

動画編集支援における個人適応化モデルの検討

炭野 重雄

宮武 孝文

上田 博唯

sumino@crl.hitachi.co.jp miyatake@crl.hitachi.co.jp ueda@crl.hitachi.co.jp

(株)日立製作所 中央研究所

ヒューマン・インタフェース開発の一環として、個人適応をアプリケーション・ソフトにおける一連の操作系列の短縮と定義し、それを実現するための機構として個人適応化モデルを提案した。本モデルでは、重み付き有向グラフを用いることにより、状態遷移図では表現不可能な、動的な構造の変形を可能にし、グラフの頂点に機能を、辺に対象物の属性の指定および機能の選択を、辺の重みに機能の選択頻度を割り当てた。頻繁に使われる操作系列は、短縮コマンドとして既存の機能の選択肢に追加される。また、現在研究・開発中の動画編集オーサリングシステム IMPACT を例題に、本モデルの基本的な動作の確認を行ない、いくつかの課題を明らかにした。

A User Adaptation Model for Supporting Motion Picture Authoring

Shigeo Sumino Takafumi Miyatake Hirotada Ueda

Central Research Laboratory, Hitachi, Ltd
1-280, Higashi-koigakubo, Kokubunji-shi, Tokyo 185, Japan

We define a user adaptation as a reduction of a series of user's operations in an application software, and propose a user adaptation model. Using weighted directed graph, the structure of this model can be transformed dynamically. In this model, vertices are assigned to functions, edges to assignment of objects and selection of functions. Weights of edges hold selection probability of executable functions. If an certain series of operations is used frequently, a new edge corresponding that series is appended to the existing directed graph. Some results and future problems of the prototype on the motion picture authoring system IMPACT is also discussed.

1 まえがき

21世紀になると益々コンピュータが普及し、コンピュータのパーソナル化、家電化が図られるであろう。すると、今までは専門家や技術者しか扱わなかったコンピュータを誰でも簡単に扱え、使っている人の好みにそのインタフェースが変化するような、すなわち、コンピュータの方が“自動的に”ユーザの嗜好を感知し、そのユーザに適応してくれるインタフェース、“個人適応”可能なインタフェースの必要性が高まってくると思われる。

本報告では、ユーザからの変更指示を受けるのではなく、システム自身がユーザの操作意図を理解し、そのユーザに適応する枠組として、重み付き有向グラフを用いた個人適応化モデルを提案する。本提案モデルは、システムが提供するコマンド体系を用いたユーザの操作系列の中から、そのユーザにとって意味を持つ一連の操作系列を抽出・短縮することを個人適応と定義し、実現するものである。有向グラフを用いることによって、状態遷移図では不可能な構造の変形を可能とした。

その際、有向グラフの各要素には、頂点にアプリケーション・ソフトにおけるある機能を、有向辺に操作対象物の属性の指定と機能の選択を、辺の重みに機能の選択頻度を割り当てた。ユーザがある特定の操作を頻繁に行なうと、一連の辺の選択頻度があるしきい値を越え、システムは対応する有向辺の始点と終点の間を短絡する新規有向辺を生成し、操作系列を短縮するコマンドを既存の機能の選択肢に追加する。

2章において、個人適応の意義について述べ、3章において、個人適応を定義し、それを実現するための機構である、重み付き有向グラフを用いた個人適応化モデルを提案する。4章において、動画像の編集過程を述べ、動画像オーサリングシステムIMPACT^[5]に対して本提案モデルを適用した結果と個人適応化の例を示す。5章において、試作結果について検討し、課題を列挙する。

2 個人適応の意義

2.1 個人適応とカスタマイズの相違

本報告では、ユーザ自身が積極的にコンピュータの世界に身を投じて陽にそれらの設定を変更しなければ、自分の意図したようにはならず、ユーザ自身が積極的にコンピュータに働きかけて諸設定を変更することをカスタマイズと呼ぶ。

一方、仮名漢字変換ソフトにおける候補文字列の並べ

変えのような、ユーザ自身が積極的に操作を行なわなくても、コンピュータがユーザの操作意図を理解して内部構造を変更し、ユーザの使いやすい環境を創ってくれるものを個人適応と呼ぶ。

このカスタマイズと個人適応の違いを、ユーザとコンピュータとの関係という面から見直すと、ユーザがカスタマイズ可能な領域は、ユーザの要求とコンピュータによる支援が一致している部分、すなわち、ユーザが何を要求したら何をコンピュータが支援してくれるのかを理解している部分である。この領域は、人間の記憶の限界や日々進歩する技術にユーザが追従するのは困難であるという理由から、それほど増加させられないであろう。

個人適応は、この今後さほど増加できないと思われるカスタマイズ可能な領域を補い、ユーザとコンピュータのインタフェース部分を拡大することを目的としている。その際、ユーザの方からコンピュータに歩み寄るのではなく、コンピュータがユーザの方に歩み寄ってくるという点が特長である。このことから、個人適応はヒューマンインタフェース技術として今後益々重要になると考えられ、個人適応化機構はシステムにとって必要不可欠な要素になると思われる。

2.2 個人適応とは

個人適応をコンピュータのシステム設計者の立場とユーザの立場から検討する。

システム設計者は、その設計したシステムをユーザがどのように操作するか操作手順を設定してシステムを提供している。一方、ユーザはシステム設計者の意図を完全に知ることはできないので、使用しているシステムの操作手順を推定して、頭の中にそのイメージを描いて操作を行なっている。

システム設計者が提供した操作手順と、ユーザが推定した操作手順の両者の間にはしばしばギャップが存在する。個人適応とは、このシステム設計者が提供した操作手順とユーザが推定した操作手順の両者の間に存在するギャップを、ユーザの操作意図に従って解消することである(図1)。言い替えると、システム設計者が提供した操作手順を、ユーザの意図を反映しているであろう操作系列を解析して、ユーザの推定した操作手順に近付けるように変形することが個人適応化のプロセスである。

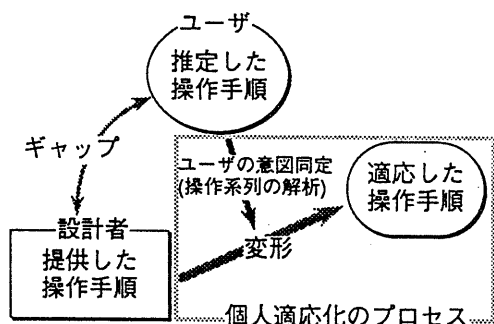


図 1: 個人適応とは

3 重み付き有向グラフを用いた個人適応化モデルの提案

3.1 個人適応の定義

本報告では、ユーザがあるアプリケーション・ソフトにおいて、対象物を指定し、その対象物に施す機能を選択・起動して、対象物を加工し、ある目的をもった処理を行なう場合を想定する。このときのあるユーザの操作系列の中からそのユーザにとって意味を持つ一連の操作系列を抽出・短縮することを個人適応と定義する。

仮名漢字変換が、“文字列”の選択頻度による並べ変えだとする、一連の操作系列の短縮を個人適応と定義することは、“操作”を選択頻度によって並べ変え、更にその前後の操作の選択頻度も吟味して、何かしらの処理をしたもの、とも見なせる。

また、Macintosh上のHyperCardで動作する実験システムEAGER^[2]は、操作履歴を1次元的に管理し、ある操作が行なわれると、それに類似したものを常に過去に遡って探索する。類似したものが存在した場合、ある操作とそれに類似する操作との間の操作系列を短縮する。すなわち、操作系列がループ(単純な操作の繰り返し)になっている場合、その操作系列の短縮を行なう。

本提案モデルでは、操作系列を2次元的にとらえ、操作系列がループになっていない場合、更には操作系列に誤った操作や遊びの操作が含まれた場合にも頑健で、より広い局面において操作系列の短縮が可能である。

3.2 重み付き有向グラフを用いた個人適応化モデル

前節で定義した個人適応を実現する機構として、本報告では重み付き有向グラフを用いた個人適応化モデルを提案する。有向グラフを用いることにより、状態遷移図では表現不可能な、動的な形状の変形を可能にした。以下に有向グラフの要素を定義する。

- (1) 頂点：アプリケーション・ソフトウェアにおいて、ある機能を起動して次の機能を起動するまでの状態、すなわち、ある機能が起動している状態と、その機能が終了して次の入力待機している状態を合わせたものを状態として頂点に割り当てる。個人適応を操作系列の短縮と定義しているため、どの機能がどの対象物に対してどのような順序で起動されたか、ということだけに注目していることに因る。現在の状態はいずれかの頂点に存在する。
- (2) 有向辺：有向辺の始点が表している機能において、次に起動可能な機能の選択操作(単純にはメニューによる機能の選択)とその機能が作用する対象物の属性の指定を割り当てる。ある対象物を指定し、その対象物に施したい機能を機能の選択肢から選択すると、現在の頂点から、その対象物の属性と機能との組み合わせに応じて、該当する有向辺に従って次の頂点に遷移する。頻繁に行なわれる操作系列の短縮は有向辺の生成に対応する。詳細は3.3節参照。
- (3) 有向辺の重み：有向辺の始点が表している機能において、有向辺の終点を表している、次に起動可能な機能の選択頻度を割り当てる。個人適応化規則適用の条件に利用する。重みの記憶方法とその更新条件などの詳細は3.4節参照。

本提案モデルは、ユーザの操作意図を同定するためのタスクモデルを具体的な概念、即ちアプリケーション・ソフトにおける状態とし、それらの依存関係を有向辺としてグラフ構造で表現したものである。

ユーザの操作は、ある究極の目的(ゴール)を達成するための処理を詳細化し、より詳細化したものからボトムアップ的に行なわれる。操作系列の短縮を順次続けて行なうことは、そのユーザがたどるボトムアップ処理を支援するものであり、ゴールを達成するためのそのユーザだけのコマンドを作成することとなる。

文献[3]においても、タスクモデルとして状態遷移図、すなわち、グラフ構造を用いている。それによって、ユーザの操作をシステムのイベント単位でトレースし、状況に応じたオンラインヘルプの提示を可能としている。しかし、静的な状態遷移図を用いるため、ユーザに応じた動的なヘルプの提示を行なうことは不可能である。

一方、タスクモデルを木構造で定義する方法もある[1]。構造を木で表現すると、抽象的な機能の上下関係、依存関係などは容易に記述できるが、操作履歴のような機能の順序関係を記述する場合、木構造の節を上下に探索しなければならず、感覚的に不自然で煩雑な操作が必要となる。グラフ構造を用いると、機能の順序関係は有向辺によって極めて自然に記述でき、また、同一の操作系列の繰り返しなどは有向グラフの強連結成分の検出などによって容易に行なうことができる。

3.3 個人適応化規則

ここで、 n 個の頂点の集合を $V = \{v_i\}_{i=1, \dots, n}$ 、 v_i を始点、 v_j を終点とする有向辺を $e_{ij} = [v_i, v_j]$ 、 v_i を終点とする IN_i 本の有向辺の集合を $E_i^+ = \{e_{ji}\}_{j=i_1^+, \dots, i_{IN_i}^+}$ 、 v_i を始点とする OUT_i 本の有向辺の集合を $E_i^- = \{e_{ik}\}_{k=i_1^-, \dots, i_{OUT_i}^-}$ 、全ての有向辺の集合を $E = \{e_{ij}\}_{i,j}$ 、有向辺の総数を $m = |E| = \sum_{i=1}^n |E_i^-| = \sum_{i=1}^n |OUT_i|$ とする。但し、 i_j^+ は頂点 v_i を終点とした有向辺のうち、 s 番目の有向辺の始点の添え字を、 i_t^- は頂点 v_i を始点とした有向辺のうち、 t 番目の有向辺の終点の添え字を表す。また、 e_{ij} に割り当てられた機能、すなわち、 v_i から v_j に遷移する際に起動する機能と対象物の属性をそれぞれ f_j 、 $attr_{ij}$ とする。図2に個人適応化モデルの概念図を示す。

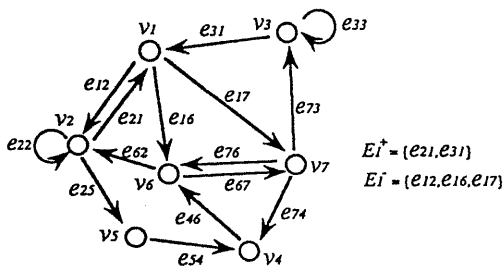


図2: 個人適応化モデルの概念図

以上の記述に従うと、一連の操作系列の短縮と定義し

た個人適応は、有向辺と、その有向辺に割り当てる機能と対象物の属性との問題に帰着する。以下に個人適応を図るための個人適応化規則を示す。2.2節に従うと、システム設計者が提供した操作手順をユーザの操作系列を解析して変形するための規則が個人適応化規則である。

- (1) 有向辺の生成：頂点 v_i 、 v_j 、 v_k に対して、有向辺 e_{ji} と e_{ik} との接続が強いと判断した場合、新たな有向辺 $e_{NEW} = e_{jk} = [v_j, v_k]$ を生成し(図3)、状態が頂点 v_j から $v_k \in e_{NEW}$ に従って遷移するための新規選択肢(メニュー)を既存の機能の選択肢に追加する。その際、 $attr_{ji}$ と $attr_{ik}$ が同一で、 f_i と f_k が異なる場合、 e_{NEW} に f_i が起動した後、 f_k が起動する機能を割り当てる。また、 $attr_{ji}$ と $attr_{ik}$ 、 f_i と f_k が共に等しい場合、 f_i を連続的に起動し、その起動毎に属性 $attr_{ji}$ の対象物を自動的に指定する機能を e_{NEW} に割り当てる。
- (2) 頂点およびそれに接続する有向辺の削除：ある条件判定(例えば、メッセージに対するユーザのインタラクション)により、 v_i および有向辺の集合 E_i^+ 、 E_i^- を削除する。

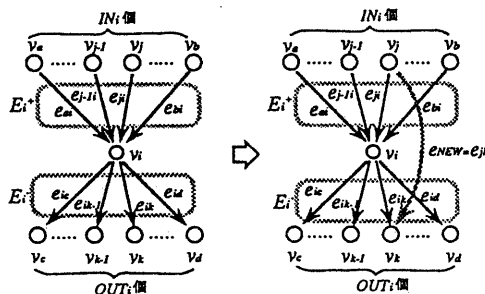


図3: 有向辺 e_{NEW} の生成
(但し、 $a = i_1^+$, $b = i_{IN_i}^+$, $c = i_1^-$, $d = i_{OUT_i}^-$)

3.4 有向辺の重みの記憶方法とその更新条件について

頂点 v_i に対して、有向辺の集合 E_i^- の重み、すなわち、次に起動可能な機能の選択頻度の記憶方法とし、次の2つの方法が考えられる。

- (a) E_i^- の内から e_{ik} を選択する確率を有向辺 e_{ik} の重み w_{ik} とする方法。 e_{ik} に対する重みの数は 1 個。
 $(k = i_1^-, \dots, i_{OUT_i}^-)$
- (b) e_{ji} を選択したという条件で、次に e_{ik} を選択する確率、すなわち、 e_{ji} に対応する操作を行なった後に e_{ik} に対応する操作を行なう確率を有向辺 e_{ik} の重み w_{jik} とする方法。 e_{ik} に対する重みの数は IN_i 個 (図 4)。($j = i_1^+, \dots, i_{IN_i}^+$; $k = i_1^-, \dots, i_{OUT_i}^-$)

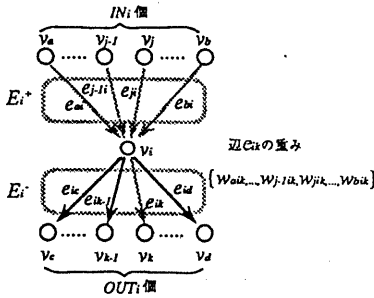


図 4: 有向辺 e_{ik} の重み w_{jik}
(但し、 $a = i_1^+$, $b = i_{IN_i}^+$, $c = i_1^-$, $d = i_{OUT_i}^-$)

(a) の E_i^- の内から e_{ik} を選択する確率を用いて有向辺の接続の強さを判定する方法は、ある機能を起動しているときは傾向として、次にどの機能が選択される確率が高いか、ということしか判断できない。その反面、重みの記憶領域の総数は $|\{w_{ik}\}| = O(n^2) = O(m)$ 個で、高々有向辺の総数でよい。

操作系列の短縮という意味では、(b) の e_{ji} に対応する操作を行なった後に e_{ik} に対応する操作を行なう確率を用いて、有向辺の接続の強さを判定する方法の方が理にかなっていると思われる。しかし、重みのための記憶領域を考察すると、その総数は $|\{w_{jik}\}| = O(n^3) = O(m^{1.5})$ 個となり、頂点の総数および有向辺の総数が増加すると、その重みを記憶する領域が急激に増加する。

また、上記 (b) は e_{ji} に対応する操作を行なうという条件における、 e_{ik} に対応する操作を行なう確率であるので、条件付き確率と見なせる。従ってその応用として、ある条件における条件付き確率、更にそれに条件を付加したものなどが導出可能であり、情報理論における多重マルコフ過程の概念が導入できる。その際、有向辺の重みの記憶領域の総数は、条件が付加される毎に $O(n)$ の掛け算の割合で増加する。例えば一つ前の操作を条件と

して加えて (b) の方法で重みを記憶した場合、その重みの総数は $O(n^3) \times O(n) = O(n^4) = O(m^2)$ となる。

本報告では、有向辺の重みの記憶方法として、ユーザの操作の依存性をより詳細に抽出可能な、多重マルコフ過程の一番単純な場合と見なせる (b) の方法を採用した。

有向辺の重みの更新は、任意の操作系列に対して行なうのではなく、ユーザの操作意図が含まれると思われる、次の 2 つの操作系列が行なわれた場合に限り行なう。これらの条件を付加することによって、ユーザの意図をより反映した操作系列の抽出を可能とした。

同一対象物に対する複数機能の順次起動時：一つの対象物を指定して、それに対して複数の機能を順次作用させて作業を行なう場合。

同一属性の連続的な対象物に対する同一機能の連続起動時：属性が同一で意味的に連続的な対象物が存在したとき、それらに対して同一の機能を連続的に作用させる場合。

有向辺の接続の強さの判定は、辺の重み w_{jik} 、すなわち、 v_j から v_i 、 v_k と機能 f_i 、 f_k を順に起動して遷移する確率と、あるしきい値 Th との大小比較によって行なう。 $w_{jik} \geq Th$ の場合、すなわち、 f_i を起動した後に f_k を起動する確率が高い場合、メッセージをユーザに提示して判断をおおぎ、その判断によって個人適応化規則を適用する。その際の新たな有向辺 $e_{NEW} = e_{jk}$ の生成に伴い、有向辺の重み $\{w_{hjk}\}_{h=j_1^+, \dots, j_{IN_j}^+}$ とその重みに対応して順次起動する機能 f_i, f_k を既存のデータ構造に追加する。直感的には、有向辺 1 本の追加に対して、その始点に接続している有向辺の数だけ重みの数を追加する。

4 本提案モデルの IMPACT への適用

4.1 動画像の編集過程

以下に、一般的な動画像の編集過程を示す。

1. プレビュー：まず始めに、撮影もしくは獲得した生の動画像の内容を把握するため、動画像をプレビューする必要がある。
2. カット分割：編集を施していない生の動画像は、まず、カットと呼ばれる、ある意味をもった動画像の塊に分割して処理を行なうのが一般的である。

3. カットの取捨選択 (属性付加) : カットには unnecessary も存在するので、そのカットを取捨選択、もしくは、名前、撮影日時、検索のためのキーワード、OK/NG などの属性情報を付加して、そのカットが必要か、不必要かを明確にする作業が必要である。
4. カットのトリミング : カットの不要部分を切り詰めて、再生時間を調整するためのカットの開始点 (イン点)、終了点 (アウト点) をフレーム単位で変更する操作である。ここで、フレームとは動画を構成している 1 秒間に 30 枚表示している各静止画像を示す。
5. カットの代表画像変更 : カットを縮小画像を用いてアイコン表示する際、そのカットを象徴的に示し、他のカットと区別がつくようなフレームを表示した方が一見して内容が把握できるため、後の操作も容易になる。代表画像変更操作は、デフォルトであるイン点の画像を適切な画像に変更する操作である。
6. カットの順序並べ替え : トリミングなどの操作で無駄な部分を排除したカットを、ストーリーに従って並べ変える操作である。
7. カット間の特殊効果 : 2 つのカット間の移行を効果的に行なうための処理である。この処理を有効に用いると、ストーリー展開をスムーズに行なえる。

但し、これらの操作は 1. から 7. まで順に行なわれるのではなく、編集を施しているユーザの嗜好に応じて任意の順で行なわれる。例えば、編集を施していない生の動画像に対して、カット分割、トリミングなどを終了してから、次の操作に移行するユーザもいれば、カット分割を行なったものに対して、その都度すぐにトリミング操作を行なうというユーザも考えられる。また、カットの並べ替えを行なってから、ストーリー全体として時間が長い場合や、リズム感が悪い場合などは、再度それぞれのカットに対してトリミング操作を行なう必要も生じるであろう。

4.2 IMPACT における機能と個人適応化モデル

IMPACT は、テープの早送り・巻き戻しなどの煩わしい操作を行なうのではなく、マウスによるダイレクトマニピュレーションでカードを並べ変えるように、誰もが容易に動画の編集・加工が行なえることを目的と

したシステムである。

以下に、4.1 節で述べた動画の編集過程を行なうために IMPACT が有している機能を示す。

- (a) プレビュー機能、(b) カット分割機能、(c) カットトリミング機能、(d) カット代表画像変更機能、(e) カット属性付加機能、(f) カット順序並べ替え機能、(g) カット連結機能、(h) カット特殊効果機能、(i) カット構造化機能、(j) カット合成機能、(k) カットコピー機能、(l) カット削除機能

本報告では、上記した 12 の機能のうち、(a)、(b)、(c)、(d)、(e)、(f) の 6 つ機能に限定して考察した。図 5 に、4.1 節の編集過程に従って個人適応化モデルを適用した例を示す。

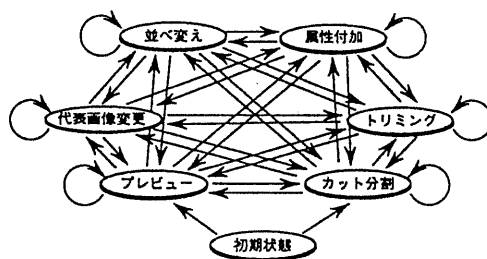


図 5: IMPACT における個人適応化モデルの適用例

4.3 IMPACT における個人適応化例

前節で示した IMPACT における個人適応化モデルの適用例において、個人適応化の例を示す。

(1) 同一機能の連続起動

動画 α を編集する手順として、ユーザ A は、(1) まず、 α をプレビューし、(2) その α を $\alpha_1, \alpha_2, \dots, \alpha_N$ の N 個にカット分割、(3) その分割されたカット $\alpha_1, \alpha_2, \dots, \alpha_N$ を、一つずつ順番にトリミングする、という方法を取ったとする。

トリミング機能は、対象物の指定であるトリミングを施したいカットの指定と、機能選択であるトリミングアイコンの押下によって起動するので、ユーザ A はトリミングを施す都度にカット α_i の指定とトリミングアイコンを押下しなければならない ($i = 1, \dots, N$)。この局面に個人適応化規則が適用されると、トリミング機能が割り当てられた頂点を始点・終点とする、トリミング機能を連続起動するための有向辺が生成される (図 6)。これ

に対応して、表示画面上にはトリミング機能を連続起動するためのアイコン、連続トリミングアイコンが追加される。その結果、その都度行なっていた対象物であるカットの指定の手間が省かれ、トリミングを施したい対象物であるカットを指定して、その連続トリミングアイコンを押下することによって、一つのカットに対するトリミング操作が終了すると、自動的にまだ操作を施していないカットが選択され、再度トリミング機能が起動するようになる。ここで、動画像 α の指定とトリミングアイコンを押下を $\alpha+$ (トリミング) と表記した。

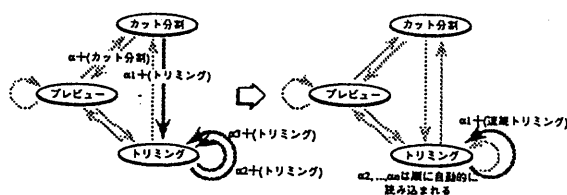


図 6: 同一機能の連続起動

(2) 同一対象物に対する複数機能の順次起動

また別のユーザ B の編集の手順は、(1) あるカット a をプレビューし、(2) a を a' と a^* の 2 つにカット分割、(3) 分割した一方のカット a' をトリミング、(4) 別の操作 (例えば、属性付加) をして、再度カット b をプレビューしてから b' と b^* の 2 つにカット分割、 b' をトリミング、カット c をプレビューしてから c' と c^* の 2 つにカット分割、 c' をトリミングという方法だったとする。この局面に個人適応化規則が適用されると、プレビュー機能が割り当てられた頂点を始点、トリミング機能が割り当てられた頂点を終点とする、カット分割機能とトリミング機能を順次起動するための有向辺が生成され (図 7)、その有向辺に対応するカット分割・トリミングアイコンが追加される。その結果、対象物であるカットを指定してその追加されたアイコンを押下すると、まずカット分割機能が起動し、終了すると同時に自動的にカット分割機能で分割されたカットが対象として指定され、トリミング機能が起動するようになる。

5 結果の検討と今後の課題

IMPACT 上に、上記した個人適応化モデルを用いた個人適応化機構を実現し、編集時のユーザの操作系列に

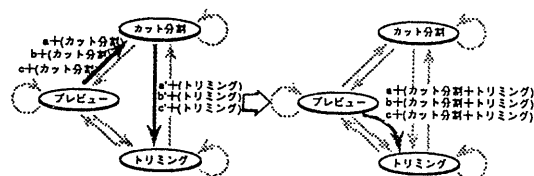


図 7: 同一対象物に対する複数機能の順次起動

応じた、

- 同一機能の連続起動による対象物の指定の省略
- 同一対象物に対する複数機能の順次起動による機能のマクロ化

によって、操作系列が短縮されたことを実証した。その際、以下のような問題点を確認した。

- 意図しない時点での個人適応化規則の適用：有向辺の重みの値が初期状態に近い場合、すなわち、ほとんどの値が 0 の場合、ある特定の操作系列が繰り返し行なわれると、その特定の操作系列に対応する辺の重みだけが更新され、容易にしきい値を越えて個人適応化規則が適用されてしまう。
- 意図しない操作系列の短縮：特定の複数の機能を同一の順序で繰り返し起動する場合、短縮される操作系列の順序によっては、ユーザの意図しないものが短縮される場合がある。例えば、トリミング機能、代表画像変更機能の 2 つのみを順に繰り返し起動した場合、ユーザが短縮を望んだ操作系列がトリミング、代表画像変更の順序だったにもかかわらず、システムは代表画像変更、トリミングの順で操作系列を短縮する場合がある。

(a) の問題に対しては、選択頻度を計算する際の分母の値がある程度大きくなるまで、すなわち、ユーザがある程度システムを使用して有向辺の重みが更新されてから、重みとしきい値とを比較して個人適応化規則を適用するか否かを判定することで対処した。しかし、個人適応化規則を適用する条件を、選択頻度を用いるのか、もしくは、絶対的に、特定の操作系列を使用した回数を用いるのかという問題が残った。この問題に対しては、仮名漢字変換における候補文字列の並び替え (再配置) の判定アルゴリズムが参考になるとと思われる。

また、(b)のユーザが短縮したいと思っている操作系列とシステムが抽出した操作系列とのギャップを埋めるためには、システムが短縮する操作系列の候補を列挙し、ユーザと対話的に短縮する操作系列を決定するという方法が考えられる。

更に個人適応機能の高度化を図るには、以下のような課題がある。

- (A) 複数対象物の指定：現在、対象物の指定は単数しか許していないが、複数の対象物を指定し、それらに対して全て同一の機能を施すという局面は容易に考えられる。よって、対象物の指定を複数にした場合でも対応可能なモデルの開発。
- (B) デフォルトで指定される対象物の特定：個人適応化規則の適用によって、ある操作系列が短縮され、複数の機能が順次起動するような局面で、その操作系列に含まれるある機能を起動したことによって対象物が複数に分割された場合、次に起動する機能の対象物として、複数に分割されたどの対象物を指定するか。
- (C) 個人適応化規則の洗練：現在の個人適応化規則による頂点間の接続の再構成において記述不足な部分、すなわち、頂点およびそれに接続する有向辺を動的に発生・消滅する規則の開発、および、既存のグラフ理論の有効利用。
- (D) 頂点に割り当てる状態の粒度：現在は有向グラフの頂点に、アプリケーション・ソフトにおける機能という比較的大きな粒度の状態を割り当てているが、それが適切なのか、もしくは、適切な粒度が存在するのか、存在するならばその指針とは何か。

6 むすび

より人間にやさしいインタフェースを提供することを目的として、ユーザの操作意図を理解し、システムが個人に適應する枠組として、重み付き有向グラフを用いた個人適応化モデルを提案した。本提案モデルでは、ユーザの操作意図が含まれると思われる、同一対象物に対する複数機能の順次起動と、連続的な対象物に対する同一機能の連続起動が頻繁に行なわれると、対応する有向辺の重みを更新する。その値があるしきい値を越えた場合、一連の有向辺を短縮する新規有向辺を生成し、それに伴って操作を短縮するコマンドを既存の機能の選択肢に追加する。

また、本提案モデルを動画像オーサリングシステム IMPACT 上に実現することによって、ユーザの操作に応じた操作系列が短縮されることを確認し、本提案モデルを更に発展させるための課題を明らかにした。

今後は、通産省の FRIEND21 プロジェクトで提唱している、分散協調可能なヒューマンインタフェース・アーキテクチャであるエージェンシーモデル^[4]上で本提案の個人適応化モデルを構築し、より使い勝手を向上させたヒューマンインタフェースの実現を目指していく予定である。

謝辞

日頃から熱心に討論、コメントなど下さるソニーの渡辺氏、三菱電機の朝日氏、富士ゼロックスの広瀬氏、富士通研究所の宇山氏を始めとする FRIEND21 研究センタシステムグループ主催エージェンシーモデル研究会、メタウエア研究会、および、タスク研究会のメンバーに感謝致します。

なお、本報告の一部は通産省の FRIEND21 プロジェクトの一環として実施されたものである。

参考文献

- [1] 朝日宣雄：“メタウエアにおける知識記述”，人工知能学会研究会資料，SIG-HICG-8903-3，pp.21-29 (1988)。
- [2] A. Cypher：“EAGER：Programming Repetitive Tasks by Example”，Proc. of CHI'91，Conf. on Human Factors in Computing Systems，ACM，pp.33-39 (1991)。
- [3] 日吉まゆみ：“操作ガイダンスシステムにおける個人適応”，第43回情報処理学会全国大会，1C-3，(1991)。
- [4] H. Ueda：“Agency Model - Architecture Model for Metaware”，Proc. FRIEND21 International Symposium on Next Generation Human Interface Technologies，(1989)。
- [5] H. Ueda, T. Miyatake and S. Yoshizawa：“IMPACT：An Interactive Natural-Motion-Picture Dedicated Multimedia Authoring System”，Proc. of CHI'91，Conf. on Human Factors in Computing Systems，ACM，pp.343-350 (1991)。