

2K-8

カーソル移動の認知モデルと 移動速度の最適化

安達政仁, 小山裕徳
東京電機大学工学部

1. はじめに

現在普及しているスクリーン・エディタにおいて、文字の入力、位置指定などを行なうデバイスは、主にキーボードとなっている。また、エディタのインターフェースを考慮する上で、最も基本的、かつ重要な要素は「カーソル移動」である。我々は、キーボードによるカーソル移動環境を改善することを目的とし、そのアプローチとして人間の認知行動を解析してきた。

移動時間に関する認知モデルは移動方向(行/列)別に示し、それぞれに対する実際の行動とを比較、検討した。さらに、その結果をエディタに応用することによりインターフェースが向上することも明らかにした^(1,3)。

本論では、先^(2,3)に示した列方向の認知モデルに関してさらに検討を加え、カーソルの連続移動時における問題の解決方法、および、その評価結果を示す。

2. 列方向移動時間の認知モデル

列方向の移動における最も基本的なMethod(方法, 機能, コマンドなど)は、文字単位の移動である。以下、これをC Methodと呼ぶ。

一般に、現在のカーソル列とGoal(目的)列との差 ΔC [列]が長い場合は、列方向移動キー(カーソル・キー)を押し続けることにより連続移動することが多い。この場合に要する時間 $t(\Delta C)$ [s]は、式(1)のようにモデル化することができる。

$$t(\Delta C) = t_m + t_k + T_w + T_I \cdot (\Delta C - 2) \quad 2 \leq \Delta C \quad (1)$$

ここで、 t_m はメンタル、すなわち人間がMethod選択の必要性を認知、判断するのにかかる時間、 t_k は1つのキーを打つ時間である。

T_w, T_I は、それぞれ、連続入力に移行するまでの待ち時間、連続入力間隔であり、カーソル移動速度を決定するパラメータである。

3. ポインティング・エラー

カーソルをGoal位置で正確に停止できないために生ずる、モデルによる予測と実際の行動との差をポインティング・エラーと呼ぶ。列方向の連続移動時においては、Goalの手前でキーを放してしまう場合、および、Goalを通り越してしまってからキーを放す場合を、ポインティ

ング・エラーとみなし、式(2)に示すPE[列]で評価する。

$$PE = \text{キーを放した位置} - \text{Goal位置} \quad (2)$$

4. カーソル移動速度の最適化

PEが発生すると、さらにそれを修正するための余計なMethodが必要となり、結果的に作業時間効率を悪化させることになる。PEを減少させるにはカーソル移動速度を個人、編集内容などに対して最適にしなければならない。

PEは移動速度に依存し、移動距離が短いほど速度を遅くすることによるPEの減少効果は著しくなる。しかし距離が長い場合、移動時間を短縮するために速度はより速い方がよい。この矛盾は、実際の編集時における ΔC , PEを常に監視し、速度をリアルタイムに適応させることにより解決することができる。

速度の適応アルゴリズムは『連続移動の終了時に、 ΔC が一定値よりも短く、かつPEが発生した場合は遅くする。 ΔC が一定値よりも長い場合はPEにかかわらず速くする』とする。式(3)は、これを付加したモデルである。

$$t(\Delta C) = t_m + t_k + T_w + T_I \cdot (\Delta C - 2) \quad 2 \leq \Delta C$$

$$T_{Ii} = T_{Ii-1} + c \quad (\text{初期値 } T_{I0}, i=1, 2, 3, \dots)$$

$$c = \begin{cases} dT_I & \Delta C < \Delta C_T \text{ かつ } |PE| > 0 \\ -dT_I & \Delta C \geq \Delta C_T \\ 0 & \text{上の条件以外} \end{cases} \quad (3)$$

ここで、 ΔC_T は ΔC の長短を決定するしきい値、 dT_I は T_I の変化ステップであり、それぞれ適応の条件、速度を決定するためのパラメータである。

5. モデルの評価

式(1), (3)における T_I に対しての評価^(2,3)に加えて、ここではさらに T_w に対しても評価する。

5.1 主観評価

主観的に最適な T_w, T_I の組合せを求めるために、アンケートによる調査を行なった。対象はエディタの熟練者(21~23才, 男女計25人)である。結果を図1に示す。これより最適な組合せは、ほぼ次式により得ることができる。

$$T_w = 10 \cdot (50 - T_I) [\text{ms}] \quad 10 \leq T_I [\text{ms}] \leq 40 \quad (4)$$

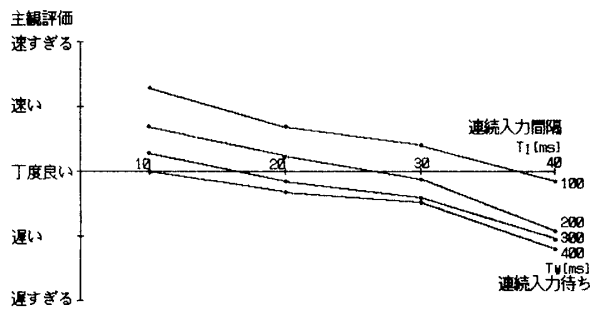


図1 連続入力間隔 T_1 の変化に対する主観評価

5.2 客観評価

T_w , T_1 を主観的に最適な組合せにすることにより、実際の編集時におけるPEは最低となるのか、また、カーソル移動速度を遅くするにしたいが、PEは確実に減少していくのか、などを検証するために以下の実験を行なった。

5.2.1 実験方法

スクリーン左端にカーソルを置き、それより右側に \mathcal{C} [列]離れて表示されている‘a’をGoal位置としてカーソルを移動する。記録はMethod選択キー \square を放した位置、すなわち最初のカーソル停止位置で \square を押すことにより行なう。

\mathcal{C} は5,10,20,40[列]とし、その出現順序はランダム、回数はそれぞれ100である。 T_1 は10,20,30,40[ms], T_w は100,200,300,400[ms]とした。

5.2.2 結果

PEの頻度分布の平均は、一様にほぼ0となる。そこで T_w , T_1 のそれぞれの組合せについて、PEの標準偏差を求めて比較した結果、図2のようになった。これより、 T_1 を長くするにしたいが、標準偏差が減少することは明らかであり、PEは T_1 に依存するといえる。しかしながら、 T_w が200[ms]以上では標準偏差の変化はほぼ同様であることから、PEは T_w に依存するとはいえない。移動時間の短縮化を考慮すると、 T_w の客観的な最適値は200[ms]であり、固定値としてよい。

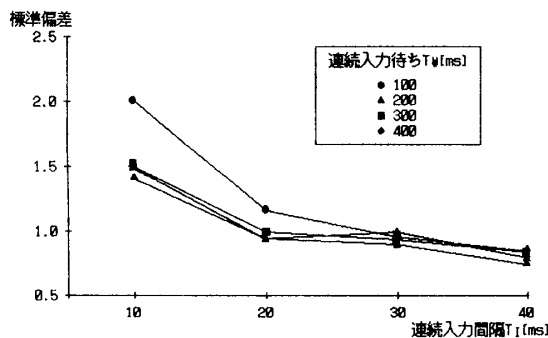


図2 ポインティング・エラーPEの標準偏差

6. 適応アルゴリズムの評価

4. に示したカーソル移動速度の適応アルゴリズムによる効果を調べるために、実験を行なった。エディタに T_1 の変化、Methodの選択過程をファイルに記録する機能を組み込み、それらをもとに評価する。

T_1 の初期値 T_{10} は20[ms], 変化範囲は10~40[ms], 変

化ステップ dT_1 は4[ms], \mathcal{C}_T は20[列]とした。また、 T_w は客観評価結果から200[ms]とした。

図3は、 \mathcal{C} の分布を周期的とした場合の結果である。図より、 T_1 が \mathcal{C} の変化に適応していく様子がよくわかり、アルゴリズムは適当であるといえる。

図4は、客観評価実験と同様に \mathcal{C} の出現順序をランダムとした場合の結果である。適応により作業時間効率が向上することを確認するため、PEの修正を含む単位作業時間(最初にC Methodを選択してからGoalに到達するまで)を求めている。適応を行う場合は、 T_1 が最も短い10[ms]固定の場合と比較し、作業時間はほぼ等しく、しかもPEの標準偏差はより小さい。これは、同じ作業時間でもPEの修正にかかる時間が少ないことを示しており、作業時間効率は向上しているといえる。

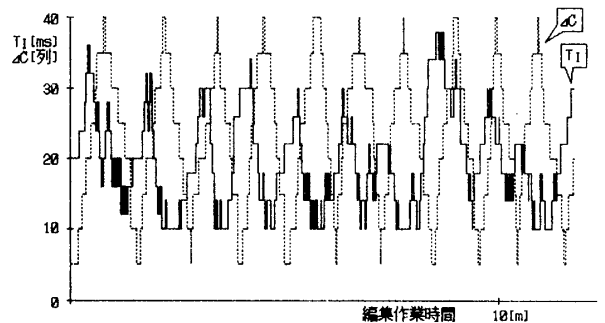


図3 移動距離 \mathcal{C} に対する連続入力間隔 T_1 の適応過程

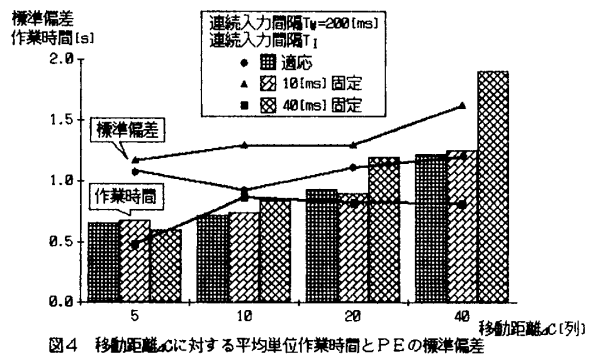


図4 移動距離 \mathcal{C} に対する平均単位作業時間とPEの標準偏差

7. おわりに

カーソル移動速度の最適化を実際の編集時に行なう手法、および、その有効性を証明した。今後はモデル自体のより精密化を検討する予定である。

謝辞 本研究を進めるにあたり、積極的に協力してくれた丸山恵美子さん、ならびに、被験者各位に感謝する。

【参考文献】

- (1)安達, 小山: カーソル移動の認知モデルとその応用, 日本人間工学会昭和62年度関東支部大会報文集, pp.41-46, 1987
- (2)安達, 小山: カーソル移動の認知モデルとポインティング・エラー, 情報処理学会第36回全国大会講演論文集, pp.919-920, 1988
- (3)安達, 小山: ポインティング・エラー情報をもとにしたカーソル移動環境の最適化手法, 日本人間工学会第29回大会講演集, pp.130-131, 1988