

Operation Competition Model: 非熟練者における作業の多様性

大野 健彦

NTT 基礎研究所

〒 243-0198 神奈川県厚木市森の里若宮 3-1
E-mail: takehiko@rudolph.brl.ntt.co.jp

一般に非熟練者がコンピュータを用いて作業をおこなう場合、その作業方法は熟練者に比べて個人差が大きい。そのため、これまで非熟練者の作業を詳細に予測することは困難であった。本研究では Operation Competition Model(OCM)と呼ぶ複数オペレータ間の競合モデルを提案する。OCMはオペレータが選択される度合いを変化させることで多様な作業を説明する。その結果、ユーザが持つ作業知識の差だけでは説明できなかった作業の多様性を簡易に取り扱うことが可能となった。本稿ではOCMの概要を述べると共に、コピー打鍵作業を題材としてユーザの作業を詳細に調べ、その結果に基づいた操作系列の生成例を示す。

Operation Competition Model: Managing the Variety of Non-Expert Users' Task

Takehiko Ohno

NTT Basic Research Laboratories

3-1 Morinosato Wakamiya, Atsugi, Kanagawa, 243-0198 JAPAN

When non-expert users operate the computer to accomplish some tasks, they use variety of methods. Therefore, it is difficult to predict their operations in detail. I propose the Operation Competition Model (OCM) which describes the competition among a set of operators. It produces the variety of non-expert user's operations by changing the parameter of each competitive operator. OCM has a potential to predict the various types of tasks which can not be explained by the difference of the non-expert users' task knowledge. In this paper, transcription typing task is selected as the example of the non-expert users' task. The detail of the OCM is described at first. Then, the result of the transcription typing is shown and the sequence of operations which is produced by OCM is shown.

1 はじめに

人はどのようにコンピュータを用いて作業をおこなうのだろうか。新しいコンピュータシステムやアプリケーション、あるいは優れたインタフェースを開発する上で人の行動を理解することは非常に重要な役割を果たす。特に最近ではコンピュータユーザの利用者層が大きく拡大し、初心者、中級者のユーザが増大しており非熟練者の作業を理解する重要性が高まりつつある。

これまでユーザの行動を説明、予測するために様々な認知モデルが提案されてきた。これまでの対象はもっぱら熟練者に限られてきたが、非熟練者についてもその作業を説明、予測することが可能となれば、熟練者と同様に様々な作業予測が可能となる。

コンピュータを用いた作業では、ユーザは自分の持つ様々な知識を利用して作業を進める。したがってユーザの行動を調べるためには、ユーザが作業に用いる知識を明らかにすれば良いと思われる。しかしながら実際にユーザの作業を観察すると個人差が非常に大きく、また同一のユーザにおいても必ずしも毎回同じ作業をおこなうわけではない。

このような作業の多様性はその一部分をユーザの持つ作業知識から説明できるが、それだけでは十分に説明できない。またこのような作業を高い精度で直接予測することは極めて困難である。

本研究では非熟練者の詳細な作業予測を可能とする認知モデルの構築を目標とする。本稿ではその前段階として、非熟練者の作業が一般にばらつきが多い点に着目する。そしてユーザの持つ作業知識に加えてオペレータ間の競合という概念を導入して、ユーザによって異なる一連の作業手順(ここでは操作系列と呼ぶ)を生成することが可能な工学的モデルを提案する。

1.1 作業知識と分析粒度

ユーザの作業を分析する場合、様々な分析の粒度が考えられる。例えば熟練者の定型的な行動を分析、予測するモデルとして有名なGOMSは複数の分析レベルを提案している[2]。GOMSはユーザの作業をゴール、オペレータ、メソッド(1つのゴールに対応するオペレータ集合)、選択規則を用いて記述し、利用者の持つ作業知識はこれら4種類の集合で説明される。また、ユーザの操作系列をこれら4種類の要素集合として記述することが可能である。

GOMSにおいてゴールおよびオペレータでどの程度詳細な作業を記述するかは分析レベルによって異なる。Unit-task levelは最も粒度の大きな分析レベルであり、ユー

ザの作業目標を記述する。またKeystroke levelはもっとも粒度が細かいレベルであり、ユーザの物理的な手や目の運動を予測する。Functional levelおよびArgument levelはこれらの中間レベルの分析粒度を持つ。

実際のユーザの作業をGOMSを用いて分析した場合、Unit-task levelによる分析では同一の知識を持つユーザ間では同一の操作系列が生成される。操作系列の差が顕著となるのは最も粒度の細かいKeystroke levelである。したがってユーザの作業差に注目する場合、Keystroke levelに相当する粒度の細かい分析が必要となる。

1.2 非熟練者の認知モデル

非熟練者の作業を予測することは以下の2点の理由から困難である。(1)非熟練者の作業は定型化しておらず、ばらつきが多い。(2)非熟練者の作業分析が難しい。例えば非熟練者は作業をおこなうために外界に存在する様々な情報を利用する。この時、作業の観察結果からは、ユーザがどの情報を元に作業をおこなったかを特定することは難しい。

しかしながら、非熟練者の作業を明らかにすることが可能となれば様々な利用方法が考えられる。そこでこれまでに様々な認知モデルを利用した作業分析および予測の研究がおこなわれてきた。例えばRiemanは探索的学習(exploratory learning)に関する認知モデルを提案している[10]。探索的学習とは試行錯誤、他者からの助言、マニュアル等の参照を通じた問題解決と学習行動を組み合わせた概念であり、Riemanらは初心者が複数の階層メニューの中から目的のメニューを選択する場面を対象としたコンピュータシミュレーションをおこなっている。シミュレーションにはプロダクションシステムに基づいた汎用認知機構であるSoar[8]を利用している。

また、Kitajimaはコンピュータのユーザが他者からの説明を理解するプロセスを対象とした認知モデルLICAIを提案している[7]。ユーザが新しい作業あるいはめったにおこなわない作業をおこなうとき、マニュアルなどの外界情報を利用して問題解決をおこなう。LICAIは外界情報を利用する時におこなう、理解作業と特定の作業に依存した説明を利用する作業の2種類をシミュレーションして、作業者にとって外界情報がどの程度有益な情報であったかを示すことに成功している。

人の作業はしばしば非合理的であるが、実は認知的負荷を軽減するためにそのような作業をおこなっている場合がある。Kirshはこのような非合理的作業をepistemic actionと呼んでいる[6]。Kirshは熟練者がコンピュータリアルタイムゲームの一種であるテトリスで遊ぶ場面を観察し、熟練者の作業の一部が熟練者の持つ作業知識か

表 1: コピー打鍵作業における被験者

被験者	キー位置に関する知識	タッチタイピング	日常の利用
A, B, C, D	有	可能	利用
E	有	可能	利用
F	有	不可	利用せず
G, H, I, J, K	無	不可	利用せず

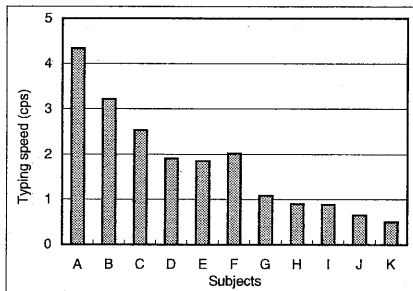


図 2: 被験者の平均打鍵速度

実験条件 実験には日本語エディタ GNU Mule 2.3 を利用した。Mule には打鍵記録を採集するように改良を加えてある。Mule の画面は左右に 2 分割されており、被験者は画面左側に表示された例文を画面右側に入力する。打鍵誤りについては、実験者は被験者に実験を開始する前に、気付いた時点で後退キーを用いて誤りを削除し、新しく文章を入力するように指示を与えた。

実験にはワークステーション Sun SS-10 を用いた。打鍵作業は Mule で記録した。また同時にアイカメラ EMR-NC で被験者の視線を記録した。EMR-NC は NTT 基礎研究所および NAC が開発した非接触型アイカメラ¹である。EMR-NC は 30Hz で視線記録をおこない、また測定精度は視野角 0.3 度である。

実験手順 実験の手順は以下の通りである。(1) 練習試行。5 分間のコピー打鍵をおこなう。(2) 本試行。時間に制限は設けず、5 セッションのコピー打鍵をおこなう。各セッションでは 30 単語から 55 単語の英文が表示される。5 セッションでの合計は 205 単語、1207 文字である。

実験結果 練習作業における各被験者の打鍵速度を図 2 に示す。被験者 A から D はいずれもタッチタイピング可能であると答えたが、打鍵速度は大きく異なる結果となつ

¹EMR-NC の詳細は <http://www.brl.ntt.co.jp/info/kameg/emr-nc/> を参照。

表 2: コピー打鍵実験における認知オペレータの回数

被験者	acquire chunk	verify chunk	find key	remove chunk
A	106	59	9	24
B	199	98	16	48
C	252	188	148	65
D	323	257	135	83
E	313	213	38	30
F	217	120	209	16
G	345	302	304	99
H	602	123	572	22
I	484	266	386	39
J	432	350	342	25
K	399	314	309	17

た。もっとも打鍵速度の速い被験者 A は 4.3 文字 / 秒、もっとも遅い K は 0.5 文字 / 秒であった。

3.3 オペレータ

実験結果を元に、各被験者について作業記録から知覚オペレータおよび運動オペレータを求め、対応する認知オペレータを決定した。被験者 A の第 1 セッションにおける作業例を図 3 に示す。図 3 は並列する複数オペレータ間の依存関係を記述する CPM-GOMS[3] の分析方法を利用しており、運動オペレータ、認知オペレータ、知覚オペレータの 3 種類の依存関係を記述している。

コピー打鍵の実験結果から求めた各被験者における認知オペレータの出現回数を表 2 に示す。これらのオペレータはユーザの打鍵記録および視線記録から求めている [9]。Acquire-chunk は例文からの文字列取得を、verify-chunk は入力済みの文字列の確認を、find-key はキー位置を取得するためにキーボードを見る作業を、remove-chunk は入力した文字列の削除を示している。

被験者 A から E まではキー位置に関する知識を持つため、find-key の回数が A で 9 回、もっとも多い C でも 148 回と少ない。一方、F から K は平均 353.7 回と多く、1 単語を入力するのに平均 1.7 回 (ただし誤打鍵の修正を考慮しない) キーボードを見ている。

3.4 作業知識

コピー打鍵に必要なユーザの知識を図 4 に示す。知識 G1 から G5 はコンピュータを用いた作業における汎用知識である。G1 から G4 まではどのユーザも保持している。G5 はキー位置に関する知識であり、実際には各キーに対応した知識が存在する。また、G5 を持たないユーザは、G1 および G4 を用いてキー位置に関する情報を取得する。T1 および T2 はコピー打鍵において必要とな

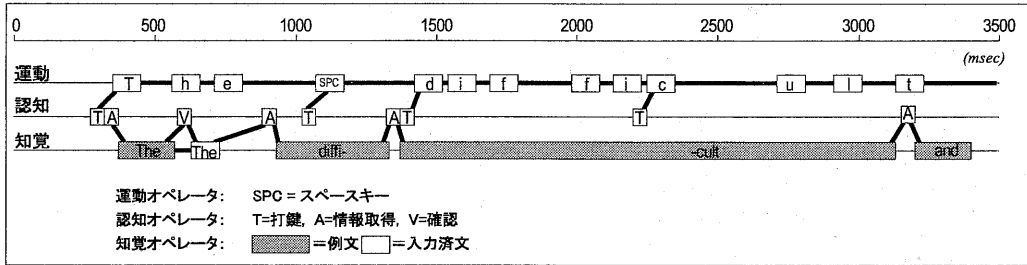


図 3: 被験者 A の作業

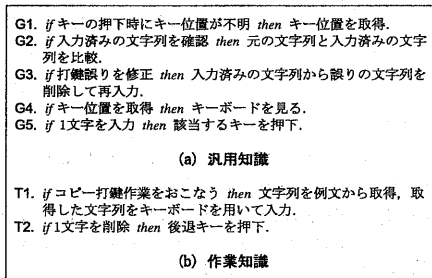


図 4: コピー打鍵作業におけるユーザの知識

る作業知識である。これらはコピー打鍵をおこなう前の説明によって容易に取得可能な知識であり、どのユーザも持っていると考えて良い、一般的な打鍵作業ではエディタの操作などに関する知識が必要になる場合もあるが、今回の実験でおこなった作業においては不要である。

4 Operation Competition Model による操作系列の生成

表 2 に示した通り、被験者によって利用したオペレータの種類は大きく異なり、その操作系列も個人差が大きかった。一方、作業に必要な知識の種類は表 4 に示す通りであり、これらの知識の有無からは操作系列の多様性を説明することはできない。そこで OCM を利用して様々な操作系列の生成をおこなう。

オペレータ コピー打鍵において利用するオペレータは表 2 に基づく。また、作業知識および実験結果のデータ分析に基づいて決定した競合状態にあるオペレータを図 5 に示す。シミュレーションでは被験者 A および K の操作系列を参考に、(1) から (3) までの 3 種類の競合状態に対して以下の通りパラメータを決定した。

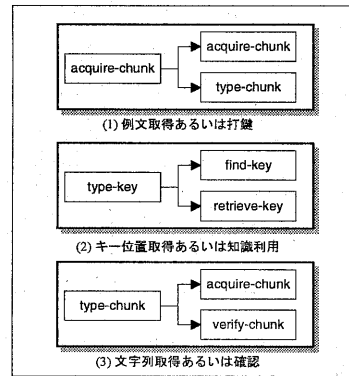


図 5: コピー打鍵作業におけるオペレータの競合状態

No.	オペレータ	ケース A	ケース B
(1)	acquire-chunk	10	20
	type-chunk	20	10
(2)	find-key	1	10
	retrieve-key	10	1
(3)	acquire-chunk	20	20
	verify-chunk	10	10

操作系列の生成結果 ケース A およびケース B について OCM によって生成された操作系列を図 6 に示す。図 6 は作業の最初の一部分であり、"The difficult and divisive" と入力する場面を表わしている。ケース A では 1 単語ずつ取得した文字列をキーボードを見ずに入力しているのに対し、ケース B では予め 2 単語を取得した上で一文字ずつキーを見ながら入力している。

5 議論

本論文では OCM の利用例として打鍵作業を題材としたが、モデルの適用範囲は打鍵作業に留まらない。例えば

1 C: acquire-chunk	1 C: acquire-chunk
1 P: look-example 'The'	1 P: look-example 'The'
2 C: type-chunk 'The'	2 C: acquire-chunk
3 C: retrieve-key 'h'	3 P: look-example 'difficult'
3 M: type-key 't'	3 C: type-chunk 'The'
4 C: retrieve-key 'h'	3 C: type-chunk 'difficult'
4 M: type-key 'h'	4 C: find-key 't'
5 C: retrieve-key 'e'	4 P: move-eyemark keyboard
5 M: type-key 'e'	4 P: keyboard
6 C: acquire-chunk	4 M: type-key 't'
6 P: look-example 'difficult'	5 C: find-key 'h'
7 C: type-chunk 'difficult'	5 P: keyboard
8 C: retrieve-key 'd'	5 M: type-key 'h'
8 M: type-key 'd'	6 C: find-key 'e'
9 C: retrieve-key 'i'	6 P: keyboard
9 M: type-key 'i'	6 M: type-key 'e'
10 C: find-key 'f'	7 C: find-key 'd'
10 P: move-eyemark keyboard	7 P: keyboard
10 P: keyboard	7 M: type-key 'd'
10 M: type-key 'f'	8 C: find-key 'i'
11 C: retrieve-key 'f'	8 P: keyboard
11 M: type-key 'f'	8 M: type-key 'i'
12 C: retrieve-key 'i'	9 C: find-key 'f'
12 M: type-key 'i'	9 P: keyboard
13 C: retrieve-key 'c'	9 M: type-key 'g'
13 M: type-key 'c'	10 C: find-key 'f'
14 C: retrieve-key 'u'	10 P: keyboard
14 M: type-key 'u'	10 M: type-key 'g'
15 C: retrieve-key 'l'	11 C: find-key 'i'
15 M: type-key 'l'	11 P: keyboard
16 C: retrieve-key 'c'	11 M: type-key 'i'
16 M: type-key 'c'	12 C: find-key 'c'
17 C: verify-chunk	12 P: keyboard
17 P: move-eyemark typed	12 M: type-key 'c'
17 P: look-typed-sentence	13 C: find-key 'l'
18 C: acquire-chunk	13 P: keyboard
18 P: look-example 'and'	13 M: type-key 'u'
19 C: acquire-chunk	14 C: find-key 'l'
19 P: look-example 'divisive'	14 P: keyboard
20 C: acquire-chunk	14 M: type-key 'l'
20 P: look-example 'national'	15 C: find-key 't'
21 C: type-chunk 'and'	15 P: keyboard
21 C: type-chunk 'divisive'	15 M: type-key 't'
21 C: type-chunk 'national'	16 C: verify-chunk
	16 P: move-eyemark typed
	16 P: look-typed-sentence
	17 C: acquire-chunk
	17 P: look-example 'and'
	18 C: acquire-chunk
	18 P: look-example 'divisive'
	19 C: acquire-chunk
	19 P: look-example 'national'
	20 C: type-chunk 'and'
	20 C: type-chunk 'divisive'
	20 C: type-chunk 'national'
	21 C: find-key 'a'

(a) Case A

C: Cognitive operators
P: Perceptual operators
M: Motor operators

(b) Case B

図 6: Operation Competition Model により生成された操作系列

Graphical User Interface では目的を達成するためのインタフェースの操作方法は一意ではなく、ユーザによって様々な操作をおこなう。このような場面において、それぞれの操作をオペレータの競合状態とすることで、選択する操作方法の違いと操作系列の関係を調べることが可能である。

作業の熟達化 OCMは競合状態にあるオペレータ間のパラメータを変化させることで、様々な習熟度のユーザをシミュレーションすることを可能としている。図6では2通りのシミュレーション例を示したが、この中間段階についてもパラメータを変化させていくことでシミュレーションすることが可能である。予測した結果の妥当性については、今後評価をおこなっていく予定である。

認知モデルとの関係 OCMは工学的な利用を想定したモデルであるが、これまでに提案されてきた認知モデルを背景とする部分もある。競合する複数のオペレータについて、各オペレータが持つパラメータを元に選択されるパラメータを決定するという方法は、プロダクションシステムに基づく汎用認知モデルのACT*[1]におけるプロダクションの強さ(strength)と同様の概念である。ACT*では強さの大きいプロダクションほど選ばれやす

く、またプロダクションの実行結果に応じて動的に強さの値が変化する。またOCMで利用している3種類のオペレータは、Model Human Processor[2]およびCPM-GOMS [3]で用いられており、人がコンピュータを利用する場面を記述するのに適している。

6 まとめ

本論文では、非熟練者の操作系列を生成するOperation Competition Modelを提案した。OCMは競合状態のオペレータを予め定義することで、多様な操作系列を容易に生成することを可能としている。現在は競合状態にあるオペレータを作業の観察によって求めるなど手作業による部分が多いが、今後はモデルを拡張して自動的に予測することの可能な枠組みを導入することを検討している。

参考文献

- [1] Anderson, J. R.: *The Architecture of Cognition*, Harvard University Press (1983).
- [2] Card, S. K., Moran, T. P. and Newell, A.: *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum Associates (1983).
- [3] Gray, W. D., John, B. E. and Atwood, M. E.: *Project Ernestine: Validating a GOMS Analysis for Predicting and Explaining Real-World Tasks Performance*, *Human-Computer Interaction*, Vol. 8, pp. 237-309 (1993).
- [4] John, B. E.: *TYPIST: A Theory of Performance in Skilled Typing*, *Human-Computer Interaction*, Vol. 11, pp. 321-355 (1996).
- [5] John, B. E. and Newell, A.: *Cumulating the Science of HCI: From S-R Compatibility to Transcription Typing*, *CHI'89*, pp. 109-114 (1989).
- [6] Kirsh, D. and Maglio, P.: *On Distinguishing Epistemic from Pragmatic Action*, *Cognitive Science*, Vol. 18, pp. 513-594 (1994).
- [7] Kitajima, M. and Polson, P. G.: *A Comprehension-Based Model of Exploration*, *Human-Computer Interaction*, Vol. 12, pp. 345-389 (1997).
- [8] Laird, J. E., Newell, A. and Rosenbloom, P. S.: *Soar: An Architecture for General Intelligence*, *Artificial Intelligence*, Vol. 33, pp. 1-64 (1987).
- [9] 大野健彦: コピー打鍵における習熟過程のモデル化を目指して、日本認知科学会第13回大会論文集, pp. 32-33 (1996).
- [10] Rieman, J., Young, R. M. and Howes, A.: *A Dual-Space Model of Iteratively Deepening Exploratory Learning*, *International Journal of Human-Computer Studies* (1995).
- [11] Salthouse, T. A.: *Perceptual, Cognitive, and Motoric Aspects of Transcription Typing*, *Psychological Bulletin*, Vol. 99, pp. 303-319 (1986).