

UNIX 高機能シェルの行編集機能に対する適応型 ヒューマンインタフェースの構築とその評価

高 田 光 男^{†,☆} 西 野 順 二^{††}
小 高 知 宏^{††} 小 倉 久 和^{††}

ソフトウェア・システムが有する機能の効果的活用に対する支援を目的として、ダイナミックな挙動解析による能動的な支援、および、利用者それぞれの個性に着目した個人適応型の支援を実現する方法を提案する。また、個人適応型の知的インタフェースの構築の際、各ユーザの挙動をモデル化したユーザモデルを構築する手法を提案する。我々は、UNIX のシェルの 1 つである tcsh のコマンドライン編集機能を対象に、本手法による支援を実際に行った。この結果、ユーザの使用コマンド数の増加傾向、システムの支援頻度を観測・分析することで、能動的な支援および個人適応型の支援の有効性が示唆された。

Implementation of Adaptive Human Interface on UNIX High Performance Shell and Its Evaluation

MITSUO TAKADA,^{†,☆} JYUNJI NISHINO,^{††} TOMOHIRO ODAKA^{††}
and HISAKAZU OGURA^{††}

We report a study on adaptive on-line assistance methods and assistance system for interactive software such as the UNIX shell program. The system analyzes shell user's behavior dynamically, then constructs the user model which describes the characteristics of different classes of users. According to the user model, the system can show adaptively the help information of the editing commands of the 'tcsh' which is one of the UNIX shell program. We evaluated the assistance methods experimentally.

1. はじめに

あるソフトウェア・システムを使い始める場合、ユーザの多くは必要最小限の機能を身に付けた時点でシステムを使い始める傾向にある。システムが多くの機能を持つようになってきているため、あらかじめすべての機能を身に付けてからシステムを使い始めるのは現実的ではない。システム開発者はユーザがシステムを使いつつ次第に高度な機能を身に付けていくことを期待しているが、実際には、ユーザはシステムの提供する有用な機能の存在すら知らずに単純な機能のみを使い続ける傾向がある。システムの提供している有用な機能が、ユーザに使われないままになるという問題も

報告されている^{1),2)}。たとえば、UNIX のシェルの 1 つである tcsh は、コマンドライン上において Emacs あるいは vi エディタと同様な編集が可能であり、約 40 種類の編集コマンドを有している^{3),4)}。しかし、その使用状況については、実際にユーザが使用している編集コマンドは半数にも満たないことが我々の調査で確認できた^{5),6)}。

従来、ヒューマンインタフェースの研究は、システムに多くの高度な機能を持たせることで、そのシステムのパフォーマンスを向上させようとする方向でなされてきた。しかし、現存のソフトウェア・システムのパフォーマンスを改善するには、ユーザのシステム使用上のスキルを向上させることも有効な手段である。この観点に基づき、本論文ではユーザのスキル向上を促進する一手法として、ユーザが未知の機能を学習していくことを支援するオンライン支援を提案する。

我々は実際に、tcsh のコマンドライン編集機能に関する支援が可能なオンライン支援システムを構築した。これは、システムとユーザの間に立ち、必要に応

† 福井大学大学院工学研究科

Graduate School of Engineering, Fukui University

☆ 現在、アトラス情報サービス株式会社

†† 福井大学工学部情報工学科

Department of Information Science, Faculty of Engineering, Fukui University

表1 tcshの編集コマンドの使用状況
Table 1 Use of tcsh commands.

使用歴	全コマンドに対する使用割合
2年6カ月	42.5%
1年7カ月	47.5%
1年5カ月	45.0%
6カ月	27.3%
4カ月	36.4%
2カ月	15.2%

じて、ユーザが未知の機能の存在と使用法を積極的に提示する知的支援システムである。本論文では、ソフトウェア・システムがユーザに対して必要に応じて支援を行うことを、システムが自律的に支援を行うという意味を込めて、能動的な支援と呼ぶ。これは、システムがユーザの指示によって指示された内容だけを支援する、受動的な支援と対立する概念である。

本論文では、オンライン支援システムを構築する際の手法として、ユーザの挙動をダイナミックに解析することによる能動的な支援、さらに利用者それぞれの個性に着目した個人適応型の支援が可能なシステムの構築法を提案する。また、実際にシステムを試作し支援を行った結果をもとに、能動的な支援および各ユーザへの適応の効果について論ずる。以下、2章では本システムの構成と実装について述べ、3章では実際に本システムによる支援を行った結果を示す。4章で支援結果に対する考察を行い、最後に5章で本論文のまとめと今後の課題について述べる。

2. 知的オンライン支援システムの構成と実装

2.1 個人適応型支援システム

表1は、使用歴の異なる6名のtcshユーザを対象に、tcshが持つコマンドライン編集機能の使用状況を調査した結果である^{5),6)}。これは、各ユーザから採取した約1カ月間の操作履歴をもとに、ユーザが使用した編集コマンドの個数の全コマンド数に対する割合を示したものである。この結果から、実際にユーザが使用している編集コマンドはtcshが持つコマンドの半数にも満たない場合があることが分かる。使いやすいシステムを構築していくためには、システムの機能拡張を行うだけでなく、同時に、ユーザのシステム使用上のスキルの向上を促進することも考慮する必要がある。また、調査期間内にユーザはマニュアル等を参照するなどして新しい機能を習得することも可能であったことを考慮すれば、ユーザ自身の自発的な学習を期待するだけでは、ユーザのスキル向上は望めないことも表1は示唆している。そこで、システム側からユーザに対し積極的に働きかけ、学習のきっかけを作るこ

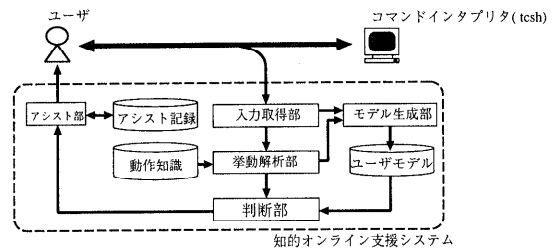


図1 システムの構成
Fig.1 System structure.

とで、ユーザがシステムの有する機能を幅広く活用できる環境を作り、作業効率を増大させることを目指す。

我々はtcshのコマンドライン編集機能に関して、ユーザの未知の機能を能動的に提示できる知的オンライン支援システムを作成した。本システムは、効果的にユーザのスキル向上を促進するためには、ユーザの学習しやすいタイミングで、しかもユーザの直観に訴えかけられるような支援が必要であることを考慮し、ユーザの編集動作に関連した支援情報を必要に応じて随時提示する。そのため、本システムではユーザの挙動に関してダイナミックな解析を試みることで、つねにユーザの置かれている状況を把握し、編集操作に関する評価を行っている。

本支援システムは、画一的な仮想ユーザを対象としたインタフェースではなく、ユーザの理解状態に柔軟に適応し、適切な支援情報を提示することのできる適応型のインタフェースを有している。これは、ユーザごとに対応を適応させることによって、各個人の能力差を考慮した支援をすることを目的としている。そのため、本システムではユーザの編集操作を解析することで、各ユーザの理解状態をモデル化し、各個人の能力あるいは経験を反映させることにより適応型の支援を行う。

本システムの概略を図1に示す。ここで、ユーザのシェルに対する一連のキー入力からその動作を解析・評価する部分が挙動解析部であり、モデル化された各ユーザごとの理解状態に関する情報が蓄えられている部分がユーザモデルである。本システムでは、ユーザのシェルに対する一連のキー入力をリアルタイムに取得することでダイナミックにユーザの挙動を解析し、その編集動作の評価を行う。そして、この評価の結果とユーザモデルからの個人情報とを統合することで、そのユーザの能力あるいは経験に応じたレベルのコマンドを提示し、その説明を行う。

2.2 動的なユーザ挙動解析

挙動解析部では、tcshにおけるユーザのコマンド行

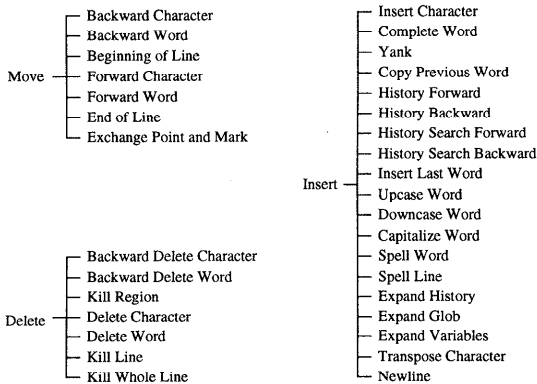


図2 挙動の分類

Fig. 2 Classification of behavior.

での編集内容を把握し、その編集方法が適切であるか否かの評価を行うことで、ユーザに支援すべき編集コマンドの候補を推論する。本システムでは、この解析をダイナミックに行い、ユーザが利用している間、つねに、ユーザ挙動の推論とその評価を行う。この部分は大きく“編集内容の把握”および“編集方法の評価”からなる。

“編集内容の把握”とは、ユーザの1連の編集動作を図2に示した33種類の動作要素の組合せとして置き換えることである。ここで、図2にあげた動作要素は十分粒度の細かいものであり、これらの組合せによりユーザのあらゆる挙動を表現することが可能である。この部分では、ユーザの一挙動に関して、動作要素と使用したコマンドの組を生成する。

ユーザの挙動を解析するにあたりシステムが得ることのできる情報は、ユーザのキー入力をリアルタイムで取得することから得られるコマンドライン上の文字列とカーソル位置の時間的变化、および使用した編集コマンドである。さらに、本システムでは各編集要素について、その編集を行った場合にはコマンドライン上の文字列およびカーソル位置がどのように変化するか、各編集を行うためにユーザが使用する可能性のあるコマンドは何かというコマンドの動作に関する知識を持っている。図1に示したように、この知識を本論文では動作知識と呼ぶ。動作知識は図3に示すような形式と内容を有する。たとえば、「カーソルをコマンドラインの先頭へ移動 (beginning-of-line)」という動作に関しては、図3に示したように、コマンドライン上の文字列に変化はなく、その動作が完了した時点でのカーソルの位置は先頭であり、ユーザが使用する可能性のあるコマンドは‘beginning-of-line’、‘backward-word’、‘backward-char’、‘exchange-point-and-mark’のい

動作要素名	: Beginning of Line
文字列の変化	: なし
カーソル位置の変化	: 開始時 任意 , 終了時 先頭 (0)
使用可能コマンド	: beginning-of-line, backward-word, backward-char, exchange-point-and-mark
備考	: なし

図3 各動作の知識表現

Fig. 3 Knowledge expression of each manners.

れかであるという形式の知識である。

システムはこれらの取得情報および知識から、ユーザの連続する編集動作を大きく、カーソル移動 (Move)、文字入力 (Insert)、文字消去 (Delete) の3種類のモードに分け、そのモードの変化点において、それまでの挙動を図3の動作要素のいずれかとして同定する。これは、取得した入力系列から、ある挙動が開始し終了するまでのコマンドライン上の文字列とカーソル位置の変化を抽出し、あらかじめシステムが持っている各コマンドの動作に関する知識と比較することで可能である。そして、この結果とユーザが実際に使用したコマンドの組を解析結果として「動作要素-使用コマンド」といった形式で次の“編集方法の評価”へ渡す。

“編集方法の評価”では、“編集内容の把握”の部分において生成された動作要素と使用したコマンドの組に関して、ユーザの使用したコマンドが最も合理的なコマンドであるか否かを評価する。この評価を行うために、システムは各編集動作に関する最も合理的なコマンドを「動作要素-合理的なコマンド」という形で知識としてプログラム中に表現した。ここで、最も合理的なコマンドとは、ある編集を行う際、最小のキーストロークで実行可能なコマンドのことであると本研究では定義する。ある編集動作に関して、ユーザが実際に使用したコマンドとシステムが知識として持つ合理的なコマンドを比較し、一致しなければ、合理的なコマンドを支援の候補として取り上げる。

2.3 ユーザモデルの構成

本システムでは、各利用者ごとの支援対象システムの利用履歴を解析しユーザモデルとして保存することで、利用者に適応した支援を行う。特に、画一的な支援を行う場合と比較して、ユーザモデルを用いることで学習効果を損なわずに支援回数を減らすことで、効率的に学習が進むことを目標とする。本システムでは、知的CAIの学習者モデルの1つであるオーバレイモデル^{7)~9)}を参考に図4のようなユーザモデルの構築を行った。オーバレイモデルとは、学習者の知識を教師の部分集合としてとらえるモデルであり、その構築方法は学習者が習得した項目にフラグを立てるといっ

コマンド名	得点	最終使用日	フラグ
backward-word	15	12/25	ON
backward-delete-word	13	12/23	ON
forward-word	2	12/1	OFF
delete-word	8	12/13	ON
beginning-of-line	1	11/28	OFF
history-backward	20	12/10	ON
end-of-line	3	12/25	OFF
		12/3	ON

図4 ユーザモデル
Fig. 4 User model.

た単純なものである。

本ユーザモデルにおいては tcsh の編集コマンドの一覧を用意し、システムの判断によりユーザが理解していると推論したコマンドにフラグを立てる。この判断は、各コマンドごとに付加した得点で行う。この得点は、ユーザが使用するたびに加点し、ユーザの使用したコマンドが合理的なコマンドではない場合には減点する。また、あるコマンドを長期間（たとえば1週間）使用していない場合にも減点する。得点があるかじめ設定した基準値を超えたコマンドが、ユーザの持つ知識であるとしてフラグを立てる。たとえば、ユーザが後方の単語を消去するコマンド “backward-delete-word” を設定回数だけ使用すれば、本ユーザモデルでは、そのコマンドはユーザの持つ知識としてフラグが立つ。しかし、後方の単語を消去する際に “backward-delete-word” を使用しなかったり、長期間使用しなかった場合は、“backward-delete-word” に付加した得点が減り、設定した基準値より低くなればフラグの取消しを行う。このように、本ユーザモデルはユーザの使用状況に応じて変化する動的なものである。

2.4 支援項目決定戦略

挙動解析により、支援の候補として取り上げたコマンドが、そのユーザの能力・経験を考慮した場合、適当であると判断したうえでシステムは実際に支援を行う。この判断は、各個人ごとに蓄えてきたユーザモデルと、プログラム中に表現した tcsh の全編集コマンドに関する知識のモデルを参照することで行う。本システムでは、tcsh の編集コマンドは図5に示したような階層構造を持つという知識をもとに、支援の候補として選んだコマンドより基本的なコマンドをすべてユーザが知識として持つ場合に限り支援を実行する。図5では上方にあるコマンドほど基本的なコマンドを示している。このコマンドの序列は、著者が経験的に決定した。

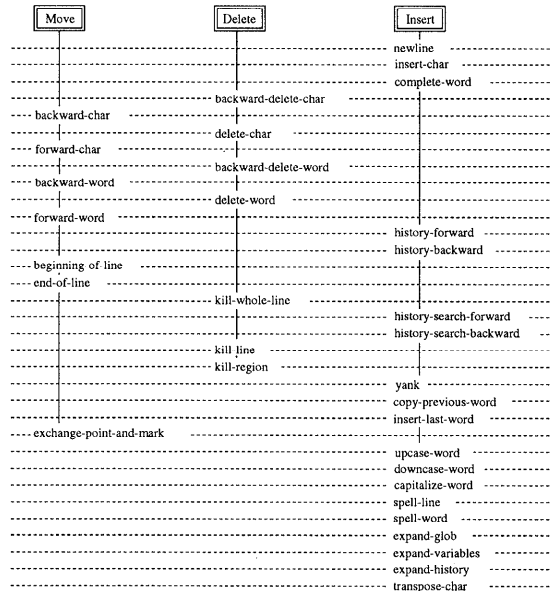


図5 コマンドの階層化
Fig. 5 Hierarchy of commands.

たとえば、挙動解析の結果、コマンドラインの先頭へカーソルを移動するコマンド “beginning-of-line” が支援の候補としてあげられた場合、このコマンドが最終的にユーザに提示されるのは、

- (1) ユーザモデル中で “beginning-of-line” にはフラグが立っていない。
- (2) ユーザモデル中で “beginning-of-line” より基本的なコマンド “backward-char”, “forward-char”, “backward-word”, “forward-word” にはフラグが立っている。

という条件を満たしている場合のみである。こうして、本システムによるオンライン支援では、各個人の能力に合わせ、基本的なコマンドから徐々に高度なコマンドを教授していくといった観点に立った支援を行っている。

3. 能動的支援と適応型ヒューマンインタフェースの評価

我々が提案する能動的なオンライン支援の有用性と、システムがユーザモデルを持ち、利用者ごとに対応を変化させることの必要性を評価するために、3つの評価実験を行った。

第一は、ユーザモデルの信頼性に関する予備実験である。本システムでは、ユーザモデルからの情報をもとに、各ユーザごとに個別支援を行うので、ユーザモデルは信頼性の高いものでなくてはならない。もし、信頼性の低いものであれば、ユーザモデルを参照する

ことが原因で支援に悪影響を及ぼす可能性がある。我々は予備的な実験として、実際に支援を行う前にユーザモデルの構築方法の検証を行った。

第二は、オンライン支援システムの効果を評価することを目的とした実験である。本システムによる学習支援が、ユーザのスキルの変化に与える影響を分析する。

第三は、ユーザモデルの役割を評価する実験である。本システムからユーザモデルを取り除いたシステムによる支援を行い、ユーザモデルの有無によるユーザのスキル向上の変化、および、支援回数の比較を行う。これは、第二の実験の対照実験であり、ユーザモデルを持たないこと以外は第二の実験とすべて同じ条件下で行った。この実験により、単に使うべきときに使っていないコマンドを提示するだけの方法による支援と、ユーザモデルを利用した適応型支援との差を分析する。

ここで、以上の3つの評価実験における被験者は、A～E, G (UNIX 使用歴3年, tcsh 使用歴1カ月未満), I, K (同4年, 同6カ月未満), F, J (同5年, 同1年未満), H (同6年, 同2年未満)の11名の学部学生および大学院生である。

3.1 ユーザモデルの評価

A～Kの11名のユーザを対象に、第一の実験を行い、ユーザモデル中で表現された知識とユーザ自身が実際に持つ知識を照合した。

「ユーザモデル中で表現された知識」とは、本システムが生成したユーザモデル中で、ユーザが知っているコマンドとしてフラグが立っているものである。本システムでは、動的なユーザモデルの構築を行っているが、実際にユーザの理解状態を把握するには、ある程度、期間が必要である。本実験では、ユーザモデルの構築を開始してから2カ月後のユーザモデルを比較の対象とした。一方、「ユーザ自身が実際に持つ知識」とは、全コマンドのリストから自分の知っているコマンドを申告してもらったものである。ただし、この調査はユーザモデルの構築を終了した直後に行い、使用方法を明確に口述できるコマンドのみを申告してもらった。

第一の実験による結果を表2にまとめる。これは、ユーザモデル中で表現された知識とユーザ自身が実際に持つ知識との相違を表したものであり、40種類の編集コマンドのうちシステムが誤認したコマンドの個数を表している。たとえば被験者Bについては、システムがユーザの知らないコマンドを知っているものと“高く”評価したコマンドが3個であり、逆に知っているコマンドを知らないものと“低く”評価したコマ

表2 ユーザモデルの検証結果

Table 2 Test of user model.

被験者	誤認コマンドの個数 [個]	
	高く評価	低く評価
A	1	1
B	3	1
C	1	0
D	1	0
E	2	0
F	2	1
G	0	2
H	0	2
I	0	2
J	0	2
K	0	0
平均	0.91	1.00

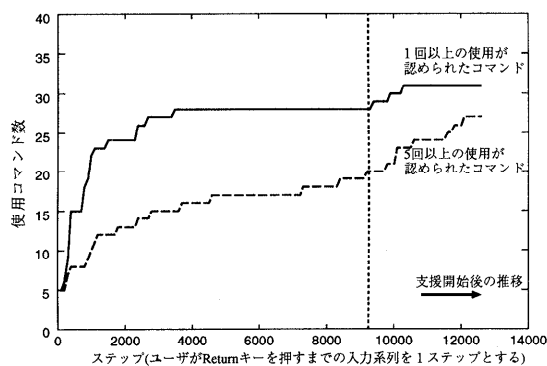
ンドが1個あったことを示している。いずれのユーザにおいてもシステムが誤認したコマンドが4個以下のユーザモデルが構築できた。また、被験者Dに関しては、ユーザモデルがユーザの知るコマンドを間違いなく把握した。平均では、ユーザの知識を“高く”誤認したコマンドが0.91個であり、“低く”誤認したコマンドが1.00個であった。つまり、tcshの編集コマンド40種類のうち平均2～3種類のコマンドに関して、実際の知識との相違がみられる程度である。

3.2 適応型支援による学習効果

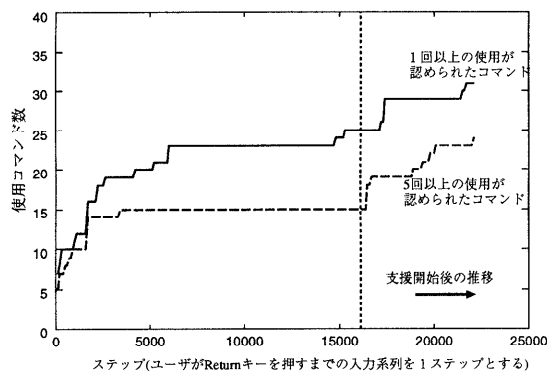
第二の実験では、ユーザの入力系列をすべて取得した。そのデータをもとにユーザが使用したコマンドの種類の変化を観察・分析し、支援の効果を検討した。また、本システムが実行した支援の回数を各ユーザごとにカウントし、支援の発生頻度を集計した。被験者として、B～E, G～Iの7名のユーザに協力を依頼した。また、観察期間は6カ月であり、最初の5カ月は支援を行わず、実際に本システムによるオンライン支援を行ったのは最後の1カ月のみである。

図6と表3が第二の実験の結果である。図6は、ユーザの使用したコマンドの累積度数の変化を表しており、実線は1回以上、破線は5回以上の使用が認められたコマンドの個数である。縦軸が、この個数を表している。横軸は、ユーザがシェルに対し入力したシェルコマンドのステップ数を表している。ただし、ユーザがReturnキーを打つまでの入力系列を1ステップとする。実際に学習支援を行ったのは、グラフ中の縦線以降である。

表3は各ユーザごとの支援回数をカウントし、その頻度を集計した結果である。たとえば被験者Dについては、ユーザがシェルに対し3,803ステップの命令を実行する間に、システムから102回の支援が行われ、



(a) 被験者 D における使用コマンドの数の時間的推移



(b) 被験者 E における使用コマンドの数の時間的推移

図 6 支援の結果 (ユーザモデルあり)

Fig. 6 Result of assistance with the user model.

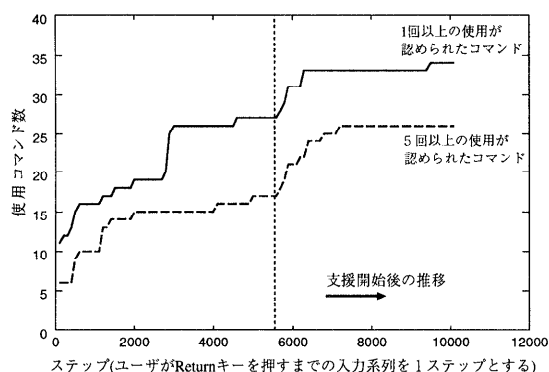
表 3 支援頻度 (ユーザモデルあり)
Table 3 Frequency of assistance with the user model.

	ステップ	支援回数	支援頻度 (ステップ/支援回数)
B	6,332	504	12.56
C	5,545	152	36.48
D	3,803	102	37.28
E	7,757	256	30.30
G	2,511	25	100.44
H	3,339	223	14.97
I	1,286	45	28.58
平均	—	—	37.23

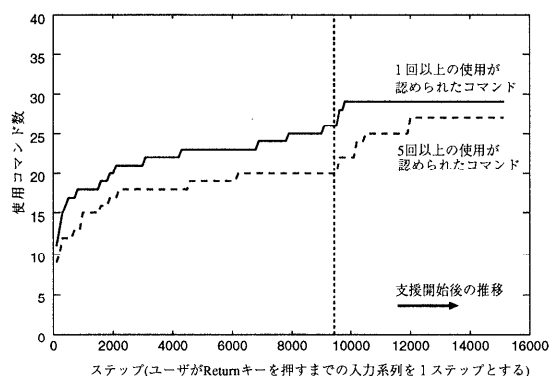
これは 37.28 ステップに 1 回の頻度で支援が行われたことを意味する。本実験により、平均 37.23 ステップに 1 回の頻度で支援が行われたという結果が得られた。

3.3 適応型ヒューマンインタフェースの評価

第三の実験では、本システムからユーザモデルを取り除き、各ユーザごとに対応を変化させることのできないシステムによる支援を行った。このときのユーザの使用コマンド数の変化と、システムが実行した支援の回数を観察する。被験者として、A, J, K の 3 名



(a) 被験者 A における使用コマンドの数の時間的推移



(b) 被験者 J における使用コマンドの数の時間的推移

図 7 支援の結果 (ユーザモデルなし)

Fig. 7 Result of assistance without the user model.

表 4 支援頻度 (ユーザモデルなし)
Table 4 Frequency of assistance without the user model.

	ステップ	支援回数	支援頻度 (ステップ/支援回数)
A	5,050	1,438	3.51
J	1,768	487	3.63
K	1,599	185	8.64
平均	—	—	5.26

のユーザに協力を依頼した。

第三の実験の結果を、図 7 と表 4 に示す。図 7 は図 6 と同様、ユーザの使用したコマンドの種類の変化を表しており、表 4 は表 3 と同様、システムが実行した支援の回数を表している。

表 4 に示したように、ユーザモデルを持たないシステムの支援回数の平均は 5.26 ステップに 1 回の割合である。これに対して、表 3 に示したユーザモデルを持つシステムの行う支援の回数は平均で 37.23 ステップに 1 回である。このように両者の支援回数を比較すると、ユーザモデルを持つシステムの支援回数は、ユーザモデルを持たないシステムの支援回数に対して、平

均で約 1/7 であった。また図 6 と図 7 を比較すると、ユーザモデルの有無による学習結果の差異はさほど大きくないことが分かる。たとえば、ユーザモデルを持つシステムの利用者である被験者 D と、ユーザモデルを持たないシステムの利用者である被験者 A を比較すると、被験者 A の方がアシスト開始後の使用コマンド数が多い。しかし被験者 E と被験者 J を比べると、ユーザモデルを持つシステムの利用者である被験者 E の方がむしろ習得コマンド数が多い。ただし学習の速度を比較すると、いずれの場合でもユーザモデルを持たないシステムの利用者の方が学習の速度が速い傾向も見られた。

4. 考 察

本論文で提案した能動的な支援は、ユーザのスキル向上を促進するうえで有効であった。これは、ユーザが使用したコマンドの種類が、能動的な支援により増加することを実験的に確認したことで示された。つまり、ユーザは本システムによる能動的な支援がなされるまでは特定のコマンド群しか使用していなかったが、支援以降はそれまで使用したことのないコマンドも使用するようになった。実際、被験者からも、いままで知らなかったコマンドを学ぶことができたことと肯定的な評価を得ている。

これまでに、ユーザが能動的に機能を学習していくことを支援するための研究は数多く行われてきた。その代表的なものとして、オンライン・ドキュメント・システム (OLDS) やチュートリアル・システムあるいは CAI などがあげられる^{10)~12)}。しかし、OLDS は、その場の一時的な問題解決には有効であるが、ユーザが新しい機能を知る手段としては不適當である。また、チュートリアル・システムや CAI などの学習支援システムでは、ユーザは短期間に多くの機能を習得しなければならず、その負荷を考えれば必ずしも有効な方法であるとはいえない。これらのシステムに対し、本システムではユーザの挙動に関してダイナミックな解析が可能であるため、ユーザがコマンドを実際に行っている局面に、コマンド利用の文脈に沿った形式で支援することができる。つまり、ユーザの動作に関連した支援情報を必要に応じて随時提示できる本手法による支援は、長期にわたってユーザが使用するコマンド数を増加させることで、効率的なコマンドの使用方法を支援できる。

本論文で提案したコマンドの階層構造やコマンドの序列については、今回の実験結果からはその正当性を検証することはできなかった。正当性を検証するため

には、別のコマンド体系を作成してユーザモデルに適用した場合と今回の実験結果を比較するなど、今回のものとは別の実験が必要になる。これは今後の課題である。

システムのユーザに対する支援頻度に関して、ユーザモデルなしの場合には表 4 に示すように平均 5.26 ステップごとに支援が行われたのに対して、ユーザモデルありの場合には表 3 に示すように平均 37.23 ステップごとに支援が行われているにすぎない。すなわち、ユーザモデルを用いると支援回数を約 1/7 に削減できる。支援の回数が 1/7 程度であるにもかかわらず、ユーザモデルの有無は習得したコマンド数に大きな影響は与えなかった。ただし、ユーザモデルを用いた場合には、学習の速度が遅くなる現象が見られた。その程度は、学習したコマンド数と学習に要した時間との比で考えると、最大でも 3 倍程度である。これは支援回数の比である 7 倍と比較すると小さい値である。このことから、ユーザモデルの能動的支援への導入は、支援を抑制し学習の効率を向上するために有用であると考えられる。すなわち、ユーザモデルに記録されたユーザの挙動解析結果と、それを用いた支援項目決定戦略による支援の抑制が有効に機能した結果、ユーザに負担を与えずに支援が実行できたと考えられる。

本論文で提案した個人の能力・経験を考慮した適応型の支援システムでは、ユーザにとって必要な支援項目をシステムが取捨選択したうえで提示する。このため、ユーザモデルを持つ適応型のシステムにおける支援は、ユーザにとってシステムからの支援を煩わしく感じない程度の頻度であった。このことは、支援頻度に関する否定的な意見を出す被験者がいなかったことから推察できる。つまり、ユーザモデルを持つ適応型のシステムでは、ユーザに負担をかけることなく、効果的に支援を行うことが可能である。逆に、ユーザモデルを持たないシステムでは、支援された項目から、ユーザ自らが必要なものを選択しなければならない。そのため、ある程度の機能を学び取ったユーザにとって、システムからの支援の多くは自分の知っているものであり、その中から知らないものを選択するのは困難となり、次第に支援内容に関心を示さなくなった。また、ユーザモデルを持たないシステムからの支援を受けた被験者の中には、システムから頻繁に支援されたが、その内容に注意を払うことはほとんどないという否定的な意見を出す者もいた。つまり、ユーザモデルを持たないシステムによる支援では、その多くがユーザにとって不必要であることを示しており、ユーザに余計な負担を与えかねない。学習だけを目的とし

た CAI システムなどでは、同じ事項を繰り返し提示されても学習が阻害されない場合も考えられるし、むしろ学習が強化される場合もありうる。しかし、本システムが対象としているような実用システムにおける支援では、利用者は学習を目的としてシステムを利用するわけではない。したがって、本実験におけるユーザモデルなしの場合の実験結果に示されるように、必要以上に情報提示を繰り返すことはむしろ適正な支援を阻害するものと考えられる。以上のように、ユーザモデルを利用した適応型支援は単純に使うべきコマンドを提示する方法と比較して、平均 1/7 程度の支援回数で同様の学習効果を得ることができる。このことは、ユーザモデルを利用した適応型支援の有効性を示している。

本システムにおけるユーザモデルは、システムがユーザの理解状態に柔軟に適応し、適切な支援を行うために必要な情報を提供するきわめて重要なモジュールである。ところで、ユーザモデルは実際のユーザの振舞い、具体的にはユーザの入力系列をもとにユーザの理解状態をモデル化するわけであるから、その際、次のような原因でノイズが混入することを考慮しなければならない^{7),8)}。

- (1) ユーザはつねに首尾一貫した振舞いをするわけではない。
- (2) 学習が進むにつれてユーザの理解状態は変化していく。

ユーザモデルの示す被験者の理解状態とユーザ自身が持つ実際の知識を比較した結果から、本論文で提案した動的なユーザモデルの構築法では、上記の理由から発生するノイズの混入は防止できたことが分かる。したがって、本手法では信頼性の高いユーザモデルが構築可能である。本システムには、ユーザのコマンドへの嗜好を考慮して、特定のコマンドに関する支援を停止する機能がある。これは、ユーザが自分の嗜好に応じて意図的に特定のコマンドを使わない場合に、そのユーザにとっての無意味な支援を停止する機能である。

5. む す び

本論文ではユーザのシステム使用上のスキル向上を促進する一手法として、ユーザの未知の機能をシステム側から能動的に提示するオンライン支援システムを提案した。実際に 11 名のユーザを対象に本システムによるオンライン支援を行った結果、ユーザの使用するコマンドの個数が増加したことから、本システムの有効性が確認できた。また、支援システムには各ユーザの持つ能力あるいは経験に基づいた個人適応型のイ

ンタフェースが必要であることも提案し、システム内部にユーザモデルを構築する手法を示した。そして、ユーザモデルの有無によるユーザのスキルの変化および支援頻度の違いを観察・分析し、ユーザモデルを持つ適応型のヒューマンインタフェースの必要性を確認した。さらに、ここで提案したユーザモデルの構築法は、これまでに知的 CAI の分野において提案されたモデルに対し比較的単純なものであるにもかかわらず、信頼性の高いモデルが構築できることも確認できた。

本論文で提案したダイナミックな挙動解析の手法および適応型ヒューマンインタフェースの構築の手法、動的なユーザモデルの構築法は、多くのソフトウェア・システムに適用可能である。たとえば、エディタやメールリーダー、ニュースリーダー、データベースの対話的探索システム¹³⁾などにも適用可能であろう。

今後の課題としては、同一の作業を行う際の作業量や作業時間の変化などを指標とし、ユーザのスキル向上による、そのシステムのパフォーマンスの変化を評価することがあげられる。このため、今後、約 100 名のユーザを対象に本システムの評価を行う実験を計画している。また、現システムにおける実装では不十分である、知識とプログラムの分離をより徹底する必要がある。たとえば、現在の C 言語による実装では、最も合理的なコマンドの表現は if-then 形式の条件分岐としてプログラムコード中に組み込んである。本稿で対象としたコマンドインタプリタ以外のソフトウェアシステムに本手法を適用するためには、このような部分をプログラムから明確に分離し、知識としてフレーム等で表現する必要がある。さらに、本手法がどの程度複雑なコマンド体系に対応できるかを検討することで、適用範囲の限界を検討することも必要である。

謝 辞

筆頭著者（高田）の卒業後、本稿の修正にあたり編集作業に協力してくれた、福井大学大学院工学研究科情報工学専攻の加藤友彦氏（現・永和システムマネジメント）および長谷川康雄氏に感謝します。

参 考 文 献

- 1) 角田真二、三宅芳雄：ワードプロセッサの使用における学習の停滞、第 6 回ヒューマンインタフェースシンポジウム論文集、pp.79-84 (1990).
- 2) Carroll, J.M. and Rosson, M.B: Paradox of the Active User, *Interfacing Thought: Cognitive Aspects of Human-computer interaction*, Carroll, J.M.(Ed.), MIT Press, Cambridge (1987).

- 3) Anderson, G. and Anderson, P. (著), 落水浩一郎, 大木敦雄 (訳): UNIX C SHELL フィールドガイド, パーソナルメディア (1987).
- 4) 太田英憲, 山本和彦: 高機能シェル, *UNIX MAGAZINE*, Vol.9, No.1, pp.89-101 (1994).
- 5) 高田光男, 小高知宏, 小倉久和: UNIX シェルユーザのための適応型ユーザインタフェースの構築とその評価, 第 50 回情報処理学会全国大会論文集 (4), pp.331-332 (1995).
- 6) 高田光男, 小高知宏, 小倉久和: UNIX シェルユーザの挙動解析とユーザモデル構築の手法, 平成 6 年度電気関係学会北陸支部連合大会講演論文集, p.299 (1994).
- 7) 溝口理一郎, 角所 収: 知的 CAI における学習者モデル, 情報処理, Vol.29, No.11, pp.1275-1282 (1988).
- 8) 溝口理一郎: 知的教育システム, 情報処理, Vol.36, No.2, pp.177-186 (1995).
- 9) Wenger, E. (著), 岡本敏雄, 溝口理一郎 (訳): 知的 CAI システム-知識の相互伝達への認知科学的アプローチ, オーム社 (1990).
- 10) Wasson, B. and Akselsen, S.: An Overview of Online Assistance: From On-line Documentation to Intelligent Help and Training, *The Knowledge Engineering Review*, Vol.7, pp.289-322 (1992).
- 11) Hancock, P.A. and Chignell, M.H. (著), 長町三生 (訳): 知的インタフェース-人とマシンとの知的相互作用, 海文堂 (1991).
- 12) 渡辺 茂, 坂元 昂: CAI ハンドブック, フジテクノシステム (1989).
- 13) 高橋武一, 小高知宏, 小倉久和: 情報文献検索者の意図を理解するためのユーザモデルの検討, 第 48 回情報処理学会全国大会論文集 (2), pp.157-158 (1994).

(平成 8 年 3 月 18 日受付)

(平成 9 年 7 月 1 日採録)



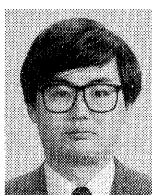
高田 光男 (正会員)

1971 年生. 1994 年 福井大学工学部情報工学科卒業. 1996 年 同大学院工学研究科博士前期課程修了. 同年アトラス情報サービス (株) 入社. 現在, ソフトウェア開発部門に所属. 主に旅客販売システムの設計, 開発業務に従事. 知的インタフェース, 認知工学の分野に興味を持つ.



西野 順二 (正会員)

1990 年東京工業大学工学部制御工学科卒業. 1992 年同大学院総合理工学研究科システム科学専攻修士課程修了. 1995 年同博士課程単位満了退学. 同年福井大学工学部情報工学科助手, 現在に至る. ファジィシステム, ファジィ制御および知的情報システムの研究に従事. 日本ファジィ学会, 日本顔学会各会員.



小高 知宏 (正会員)

1983 年早稲田大学理工学部卒業. 1990 年同大学院理工学研究科後期課程卒業. 工学博士. 同年 4 月九州大学医学部附属病院助手. 1993 年福井大学工学部情報工学科助教授, 現在に至る. 知識情報処理, 医療情報処理の研究に従事. IEEE, ACM, 電子情報通信学会, 日本 ME 学会会員.



小倉 久和 (正会員)

1969 年京都大学理工学部物理学科卒業. 同大学院博士課程修了後, 高知医学大学助手, 同助教授, 福井大学工学部助教授を経て, 現在同情報工学科教授. 理学博士. 知識の表現・獲得・処理を中心に, GA, NNW, ファジィ言語, 言語表現, 医用画像理解, 医療情報処理などの分野で研究に従事. 著書「情報の論理数学入門」(共著, 近代科学社)「形式言語と有限オートマトン入門」(コロナ社)など. 電子情報通信学会, 人工知能学会, 日本ファジィ学会, 日本医療情報学会, IEEE, ACM など各会員.