

高精細スライダ “FineSlider”

柏木宏一, ボーデン・ジョージ, 増井俊之
(シャープ(株)技術本部 ソフトウェア研究所)

FineSlider - A Precisely Controllable Input Device

Kouichi KASHIWAGI, George BORDEN and Toshiyuki MASUI

Software Research Lab. SHARP Corporation

2613-1 Ichinomoto-cho, Tenri-shi, Nara-ken 632, Japan

Abstract : FineSlider is an input device in Graphical User Interface(GUI), that is rapidly able to make a fine adjustment of parameter value or select a word from word-list in database. FineSlider have a scale corresponded to range of parameter and a knob moved on the scale, like general slider. However it differs from the general slider in a point that a portion of the scale, which the knob is not existing, is a fine adjustment area. Working on the area by pointing device as mouse, pen and so on, users can set a parameter value precisely and rapidly. This paper introduce a concept of FineSlider and consider the implementation of fine adjustment experimentally.

Keywords : GUI, slider, parameter setting, fine adjustment

1 はじめに

GUI(Graphical User Interface)におけるパラメータ値の選択、範囲の設定といった入力操作には通常、スライダ、あるいはスクロールバーが使用される。これらの入力ツールは主に、パラメータのレンジ分の長さを持つスケールと、その上を移動するノブから構成されており、スケール上のノブの位置がパラメータ値に対応している。ユーザはマウス、ペンなどの入力デバイスを用いてノブを直接移動することにより、あるいはノブ以外の箇所をクリックすることにより、パラメータ値を意図した値に設定することができる。また、ノブやスケールの両端等に矢印が施され、ユーザがこの矢印を選択し続けることによって矢印の方向にノブが移動するものもある。

しかし、従来のスライダ、スクロールバーでは、微細なパラメータ値の設定を行う場合、ノブを細かく動かさなければならず、ユーザの熟練を要する作業となっていた。また、ディスプレイ上に表示されたスケールのピクセル数がパラメータの数より少ない場合、細かなパラメータ設定は、単一のスライダのみでは不可能である。これらの微調整に関する問題は、ノブやスケールの両端等に矢印を施したスライダを用いることである程度は解消されるが、矢印に割り当てられた微調整のための単位は予め設定されているものであり、効率的な微調整を行うことは困難である。

一方、携帯情報端末のようにキーボードレスで、ディスプレ

イ領域の狭いツールにおけるパラメータ値の設定としては、スライダ、スクロールバーの他に、文字認識技術を応用する方法や、ディスプレイ上に実現された超小型簡易キーボードを使用する方法等が用いられる。しかし、いずれも操作性に課題を残しており、限られたスペースで操作性に優れたパラメータ値入力ツールの実現が待たれている。

そこで本稿では、携帯情報端末のような限られたスペースで、広範囲のレンジを持つパラメータの値設定、並びにワード検索を容易に行うことができるスライダとして FineSlider を提案する。

2 スライダに関する工夫と研究

上述したように従来のスライダやスクロールバーを用いたパラメータ値の設定には、微調整に関して問題が残されている。この問題を解決するために、以下のような工夫がなされている。

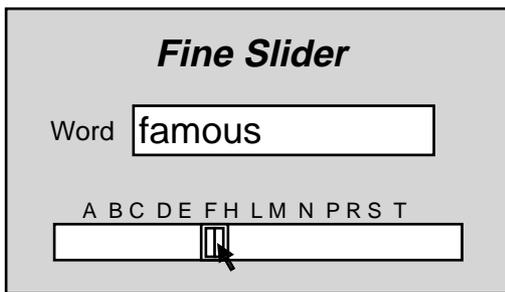
1. 複数のスライダを準備し、それぞれに異なる設定単位を割り当てる方法
2. ひとつのスライダと設定単位を変換するボタンを準備する方法
3. スライダを動かす速度を検知し、それに応じて微調整の単位を変化する方法

1 はよく用いられる方法であるが、幾つかのスライダを交互に操作しなければならないケースが見受けられ、さらに、空間的な制約がある場合には実現が困難となる。また、2 は1 のように空間的な制約はほとんどないものの、1 と同様に幾つかの設定単位を行き来する場合、操作性に問題が残る。さらに、3 は細かな値を設定する場合、非常にゆっくりとノブを移動させなければならず、やはり熟練を要す作業となる。

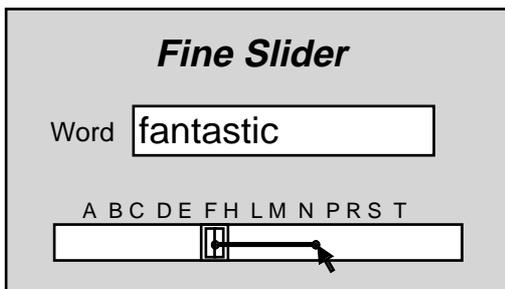
一方で、単一のスライダで細かいパラメータ値の設定を可能とするスライダとして、AlphaSlider^{[1],[2]} が提案されている。

文献 [1] の AlphaSlider は、ノブの両端に微調整用のボタンを取り付け、このボタンをクリックすることで微調整を行うスライダであり、電話番号の検索に適用されている。しかし、この AlphaSlider はスケールのピクセル数がパラメータの絶対数より少ない場合には適用されていない。それに対して、文献 [2] の AlphaSlider はスケールのピクセル数よりはるかに多い 10,000 件の映画タイトルの検索に適用されている。この研究では、パラメータ値の調整単位として Coarse、Medium、Fine の 3 つを準備しておき、これらを用いてパラメータ値の微調整を行なうことができる種類のスライダを提案している。しかし、ユーザは予め設定されている 3 つの調整単位を使用しなければならないため、検索タスクの最中に意図した設定単位を用いることができない。従って、いずれの形式においても、最終的な微調整は調整単位 Fineのみで行わなければならないため、効率的な検索を行うことができないと考えられる。

それに対して、本稿で提案する FineSlider は、広範囲のレンジを持つパラメータの値設定において、ポインティング・デバイスによるポインティング、あるいはドラッグという簡単な操作によって調整単位をリアルタイムに調整でき、パラメータ値を容易に、しかも効率よく設定することができる。



(a) ノブ移動



(b) 微調整

図 1: FineSlider の実施例

3 高精細スライダ “FineSlider”

3.1 概要

FineSlider は、例えば携帯端末のディスプレイのような限られたスペースで、非常に広範囲のレンジを持つパラメータの値設定を容易に行うことが可能である。その実施例を図 1 に示す。図 1 は FineSlider をワード検索に適用した例である。

FineSlider は通常のスライダと同様にスケールとノブから構成されているが、通常のスライダと異なる部分は、スケール上のノブのない部分が微調整領域となっているところにある。この微調整領域における任意の点とノブとの距離が微調整の単位に相当している。すなわち、ユーザがポインティング・デバイスによって微調整領域をポインティング、あるいはドラッグすることで、容易に微調整単位を変更することができ、現在のパラメータ値に対して微調整単位を加算していくことでパラメータ値を更新する。

図 1(b) は微調整を行っているところであり、ノブをゴムで徐々に引っ張っているようにインプリメントしている。

3.2 FineSlider の動作

FineSlider の動作は、ポインティング・デバイスによって指定された位置により、ノブ移動と微調整の 2 フェーズから成る。

・ ノブ移動

FineSlider におけるノブ移動は、通常のスライダと同様に、ノブをポインティング・デバイスで選択し、スケール上をドラッグすることにより行われる。このときノブ位置に対応したパラメータ値はリアルタイムにユーザに提示される。

ただし、FineSlider は、スケール上のノブの存在しない部分が微調整領域になっているため、通常のスライダのように任意のスケール上をクリックすることによってノブを移動させることはできない。

・ 微調整

微調整は、微調整領域をポインティング・デバイスで位置指定(ドラッグを含む)することにより行われる。上述したように、微調整単位はノブ位置と指定位置との距離によって変化し、一般的にこの距離の単調増加関数として表現される。すなわち、指定位置がノブに近いほど微細な単位となり、遠くなるほど大きな単位となる。本研究ではこの関数のことを微調整関数と呼ぶ。そして、現在のパラメータ値に対して、指定位置に対応した微調整単位を加算していくことでパラメータ値を更新し、リアルタイムにユーザに提示する。

また、スケールの両端とスケール上に配置されるパラメータの最大値あるいは最小値との間に余裕を持たせ、さらに、微調整領域からスケールの両端外へドラッグしても、所定の微調整が行えるようにした。これにより、ノブがスケールの両端近辺にある場合でも所望の微調整が可能となる。

多くの場合、ユーザは、まずノブを所望のパラメータ値近辺までノブを移動し、その後、微調整領域の任意の点をクリック、あるいはドラッグすることで微調整単位を随時変化させてパラメータ値の微調整を行う。このような操作によって、ユーザは、広範囲のレンジをもつパラメータの値設定、あるいはワード検索を効率的に行うことができる。



図 2: 実験で用いた FineSlider

4 実装実験

4.1 実験の目的と環境

・目的

本実験は、FineSlider における微調整関数について考察することを目的としている。従って、実験内容としては、単一のタスクに対して、幾つかの微調整関数を準備しておき、タスクの達成時間を評価指標にして、これらの関数の比較を行う。実験で行うタスクは文献 [2] と同様に、10,000 作の映画タイトルの検索である。但し、映画タイトルのデータは文献と同じものではない。なお、本実験で用いる FineSlider は NeXTstation 上でインプリメントしたもので、被験者はマウスによって操作を行う。

・被験者

被験者は、25 才から 34 才までの男性 8 名であり、全員、GUI を用いたシステムの使用経験者である。また、実験を行う前に FineSlider と本実験についての説明を行い、十分にトレーニングを行った。

・スライダの外観

実験に用いた FineSlider の外観は 3 つの微調整関数とも同様である。図 2 にその外観を示す。スライダは、FineSlider 本体、Find 欄、Movie 欄、Time 欄、Rand ボタン、Start ボタン、Done ボタン、微調整関数選択ボタンから構成されている。

FineSlider 本体におけるノブの可動領域長を 262pixel とした。従って、1pixel 分に約 38 個のタイトルが配置されていることになる。また、ノブの横幅は 18pixel とした。

Find 欄は、検索するタイトルが表示される部分で、このタイトルは、ユーザが Rand ボタンをクリックすることでランダムに選ばれる。Movie 欄は、FineSlider によって示される現在のタイトルが表示され、また、Time 欄は、検索の開始から終了までの時間、すなわち、Start ボタンがクリックされてから Done ボタンがクリックされるまでの時間が表示される。なお、Start ボタンがクリックされると FineSlider のノブはタイトルリストの第一項目の位置に移動する。

・微調整関数

実験で準備した微調整関数 $f(x)$ は次の 3 つである。

$$\begin{cases} \text{Func.A} & f(x) = (\kappa \cdot x)^2 + 1 \\ \text{Func.B} & f(x) = \sqrt{\kappa \cdot x} + 1 \\ \text{Func.C} & f(x) = \kappa \cdot x + 1 \end{cases}$$

ただし、 κ は任意定数である。また、 x は微調整領域の任意の点とノブ位置との距離 d に関する中間変数であり、ノブの横幅を a 、ノブの両端にある微調整単位が常に 1 である範囲を b とすると次式で表現される。

$$x = \begin{cases} d - (a/2 + b) & d > (a/2 + b) \\ 0 & a/2 < d \leq (a/2 + b) \\ - & 0 \leq d \leq a/2 \end{cases}$$

ノブの端から距離 d までの部分に微調整単位が常に 1 の領域を設けたのは、微調整の最終局面における操作性をよくするためで、この部分を選択し続けると、微調整単位 1 がゆっくりと加算される。本実験では、 $b = 45[\text{pixel}]$ とした。

今回準備した微調整関数のグラフを図 3 に示す。上式は、連続関数のように表現しているが、実際には距離 d は離散量であるので、3 つの関数は図中の階段状関数としてインプリメントされている。本実験では、上式において $\kappa = 1/10$ としたので、3 つの関数の交点は、距離 d について $x_1 = 10[\text{pixel}]$ で、微調整単位 $f(x_1) = 2$ となる。

・タスクと評価尺度

検索タスクは、以下のような手順で行う。

Step.1 被験者は Rand ボタンをクリックする。これにより、検索を行なう映画のタイトルがランダムに選択され、Find 欄に提示される。

Step.2 被験者は、検索するタイトルを十分認識した後、Start ボタンをクリックし、検索を開始する。

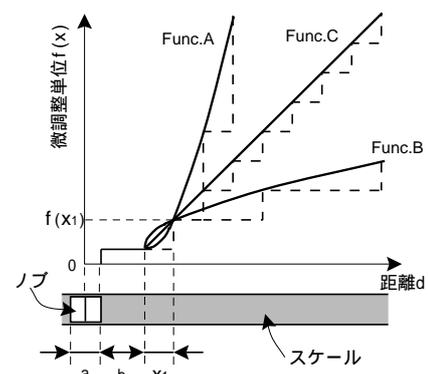


図 3: 微調整関数

表 1: 各被験者の検索時間

	被験者1	被験者2	被験者3	被験者4	被験者5	被験者6	被験者7	被験者8	平均(標準偏差)
Func.A	16.95	15.80	13.32	18.67	14.54	23.64	13.79	15.29	16.50(3.11)
Func.B	13.40	13.51	13.15	18.23	14.31	20.50	13.24	15.08	15.18(2.55)
Func.C	11.65	14.15	12.73	16.25	13.52	18.00	13.17	10.90	13.79(2.19)

単位[sec.]

Step.3 検索中のノブ位置に対応する映画タイトルは Movie 欄に表示され、被験者は検索タイトルが検索できたら Done ボタンをクリックする。

各ユーザは、各微調整関数ごとに上記手順に従って 20 タイトルの検索を行う。なお、3つの微調整関数に関する実験の順序は、それぞれの被験者でランダムであり、それぞれの微調整関数に関する実験の前に数回の練習タスクを行った。

また、評価尺度は、上記タスク中の Start ボタンをクリックしてから、Done ボタンをクリックするまでの時間である。さらに、一通り実験が終了した後、被験者に対して使いやすさに関するインタビューを行った。

4.2 実験結果

・実験結果と考察

実験結果について、3つの微調整関数に対する各被験者の検索時間の代表値として、20回の検索タスクの平均達成時間を採用する。表1と図4に各被験者の検索時間を示す。

実験結果から、ほぼすべての被験者に関して、検索時間の速さが Func.C、Func.B、Func.A の順序になっていることが分かる。また、検索時間の標準偏差は Func.A が最も大きく、Func.B、Func.C の順で小さくなっている。すなわち、Func.C は他の2つに比べて、ユーザによる検索時間のばらつきが小さく、平均的に速く検索を行うことができるものと考えられる。

また、インタビューの結果、次ぎのような意見が多かった。

1. Func.C は微調整領域に割り当てられる微調整単位がある程度予想できる。
2. Func.A は微調整領域のある地点から突然微調整単位が大きくなって微調整しにくい。

意見1は、Func.C が最も速かったことを裏付けるものである。一方で、意見2は、Func.A のようにノブの回りの微調整単位が1の領域を越えると突然微調整単位が増加するような関数を用いると微調整しにくくなることを示唆するものである。また、今回のインタビューでは Func.B についての意見は得られなかったが、Func.B は Func.A よりも全体的に微調整単位を小さくするため、その分だけ Func.A よりも遅くなっていると考えられる。

以上のことから、本実験で用いた3つの微調整関数のうちでは、ユーザが微調整単位の分布を容易に認識できる Func.C が最適であると考えられる。

また、AlphaSlider を用いて同様の映画タイトル検索のタスクを行っている文献 [2] の実験結果では、提案された4つの AlphaSlider に関して検索時間が、通常の被験者で 24 から 32 秒で、AlphaSlider のエキスパートで 13 から 19 秒である。実験環境が少々異なっているが、本実験では、3つの微調整関数

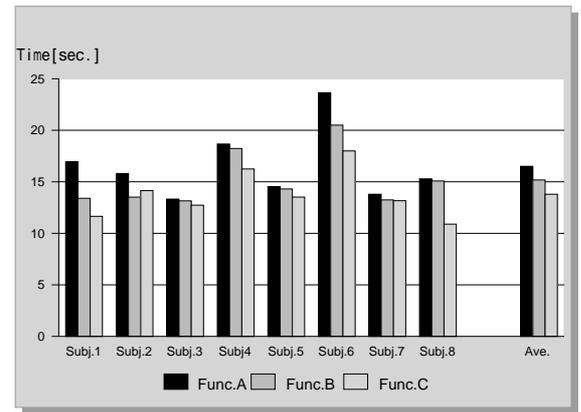


図 4: 各被験者の検索時間

に関して 13.8 から 16.5 秒であるので、AlphaSlider のエキスパートなみの結果が得られていることになる。

・結論

以上の実験結果より、FineSlider は広範囲のレンジを持つパラメータの値設定、ワード検索を効率的に行うことができると考えられ、FineSlider における微調整関数として、Func.C のような線形関数が有効であると考えられる。

しかし、線形関数を用いたとしてもその傾きが大きい場合は Func.A のように微調整しにくくなる可能性があり、すべての線形関数が最適であるとはいえない。従って、対象とするパラメータのレンジによって線形関数の最適な傾きを求める必要性があると思われる。

5 まとめ

本稿では、限られたスペースで広範囲のレンジを持つパラメータの値設定を、正確に、効率的に行うことのできるスライダとして FineSlider を提案した。そして、実験によって FineSlider の有効性を示し、また、FineSlider における最適な微調整関数について考察した。

本稿における実験は、NeXTstation 上でインプリメントした FineSlider をマウスによって操作したが、携帯情報端末上でペンを用いて操作しても同等に有効であると考えられる。

今後の課題としては、対象とするパラメータのレンジにより、あるいは使用するユーザにより微調整関数をチューニングすることが挙げられる。

参考文献

- [1] M. Osada, H. Liao and Ben Shneiderman, "AlphaSlider: Searching Textual Lists with Slider", *Progress in Human Interface*, vol.3, No.1, pp.5-14, Apr, 1994.
- [2] C. Ahlberg and B. Shneiderman, "AlphaSlider: A Compact and Rapid Selector", *Human Factors in Computing Systems, CHI'94 Conference Proceedings*, pp.365-371, Apr, 1994.