新のドラフトを共著者に送る
 - ・ サブゴール:参考文献をフォーマットする
 - ・ サブゴール:コメントに従って文書を修正する

執筆作業レベルの表現

- ・ サブゴール:文書を修正する
 - ・ サブサブゴール:テキストを移動する
 - ・ サブサブゴール:単語を削除する
 - ・ サブサブゴール:語を挿入する

文字列操作|コマンドレベルの表現

O:オペレータ＝ 「知覚行為、認知行為、運動行為」

- オペレータの実行によりユーザの心的状態や、外部環境の状態などに何らかの変化が生じる
- オペレータを特徴づけるパラメタ(実行時間など)は、ユーザやシステムが現在の状態に至った経緯(オペレータの履歴)の影響を受けない

サブサブゴール:「単語を削除する」を達成するためのオペレータ

- マウスを移動する
- マウスボタンをクリックする
- シフトキーを押しながらマウスボタンをクリックする
- 削除キーを押す

サブサブゴール「単語を削除する」は、これらのオペレータを順次実行して達成される

オペレータについて

- 複雑な行動は、要素的な行為からなっている
- 要素的な行為が作用(オペレート)して、外界の状態の物理的な変化やユーザの心的状態の変化を生じさせる
- 「要素的な行為」＝「オペレータ」
- 互いに独立なオペレータを定義できる
 - ▶ オペレータが単独で実行されたときに要する時間 = オペレータが系列の中で実行されたときに要する時間
- 全タスクを遂行する過程は要素オペレータに分解し、それらの系列として近似的に表現できる。また、要素オペレータの所要時間は、全タスクから切り離して測定できる

ヒューマンパフォーマンスⅢ, 25ページ(第5回)

オペレータの独立性を確認する 為の実験

□ オペレータ時間の計測

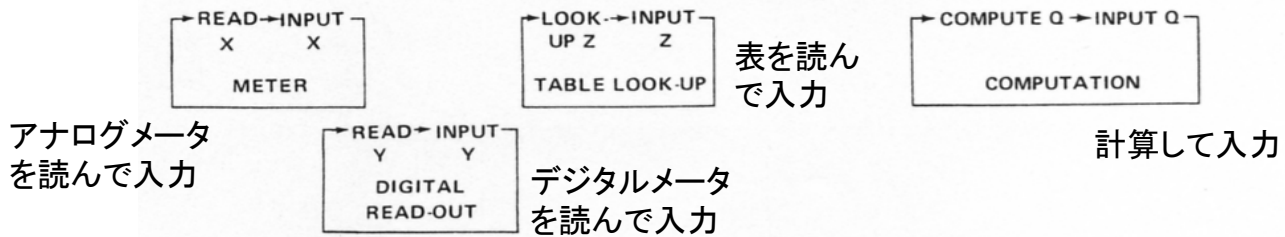
1. アナログメータから数値を読み取り、入力する
2. デジタルメータから数値を読み取り、入力する
3. 表から数値を読み取り、入力する
4. 入力した数値を使って計算して、結果を入力する

□ 複合タスクの遂行時間の計測

- ▶ ①アナログメータから数値を読み取り入力し、次に、②デジタルメータから数値を読み取り入力し、次に、③表から数値を読み取り入力し、最後に、④入力した数値を使って計算を行い、その結果を入力する

独立に実行した場合

(a) INDEPENDENT TASKS



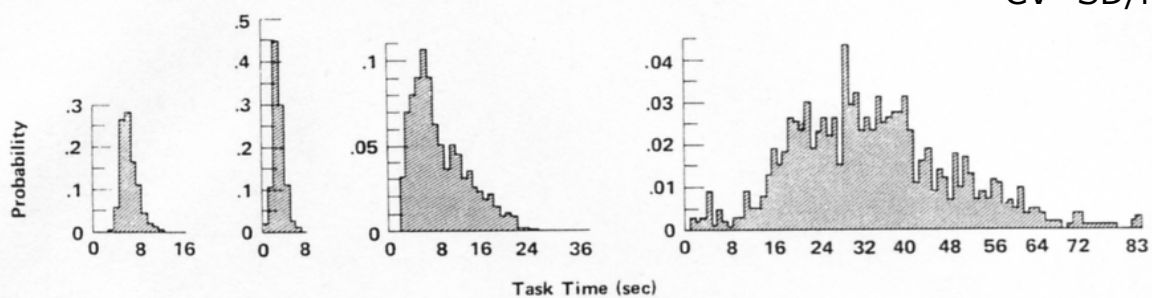
N = 986
Median = 6.04
Mean = 6.24
CV = .25

N = 987
Median = 3.36
Mean = 3.45
CV = .26

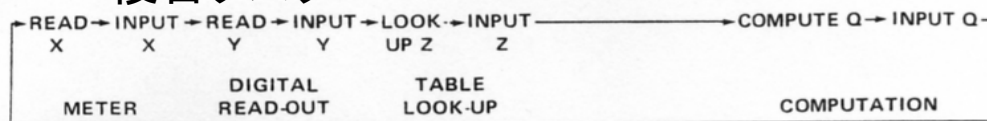
N = 984
Median = 7.99
Mean = 9.26
CV = .55

N = 989
Median = 32.83
Mean = 34.20
CV = .43

CV=SD/Mean



複合タスク (b) INTEGRATED TASK



N = 1226
Median = 6.31
Mean = 6.67
CV = .31

N = 1230
Median = 3.33
Mean = 3.47
CV = .30

N = 1225
Median = 8.36
Mean = 10.80;
CV = .65

N = 1236
Median = 30.04
Mean = 31.19
CV = .48

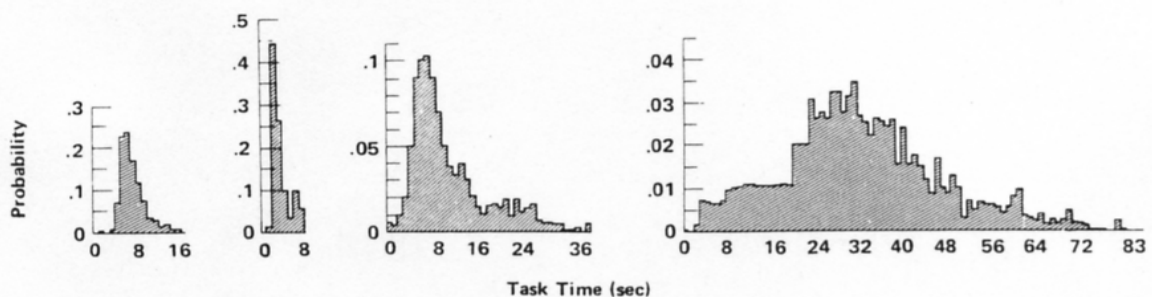


Figure 2.28. Time distributions for four operators (a) when measured in isolation and (b) when measured as part of an integrated task.

Five university students performed each of the following operators: READ-METER-AND-TYPE-INPUT, READ-DIGITAL-DISPLAY-AND-TYPE-INPUT, READ-X-Y-AND-LOOKUP-Z, READ-X-Y-Z-AND-COMPUTE-Q. They performed the operators both in isolation and as part of a larger integrated task. From Mills and Hatfield (1974, Figures 3 and 4).

ヒューマンパフォーマンスⅢ, 26ページ(第5回)

M:メソッド＝ 「サブゴールとオペレータの系列」

- メソッドは、サブゴールを達成するための十分に学習されたオペレータ系列
- メソッドは、自動的に実行され、実行時に、次にどのオペレータを実行するかを意識することはない

M1: マーク-削除メソッド

サブゴール「単語を削除する」を達成するためのメソッド

- ・ マウスを(単語の最初に)移動する
- ・ マウスボタンをクリックする
- ・ マウスを(単語の最後に)移動する
- ・ シフトキーを押しながらマウスボタンをクリックする
- ・ 削除キーを押す

S: 選択規則＝ 「メソッド選択のための知識」

- 選択規則は、個々のユーザに固有
- 判断基準となるパラメタの値はユーザ毎に異なっても構わない
- 選択規則は、個々のユーザの内部ではタスクの状況に拠らない一定基準

M1: マーク-削除メソッド

サブゴール「単語を削除する」を達成するためのメソッド

- ・ マウスを(単語の最初に)移動する
- ・ マウスボタンをクリックする
- ・ マウスを(単語の最後に)移動する
- ・ シフトキーを押しながらマウスボタンをクリックする
- ・ 削除キーを押す

M2: 削除キーメソッド

サブゴール「単語を削除する」を達成するための代替メソッド

- ・ マウスを(単語の最後に)移動する
- ・ マウスボタンをクリックする
- ・ 削除キーを(単語長分)押す

「単語を削除する」に関する選択規則

単語の長さが5文字以上ならばM1:マーク-削除メソッドを、それ以外の場合はM2:削除キーメソッドを用いる

ゴールとオペレータの関係

- ある行動をゴールとするかオペレータとするかは、分析者の選択する分析のレベルに依存して相対的に決められる

打鍵レベルのオペレータを設定している場合

ゴール:「論文を書くこと」

サブゴール:最新のドラフトを共著者に送る

サブゴール:参考論文をフォーマットする

サブゴール:文書を修正する

サブサブゴール:テキストを移動する(※)

サブサブゴール:単語を削除する(※)

メソッド: マーク-削除メソッド

オペレータ:マウスを(単語の最初に)
移動する

オペレータ:マウスボタンをクリックする

オペレータ:マウスを(単語の最後に)
移動する

オペレータ:シフトキーを押しながらマウ
スボタンをクリックする

オペレータ:削除キーを押す

サブサブゴール:語を挿入する(※)

ゴールとオペレータの関係(つづき)

文字列操作コマンドレベルの オペレータを設定している場合

ゴール:「論文を書くこと」

サブゴール: 最新のドラフトを共著者に送る

サブゴール: 参考論文をフォーマットする

サブゴール: 文書を修正する

メソッド: 単語削除挿入法

オペレータ: テキストを移動する(※)

オペレータ: 単語を削除する(※)

オペレータ: マウスを(単語の最初に)
移動する

オペレータ: マウスボタンをクリックする

オペレータ: マウスを(単語の最後に)
移動する

オペレータ: シフトキーを押しながらマウ
スボタンをクリックする

オペレータ: 削除キーを押す

オペレータ: 語を挿入する(※)

打鍵レベルでのサブサブゴールがオペレータとして設定されている

GOMSモデルで設定されるオペレータのレベル

□ オペレータの要件

1. オペレータの実行時間は履歴に依存しないこと
2. オペレータの実行時間の評価を既存のデータを用いて行うことができること。つまり、新たにオペレータ時間を決定する為に実験を行うことを必要としないこと

□ これらの制約条件により、GOMSモデルによる分析は、通常、コマンドレベル、あるいは打鍵レベルで行われている

GOMSモデルに用いられるオペレータの例(キーストロークレベル)

オペレータ	内容	時間(秒)
K	<p>キーまたはボタンを押す</p> <p>シフトキーを押すこと、コントロールキーを押すことは、ひとつのK操作とみなされる。また、マウスボタンを押すこともひとつのK操作とみなされる。時間はユーザのタイプ技能によって変化する。</p> <ul style="list-style-type: none"> ● 最も優れたタイピスト(135wpm) .08 ● 優れたタイピスト(90wpm) .12 ● 平均的なタイピスト(55wpm) .20 ● 平均的な非職業的タイピスト(40wpm) .28 ● ランダムに文字をタイプする .50 ● 複雑なコードをタイプする .75 ● キーボードに不慣れなタイピスト 1.20 	
P	<p>ディスプレイ上の目標をマウスでポイントする</p> <p>ポイントする時間は目標までの距離、目標の大きさに応じてFittsの法則に従って0.8から1.5秒の範囲で変化する。1.10秒はその平均値である。このオペレータには、ポイントの後に通常行われるボタンを押す操作(0.2秒)は含まれていない。マウスによるポインティング時間はジョイスティックなどの他のアナログポインティングデバイスに対してもよい評価を与える。</p>	1.10
H	キーボードや他のデバイス上の定位置に手を移動する	.40
M	心的準備	1.35
$R(t)$	<p>システム応答</p> <p>コマンド毎に異なった応答時間が必要になる。応答時間はユーザが待たされる場合にのみ考慮される。</p>	T

GOMSモデルの形式と利用法

□ 系列形式

- ▶ 観察された操作系列や、タスクを達成するのに必要とされる操作系列(キーストローク列、コマンド列)が与えられたときに、操作系列の分析を行い、タスク遂行時間を予測する

1. KLM: **K**eystroke **L**evel **M**odel
2. CPM-GOMS:
Critical-**P**ath-**M**ethod GOMS または
Cognition-**P**erception-**M**otor GOMS

□ プログラム形式

- ▶ 分析対象とするタスク(例えば、ワープロで実行できるタスクの集合)が与えられたとき、あらかじめ定義しておいたゴール、メソッド、選択規則、オペレータを用いて、シミュレーションを行う。オペレータ系列、実行時間を予測する

3. CMN-GOMS:
Card-Moran-Newell GOMS
4. NGOMSL: Natural GOMS Language

1. キーストロークレベルモデル

- 最も単純な系列形式のモデル
- オペレータとして、キーストロークレベルのものを設定
 - ▶ キー入力、キー押下、キー解放、マウスカーソル移動、マウスボタンクリック、手の移動(キーボード⇔マウス)、心的準備
 - ▶ 心的準備オペレータ(M): 種々のタスクを遂行する際のキー入力操作を大量に観察した結果、コマンド入力前、コマンド確定前など、特定の局面でポーズが入ることがわかった。心的準備オペレータはそれを表現するものとして導入された
- タスクを遂行するための操作系列が与えられたとき、オペレータ系列を明らかにし、個々のオペレータ実行時間を合計しタスク実行時間を予測する

M(心的準備オペレータ)挿入規則

メソッドのエンコーディングは、まず物理的な操作、システムの応答から始める。まず、規則0を使って M を仮に挿入する。そして、個々の M に対して規則1から規則4を適用し削除すべきかどうかを確かめる。

規則0 すべてのKの前にMを挿入する(ただし、引数である場合は除く)。すべてのコマンド選択のためのPの前にMを挿入する(ただし、引数選択は除く)。

規則1 Mに続くオペレータが、Mの直前のオペレータから完全に予期できるものである場合にはそのMを削除する(例えば、PMK→PK)。

規則2 MKの繰り返し系列が認知単位に属する(例えば、コマンド名)場合には、最初のMだけを残し、その他のMは削除する。

規則3 あるKが冗長なターミネーター(例えば、コマンド引数に続くコマンドターミネーター)である場合は、その前のMを削除する。

規則4 あるKが一定の文字列(例えば、コマンド名)を終了させるものである場合は、その直前のMを削除する。しかし、Kが一定でない文字列(例えば、引数文字列)を終了させる場合はその直前のMを保持する。

KLMの例

分析対象のキーストローケ列

NAME<SP>NEWMAN<SP>55<RETURN><応答>

規則0を適用

_MN_MA_MM_ME_M<SP>NEWMAN_M<SP>55_M<RETURN>
<応答>

規則2を適用

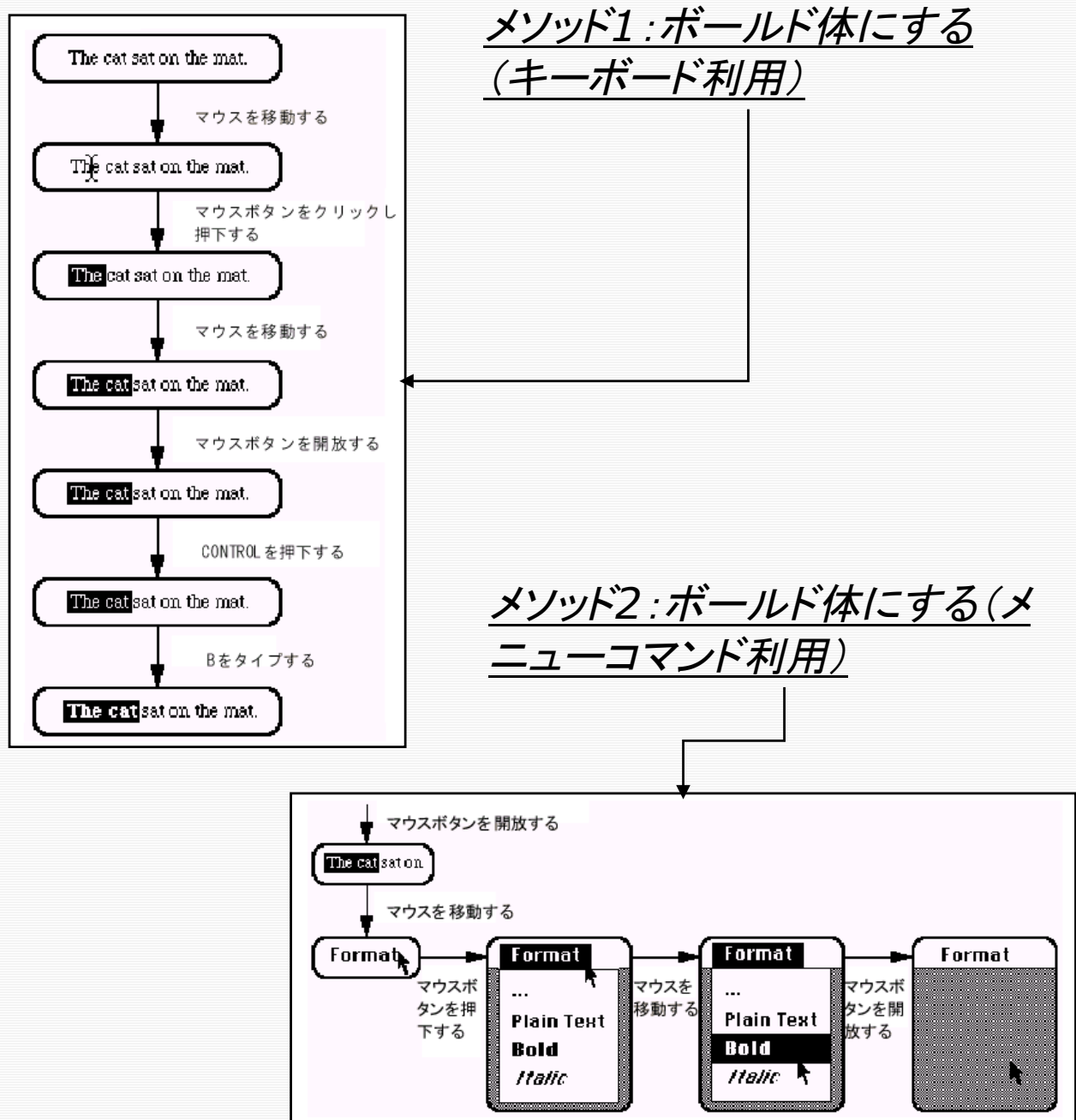
_MNAME_M<SP>NEWMAN_M<SP>55_M<RETURN>
<応答>

規則4を適用(コマンド後、削除;引数後、保持)

_MNAME<SP>NEWMAN_M<SP>55_M<RETURN><応答>

$$\begin{aligned} & K \times 15 + M \times 3 + \text{応答時間} \\ & = 0.5 \times 15 + \underline{1.35 \times 3} + 2.5 \\ & = 7.5 + \underline{4.05} + 2.5 = 14.05 \text{秒} \end{aligned}$$

KLMによる分析： フォントのスタイルの変更



KLMによる分析: フォントスタイルの変更

	オペレータ	時間	M除去規則	時間(合計)
<u>メソッド: 単語を選択する</u>				
1. マウスに手を伸ばす	H	0.40		
2. マウスポインタで 'The' の先頭を指す	P	1.10		
3. マウスボタンを押す	$\overline{\text{MK}}$	0.60	規則1	
4. 'cat' の末尾までマウスポインタを動かす	P	1.10		
5. マウスボタンを解放する	$\overline{\text{MK}}$	0.60	規則1	
単語選択時間				3.80 秒
<u>メソッド1: ボールド体にする(キーボード利用)</u>				
6. コントロールキーを押す	$\overline{\overline{\text{MK}}}$	0.60	規則2	
7. 文字キー 'b' を押して解放する	$\overline{\overline{\text{MK}}}$	0.60	規則2	
8. コントロールキーを解放する	$\overline{\overline{\text{MK}}}$	0.60	規則3	
メソッド1 合計時間 (3.80+1.80)				5.60 秒
<u>メソッド2: ボールド体にする(メニューコマンド利用)</u>				
6. Format メニューをマウスポインタで指す	$\underline{\text{P}}$	<u>1.10</u>		←
7. マウスボタンを押す	$\overline{\overline{\text{MK}}}$	0.60	規則1	
8. ボールド体をマウスポインタで指す	$\underline{\text{P}}$	<u>1.10</u>		←
9. マウスボタンを解放する	$\overline{\overline{\text{MK}}}$	0.60	規則1	
メソッド2 合計時間 (3.80+3.40)				7.20 秒

2. CPM-GOMSモデル

- KLMでは並列処理(左手、右手、先読み、など)をモデル化できない
- モデルヒューマンプロセッサにタスクを実行させる
 - ▶ 知覚、認知、運動プロセッサの並列実行
 - ▶ プロセッサ内では系列実行
 - ▶ プロセッサ間の協調
 - ▶ サイクル時間は50ミリ秒程度
- CPM-GOMS:
 - ▶ Cognitive-Perceptual-Motor: 認知・知覚・運動 レベルの解析
 - ▶ Critical-Path Method: 限界経路法

サブシステム	平均サイクル時間	範囲(ミリ秒)
知覚(視覚, 聴覚)	100	50-200
認知	70	25-170
運動	70	30-100

CPM-GOMSによるモデル化の方法

□ 手順

- ▶ 各プロセッサが系列的に実行する処理を記述
- ▶ プロセッサ間におけるデータの受け渡しを記述
- ▶ データが利用できるようになったら直ちに処理を始める (プロセッサはフルに働く) という条件で、全体のスケジュールを調整

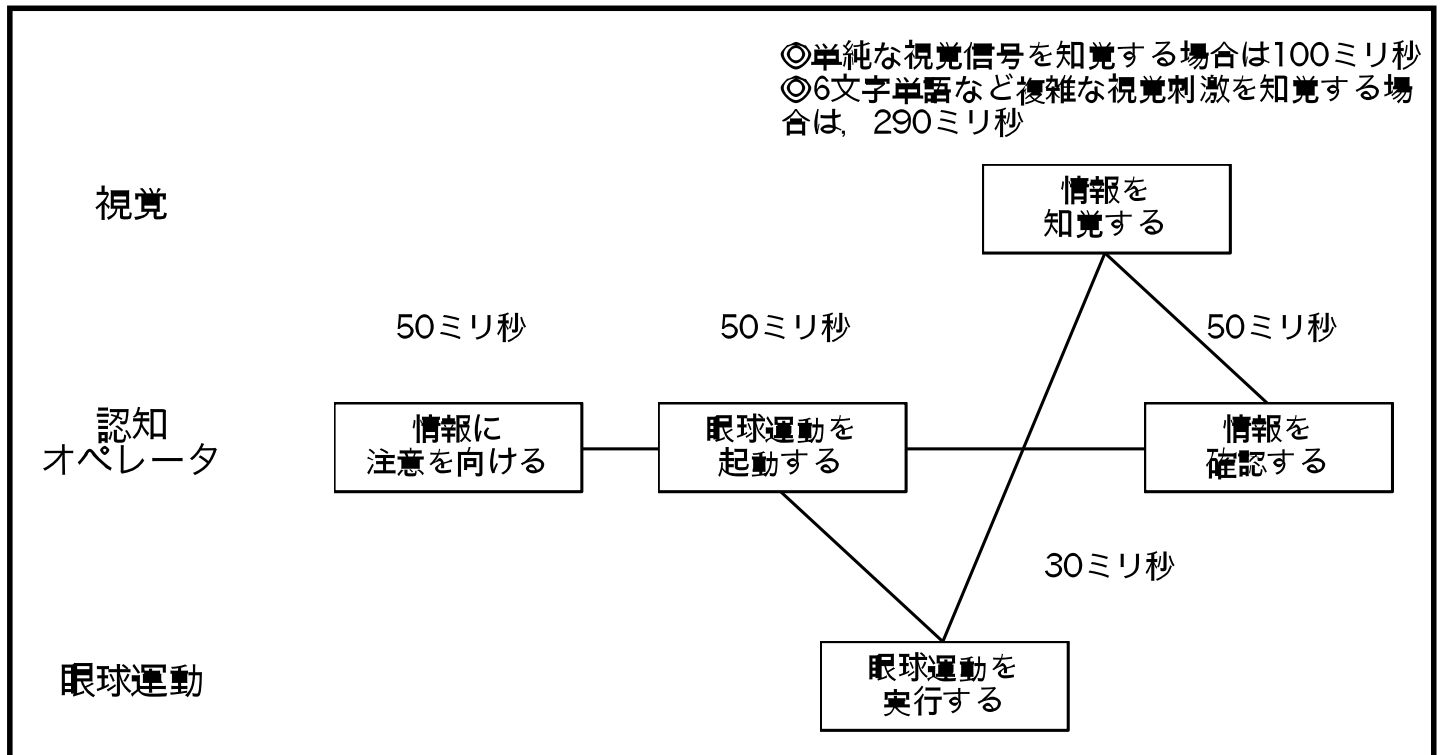
□ つまり、CPM-GOMSは、完全に最適化された専門家ユーザのルーチンのパフォーマンスのモデルとなっている

□ CPM-GOMSの応用事例:

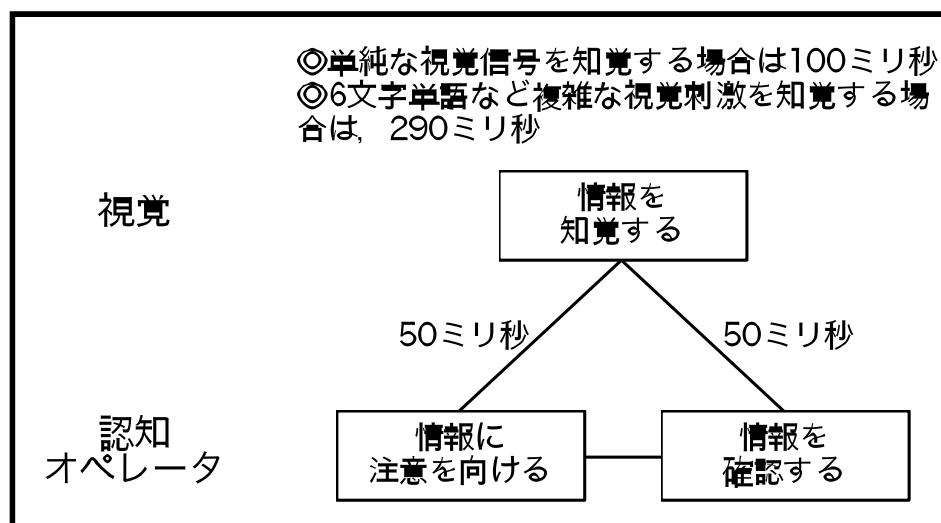
- ▶ 電話会社(NYNEX)の電話交換手ワークステーションのデザインの評価に利用され、新ワークステーションのデザインの不具合の発見に貢献した。それにより、電話会社は、無駄な投資せずに済んだ
- ▶ NYNEX Corporation (pronounced Nine-x) was a telephone company which served five New England states (Maine, Massachusetts, New Hampshire, Rhode Island and Vermont) as well as New York. Formed January 1, 1984 as a result of the Bell System Divestiture, NYNEX was a Regional Bell operating company made up of former AT&T subsidiaries New York Telephone Company and New England Telephone. Wikipedia より抜粋

CPM-GOMSモデルによる眼球運動のモデル化

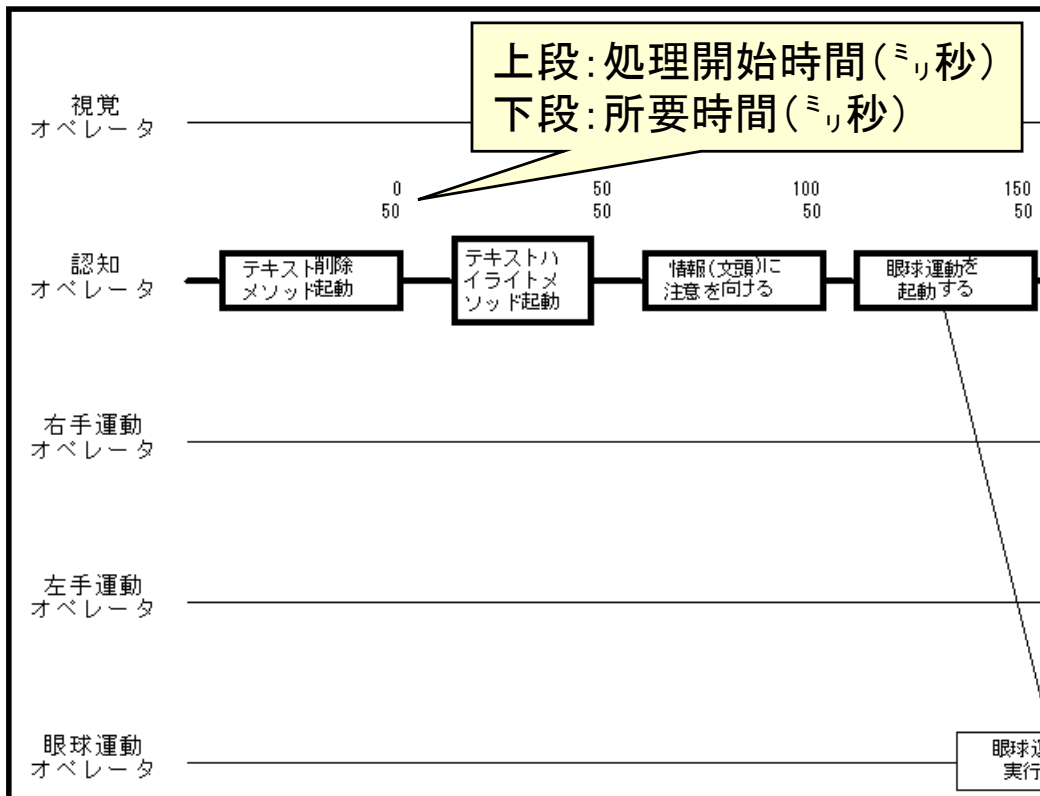
ゴール: 画面を読む(眼球運動を伴う場合)



ゴール: 画面を読む(眼球運動を伴わない場合)

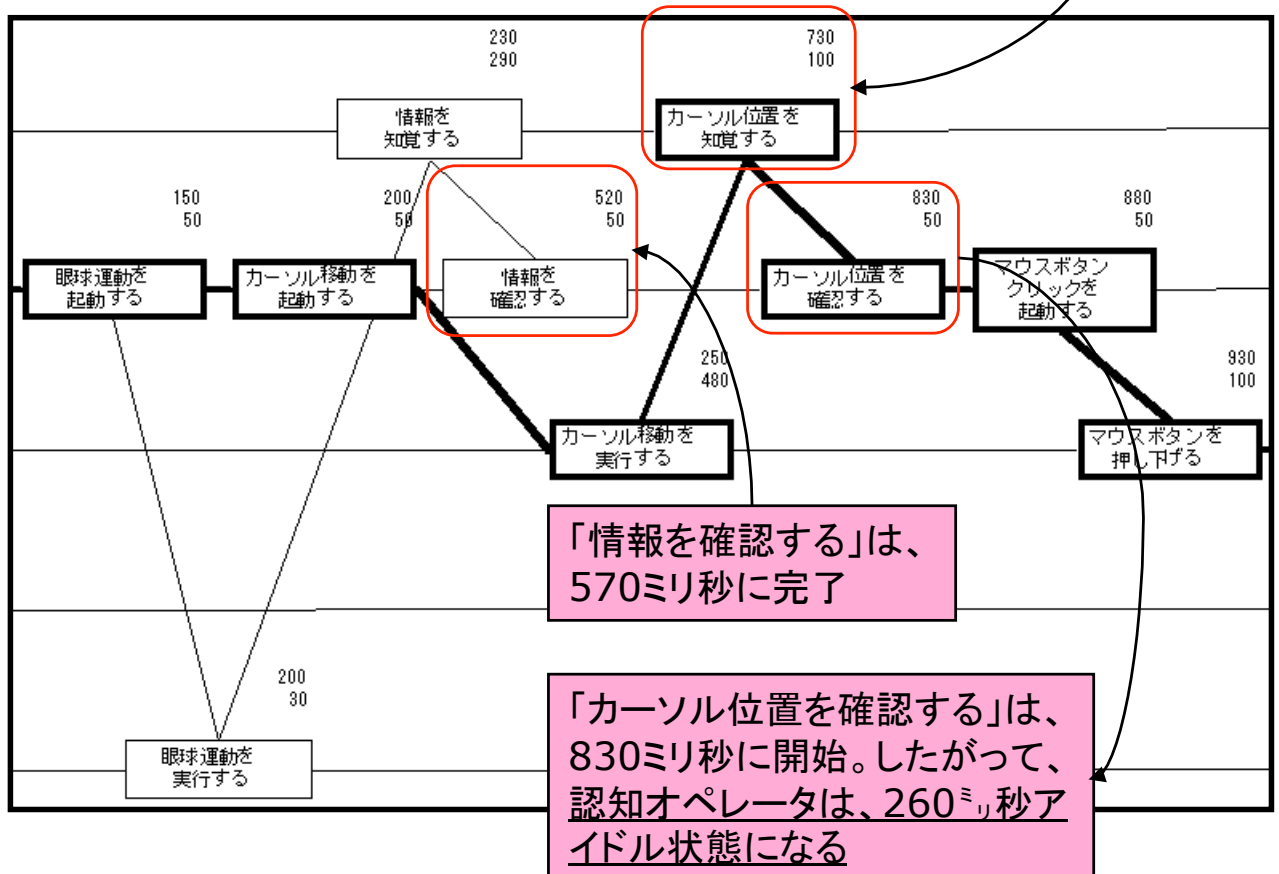


CPM-GOMS: マーク-削除メソッドの分析



分析の目的: 対象として
るインタフェースを利用して
タスクを遂行するときに、効
率の悪い部分を見つける

「カーソル位置を知覚する」は、830ミリ秒に完了



3. CMN-GOMSモデル

- Card, Moran, & Newell (1983) の第5章に示されている形式のGOMSモデル
- プログラム形式のGOMSモデル
 - ▶ GOMSモデルは、タスクの具体的な内容が与えられたときに、オペレータ系列に変換するプログラムを記述する
 - ゴールは完全な階層構造により表現される
 - 選択規則により、サブゴールの選択、メソッドの選択が制御される
 - メソッドは、オペレータ系列により、実行される
- GOMSモデルを作成すれば、具体的なタスクの状況を与えることにより、オペレータ系列を生成することができる。オペレータ時間を足し合わせて実行時間を予測することができる

GOMS:「文書編集」タスクの分析

ゴール：文書を編集する

- ゴール：単位タスクを編集する ・ ・ ・ 単位タスクがなくなるまで繰り返す
- ゴール：単位タスクを得る ・ ・ ・ タスクを思い出せない場合
- ・ ・ ・ ゴール：ページをめくる ・ ・ ・ 文書ページの最後だった場合
- ・ ・ ・ ゴール：文書から単位タスクを得る
- ゴール：単位タスクを実行する ・ ・ ・ 単位タスクが見つかった場合
- ゴール：テキスト変更
- ・ ・ ・ ・ [選択：
- ・ ・ ・ ・ ゴール：テキストを移動する
- ・ ・ ・ ・ ゴール：単語を削除する
- ・ ・ ・ ・ ゴール：語を挿入する]

→ ゴール：単語を削除する

- ・ [選択：
- ・ ゴール：マーク-削除メソッドを使う
- ・ ゴール：削除キーメソッドを使う]

マウスボタンクリック: 0.2
キー入力: 0.28

→ ゴール：マーク-削除メソッドを使う

- ・ カーソルを先頭に移動する (1.10)
- ・ マウスボタンをクリックする (0.20)
- ・ カーソルを末尾に移動する (1.10)
- ・ シフトキーを押しながらマウスボタンをクリックする (0.48)
- ・ ハイライトされていることを確認する (1.35)
- ・ 削除キーを押す (0.28)
- ・ 編集結果を確認する (1.35)

予測時間：

5.86秒

4. NGOMSLモデル

- Natural GOMS Language、エヌゴムズル: 構造化された自然言語によるGOMSモデル記述法
- プログラム形式のGOMSモデルであり、CMN-GOMSモデルの記述法を強化したもの
 - ▶ CMN-GOMSモデルでは、ゴール、メソッド、選択規則、オペレータの記述方法に何も制約がない
 - ▶ その結果、タスクを遂行するのに必要とされる知識の総体としてのGOMSモデルに、いくつものバリエーションができてしまう
 - ▶ そこで、プログラム記述上の制約として、オペレータの実行をプロダクションルールの発火に相当するものとしてとらえることにより、プログラムの各ステートメントに、認知プロセスとしての意味づけを行う
- 利用法:
 - ▶ オペレータ系列の予測、実行時間の予測、メソッド学習時間の予測

NGOMSL:「文書編集」タスクの分析

NGOMSLによる記述		外部オペレータ時間
ゴール[単語を削除する]のためのメソッド		
ステップ1.	ゴール[テキストをハイライトする]を達成する	
ステップ2.	削除キーを押す	0.20
ステップ3.	ゴールの達成により戻る	
ゴール[テキストをハイライトする]のためのメソッド		
ステップ1.	テキストの先頭の位置を決定する	1.20
ステップ2.	カーソルをテキストの先頭に移動する	1.10
ステップ3.	マウスボタンをクリックする	0.20
ステップ4.	テキストの末尾の位置を決定する(既知)	0.00
ステップ5.	カーソルをテキストの末尾に移動する	1.10
ステップ6.	シフトキーを押しながらマウスボタンをクリックする	0.48
ステップ7.	正しくテキストがハイライトされていることを確認する	1.20
ステップ8.	ゴールの達成により戻る	1.20
予測時間: $5.48 + 0.1 \times 13 = 6.78$ 秒		

各NGOMSLの記述の実行時間: 0.1秒

NGOMSLにおけるメソッド

- メソッドは、発火するプロダクションルールの系列を指定する
 - ▶ オペレータの実行＝プロダクションルールの発火
 - ▶ オペレータ系列の実行＝プロダクションルールの連鎖的発火
 - ▶ メソッドは、その系列を規定

- NGOMSLにおけるメソッドの学習
 - ▶ メソッドに含まれるプロダクションルールを学習することによってなされる
 - ▶ 一度学習されたルールは新しいメソッドの学習にトランスファーする
 - － 例えば、メソッド：CUTを学習すると、メソッド：CUT & PASTEにおけるCUTの部分は学習済みとなる

NGOMSL: 学習時間の予測

$$\begin{aligned} \text{学習時間} = & \\ & 17\text{秒} \times \text{NGOMSL記述数} \\ & + 6\text{秒} \times \text{LTMチャンク数} \end{aligned}$$

- 第1項: NGOMSLメソッドの学習時間
 - NGOMSLメソッドに含まれる記述数(ステップ数)に比例する
- 第2項: 長期記憶に記憶すべき宣言的知識(長期記憶チャンク)の学習時間
 - 例) コマンド“CUT”に対しては3個の長期記憶チャンクを記憶することが必要
 1. テキストをCUTするという概念(C1)
 2. “CUT”というコマンド(C2)
 3. C1とC2が結合しているという事実

GOMSモデルまとめ：系列形式のGOMSモデルの特徴

- タスクは分析者が定めたメソッドの系列によって実行される
- メソッドの表現
 - ▶ 一定のオペレータ系列による表現
 - ▶ 任意のタスクが実行されるのではなく、あらかじめ想定されたタスクが決められたオペレータ系列によって実行される
- 長所
 - ▶ オペレータ系列の把握が容易であること
 - ▶ 表現の簡便さ
 - 手続き的知識を明示的に定義する必要がない
- 短所
 - ▶ メソッドの階層構造を表現できない
 - ▶ オペレータ系列を枚挙するのは容易であるが、具体的なタスクの例の数が多くなるに従ってそれに要する時間が増大する

GOMSモデルまとめ: プログラム形式のGOMSモデルの特徴

- 想定し得る全てのタスクを実行させられる
- メソッドの表現
 - ▶ メソッドに任意のタスクパラメタを持たせることができる
 - タスクの状況に応じてパラメタを設定してタスクを遂行
例: マーク-削除メソッドのタスクパラメタ
 - ✓ 削除されるべき単語の開始位置と終了位置
 - ▶ メソッドの適用条件を条件付き分岐により明示
例: 「単語を削除する」に関する選択規則
 - ✓ 単語の長さが5文字以上ならば「マーク-削除メソッド」を、それ以外の場合は「削除キーメソッド」を用いる
- 長所
 - ▶ 分析者が全ての手続き的知識を把握できる
 - ▶ 多くのタスクの具体例を解析する必要がある場合には、生成的な特徴を利用して解析したいタスクを即座に生成させられる
- 短所
 - ▶ あるタスクを遂行するときに実行されるオペレータ系列を効率よく得ることができない
 - その唯一の方法は、モデルをタスクパラメタを用いて実際に実行してどのメソッドが使われるかを記録すること
 - ▶ 完全かつ正確なプログラム形式のGOMSモデルを定義するためには多くの時間が必要とされる

GOMSモデルの応用事例

文献	GOMS技法	タスク領域	予測内容
Card, Moran & Newell (1983).	KLM	テキストエディタ	実行時間
Lane, Napier, Batsell & Naman (1993).	KLM	スプレッドシート	実行時間
Haunold & Kuhn (1994).	KLM	地図のデジタル化	実行時間
Card, Moran & Newell (1983).	GOMS	テキストエディタ, オペレーティングシステム, VLSI CAD	オペレータ系列, 実行時間
Lerch, Mantel & Olson (1989).	GOMS	スプレッドシート	作業記憶エラー
Ekenon & Palmiter (1991).	GOMS	HyperCard™	学習時間
John, Vera & Newell (1994).	GOMS	ビデオゲーム	オペレータ系列
Kieras (1988).	NGOMSL	テキストエディタ	学習時間, 実行時間, 作業記憶エラー
Bovair, Kieras & Polson (1990).	NGOMSL	テキストエディタ	学習時間, 実行時間
Gong & Kieras (1994).	NGOMSL	人間工学CAD	オペレータ系列, 実行時間, 学習時間
Gray, John & Atwood (1993).	CPM-GOMS	電話交換手のワークステーション	実行時間
Cluah, John & Pane (1994).	CPM-GOMS	航空路スケジュール	実行時間

A Keystroke Level Analysis of a Graphics Application: Manual Map Digitizing

- Peter Haunold and Werner Kuhn, Technical University of Vienna, Department of Geoinformation, Austria, CHI'94, 1994.
- アナログ地図データのデジタル化(手動)
- 土地台帳データベース化事業(国土地理院)
- 地図シート数: 260,000枚
- ワークステーション数: 150台
- 最初の4年間の達成率: 43,000枚(17%)
- 遅い。お金がかかる。高信頼度の自動デジタル化システムがない。
- ⇒ デジタル化作業に必要とされるユーザインタフェースの特性の調査
- メソッドとタスクの記述。実行時間の計測。操作時間の予測、実測値との比較。タスク最適化(ズーム、パン)。

デジタル地図作成タスク

- アナログ土地台帳のスキャン ⇒ スクリーン上でデジタル化
- 入力デバイス: パッド上を移動する16個のボタンのあるカーソル
- 総タスク数: 80
 - ▶ 区画境界・家屋・区画識別子・記号のデジタル化
 - ▶ ビューの移動・変更(ズーム、パン)
 - ▶ 複写
 - ▶ 修正
- KLMでモデル化できるタスク: 38
- ユニットタスクへの分解、オペレータ
 - ▶ H: キーボードとカーソル間の手の移動
 - ▶ K: キーストローク
 - ▶ P: ポインティング
 - ▶ M: 心的準備
 - ▶ R: システム応答

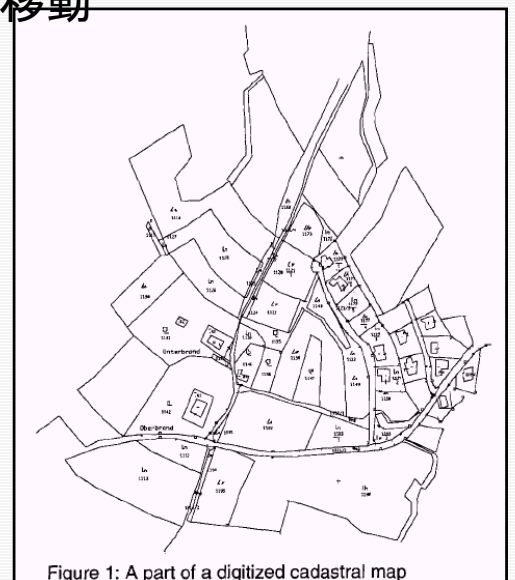


Figure 1: A part of a digitized cadastral map

例) 連続境界線のデジタル化

- P 頂点をポイントする
- K[0] 頂点の入力

$K \longrightarrow P \longrightarrow K \rightarrow P \rightarrow K$

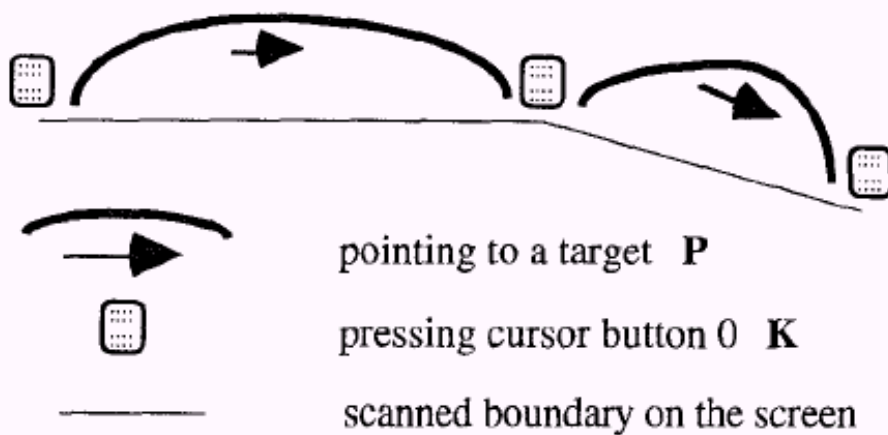


Figure 2: Continuous boundary digitizing

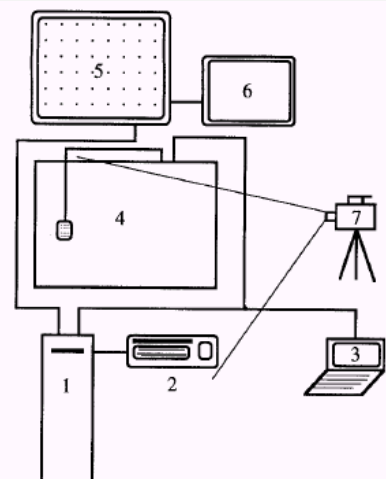
実行時間の計測

- 計測内容
- タスク数: 38
- 場所: 国土地理院の作業現場
- 精度: 100msec

- 作業環境
- プログラム: AUTOCAD
- タブレット: 16ボタン付カーソル
- モニター: 21インチカラーモニター

- 計測方法
- 計測用のプログラムを開発
 - ▶ タブレットとAUTOCADの間に挿入
 - ▶ カーソル、ボタン
- ビデオ録画
 - ▶ キーストローク数、ホーム移動

- 計測結果
- 被験者数: 7名
- 時間: 2～4時間
- アクション数: 25,000
- タスク数: 10,447

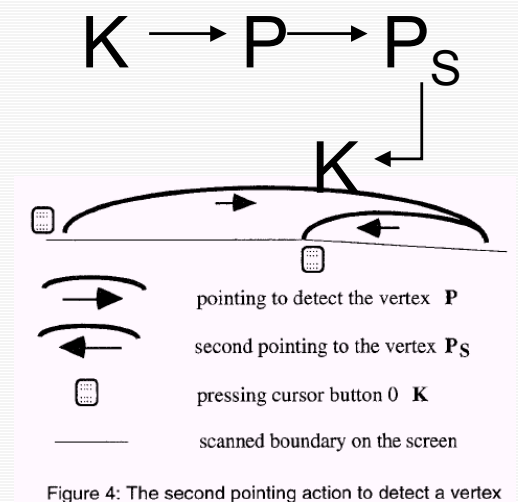


1...Digitizing PC (386er 33 MHz)
2...Keyboard
3...Observation PC (386er 25 MHz)
4...Digitizing Pad
5...Digitizing Monitor
6...Controlling Monitor
7...Videocamera

Figure 3: The Hardware Configuration of the Experiment

オペレータ時間の決定

- K_C : カーソルのボタンを押す
- P_S : 目標をいったん超え、戻りながらポイント
スナップ機能を利用してポイント
- R_1 : コマンドメニューから機能選択
ズーム／パンウィンドウ
- R_2 : 多角形の形成



Operator	# observed	Result [sec]	σ [sec]	Card et al. [sec]
K	400	0.25-0.50	0.08	0.08-1.20
K_C	1948	0.28-0.47	0.10	--
M	--	--	--	1.35
P	50	1.16	0.26	1.10
P_S	100	0.85	0.19	--
H	50	0.38	0.16	0.40
R_1	120	0.25	0.06	--
R_2	60	0.85	0.18	--

Table 1: Comparison between map digitizing and existing operators

アクション系列の分析を行って ユニットタスクを抽出した結果

Task	Function	Operator sequence	Number	T_m [sec]	T_n [sec]	D[%]
T-1	Start boundary digitizing with command selection	M,P,K,M,P,K	10	5.26 ± 0.36	5.70	-8
T-2	Start boundary digitizing using a snap function	2K,P _S ,K	164	2.28 ± 0.32	2.05	11
T-3'	Start boundary digitizing using a snap function	M,2K,P _S ,K	181	3.08 ± 0.58	3.40	-9
T-4	Continuous boundary digitizing	P,K	1621	1.57 ± 0.34	1.50	5
T-5'	Continuous boundary digitizing with path decision	M,P,K	103	3.11 ± 0.44	2.85	9
T-6	Continuous boundary digitizing using a second pointing action	P,P _S ,K	185	2.35 ± 0.25	2.35	0
T-7'	Continuous boundary digitizing using a second pointing action	P,M,P _S ,K	147	3.65 ± 0.61	3.70	-1
T-8*	Continuous boundary digitizing with reorientation	M,P,K	463	3.03 ± 0.54	2.85	6
T-9*	Continuous boundary digitizing using a second pointing action	M,P,M,P _S ,K	163	5.06 ± 0.61	5.05	0
T-10	Continuous boundary digitizing and defining a vertex as control point	P,2K	676	1.97 ± 0.40	1.90	4
T-11*	Continuous boundary digitizing and defining a vertex as control point	M,P,2K	128	3.22 ± 0.44	3.25	-1
T-12	End boundary digitizing	P,3K	147	2.41 ± 0.48	2.30	5
T-13	End boundary digitizing using a snap function	K,P _S ,3K	252	2.43 ± 0.35	2.45	-1
T-14'	End boundary digitizing using a snap function	K,P _S ,M,3K	105	3.65 ± 0.44	3.80	-4
T-15	Continuous house digitizing	P,K	292	1.55 ± 0.32	1.50	3
T-16	Continuous house digitizing using a snap function	K,P _S ,K	105	1.69 ± 0.35	1.65	2
T-17	End house digitizing and close polygon	4K,R ₂	16	2.56 ± 0.30	2.45	4
T-18	End house digitizing using a snap function and close polygon	K,P _S ,4K,R ₂	26	3.39 ± 0.33	3.70	-8
T-19	Continuous line digitizing for land use	P _S ,K	180	1.18 ± 0.32	1.25	-6
T-20	Start symbol digitizing with command selection	M,P,K,M,P _S ,K	94	5.80 ± 1.03	5.45	6
T-21	Continuous symbol digitizing	P _S ,2K	245	1.68 ± 0.36	1.65	2
T-22	Continuous symbol digitizing	P _S ,K	80	1.33 ± 0.24	1.25	6
T-23	Zoom/Pan task	K,R ₁ ,P,K,R ₁	712	2.64 ± 0.35	2.40	10
T-24'	Zoom/Pan task	M,K,R ₁ ,P,K,R ₁	265	3.70 ± 0.54	3.75	-1
T-25	Zoom/Pan task	K,R ₁ ,P,K,P _S ,K,R ₁	60	4.06 ± 0.54	3.65	11
T-26'	Zoom/Pan task	M,K,M,P,K,P _S ,K,R ₁	58	6.54 ± 0.95	6.10	7
T-27	Start parcel-id digitizing with command selection	M,P,K,R ₁ ,H,M,5K,H,P _S ,K	10	9.00 ± 0.46	8.50	6
T-28	Continuous parcel-id digitizing	M,K,5K,P _S ,K	38	5.08 ± 0.37	5.00	2
T-29	Continuous parcel-id digitizing	M,K,H,5K,H,P _S ,K	30	6.43 ± 0.74	5.80	11
T-30	Continuous parcel-id digitizing	M,K,H,4K,H,P _S ,K	66	5.55 ± 0.79	5.40	3

Table 2: Comparison of measured and predicted performance times.

A * after the task number means that this task is performed after a zoom/pan task.

A ' after the task number means that a mental operator is added.

- 97のユニットタスクを同定
- 90%の説明率(9,394/10,447)、うち、エラーなしなのは85%(7,985)
- トップ30のユニットタスクで83%のタスクを説明(6,621/7985)

タスクの最適化による経費節減

T-23:ズーム／パンタスク	時間	
K	0.40	コマンドウィンドウを開くためにカーソルボタン“9”を押す
R ₁	0.25	システムがウィンドウを開く
P	1.10	ズーム／パンコマンドをポイントする
K	0.40	ボタンを押す
R ₁	0.25	システムが画面を更新する
合計	2.40	

最適化されたT-23	時間	
K	0.40	ズーム／パンコマンドが割り当てられたボタンを押す
R ₁	0.25	システムが画面を更新する
合計	0.65	

効果

総予算 \$12M 12億円

作業の 3／4 はルーチンタスク

T-23が占める時間の割合はその中の11% ⇒ \$1M(1億円)

最適化により73%の改善 ⇒ \$730,000