

ユーザビリティ評価システム「UI テスタ」における 評価機能の検証

旭 敏之[†] 岡田 英彦[†] 井 関 治[†]

製品のユーザビリティを向上させるにはその評価が不可欠であるが、従来の方式では実施上の制約が多い。本論文では、ユーザの対話履歴分析をベースとするユーザビリティ評価支援システム「UI テスタ」を提案し、評価機能の有効性を検証する。UI テスタではユーザ対話履歴とユーザインタフェース構造との差異を検出することで、ユーザビリティ問題点の抽出を支援する。このため、対話履歴の視覚化方式である対話構造ダイアグラムと、複数ユーザ間に共通する誤対話パターンを抽出する共通誤対話分析機能を提供した。実用性と問題点抽出能力の2側面から本機能の有効性を検討した。前者の検証のため、実際のファクシミリ製品を対象にしたユーザビリティ評価を実施した結果、評価の非専門家1人が18人分の対話履歴データを約8時間で分析することができた。抽出された9項目の問題点のうち、5項目を出荷前に改善することができ、製品開発サイクルにおける実用性を実証した。また問題点抽出能力検証のため、ヒューリスティック技法で同じ対象を評価し、抽出された問題点を比較した。全体の問題点のうち、UI テスタで非専門家が抽出できたものが1/3、誤対話パターンを手がかりとして抽出可能なものは6割程度であることが分かった。本結果をベースに、UI テスタの機能強化やユーザビリティ評価方法論の確立に取り組んでいく。

Evaluation of Testing Functionalities for the Computer Aided Usability Testing Tool: "UI-tester"

TOSHIYUKI ASAH[†], HIDEHIKO OKADA[†] and OSAMU ISEKI[†]

Usability testing methods are necessary for insuring the usability of product. Traditional methods, unfortunately, have some disadvantages. To overcome these disadvantages, this paper proposes a computer aided usability testing tool named "UI-tester" which evaluates usability by analyzing user-machine dialogue data. A dialogue visualization technique called dialogue structure diagram and a common erroneous pattern extraction function are provided for usability analysis. Effectiveness of the functions are verified from the viewpoint of practical value and problem extraction capability. Data from a feasibility study being conducted for a facsimile product development project shows that the evaluation functions work effectively in extracting nine usability problems, and that user interface redesigns are completed without additional delays in the product development cycle. Comparison of the testing results from those of the heuristic evaluation method shows that 1/3 of the usability problems can be extracted with UI-tester, and that the erroneous dialogue patterns indicate more than 60% of the problems. Research plan includes improvement of usability analysis functions and establishment of usability testing methodology.

1. はじめに

情報処理機器のユーザビリティ^{*}は、機能や性能とともに製品価値を左右する重要な要因である。ただしこれを定量/定性的に厳密に分析することは難しく、デザイナーや開発者のセンスに依存したり、評価や改善が場当たり的になりがちであった。ユーザビリティ問題には曖昧な点が多く理論的アプローチが困難であるが、

それだけに、これを現実的にどう対処するかといったまさに工学的な解決策が希求されている^{1),2)}。

ユーザビリティ評価は製品に内在する問題点をユーザの視点から明らかにすることが目的である。「典型的なユーザをある統制された環境に置き、製品を用いて目的のタスクを実行させるテスト」と定義づけされている³⁾。評価技法としてはモニタリングやプロトコル解析が代表的であり、ビデオ記録されたユーザ操作状

[†] NEC 関西 C&C 研究所

Kansai C&C Research Laboratories, NEC Corporation

^{*} 一般には「使いやすさ」と訳されるが、その他にも「学習の容易さ」「操作効率」「エラーの起こしにくさ」なども含んだ総称的な用語として使用することが多い。

況を評価者が詳細に分析することが基本である。これらは問題点抽出能力が高く実施例が多い一方で、次のような欠点も指摘されている^{4),5)}。

- データ分析に要する工数が多い。特に操作状況を記録したビデオテープの解析に時間がかかる。
- したがって評価対象の母集団（参加ユーザ数）を大きくできず、結果がユーザや評価者の個人差に影響されやすい。
- データ解析に専門家のスキルが要求される。ユーザの操作状況から、問題点を的確にかつもれなく指摘できる能力が要求される。

これらの欠点は、製品開発の現場におけるユーザビリティ評価活動の定着を阻む大きな原因となっている。これに対し、ユーザビリティ評価のコンピュータ支援方式が提案されてきている。これらは次の2種類に大別することができる。

(1) ビデオデータ分析支援型

ビデオテープに記録されたデータの分析を支援するツールである。パソコンからのVTR制御、記録された場面に同期した記録者コメントの挿入、テープ編集を容易にするための諸機能、評価対象の画面や対話履歴の同時記録、などが実現されている^{6),7)}。ただしこれらのツールの主な目的は、評価者のデータ解釈を容易にするためのいわば前処理である。記録テープのレビューをベースにする点は同じであり、工数的には飛躍的な改善は望めない。また、ユーザビリティ問題点の抽出においても、依然として評価者のスキルに負うところが大きい。

(2) 対話履歴データ分析支援型

自動記録が可能なユーザの対話履歴を分析するためのツールである。テープ編集など前処理的な作業ではなく、問題点抽出そのものの支援を試みる。繰返し操作に問題点があると仮定しこれを抽出する方式⁸⁾や、標準手順との差異を検出する方式^{9),10)}、操作時間や失敗/むだ操作等を分析する方式¹¹⁾、状態遷移を視覚表示して問題点を推定する方式^{12),13)}などが提案されている。

筆者らが提案する自動評価システム「UI テスタ[☆]」は(2)のカテゴリに属し、ユーザビリティ上の問題点抽出そのものを支援し評価工数を大幅に短縮するとともに、必要なスキルを最小限にすることが狙いである。ただし、この方式では分析対象のデータが対話履歴で

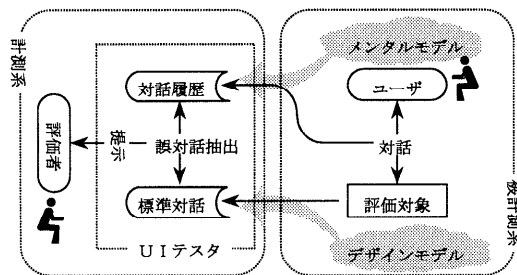


図1 UI テスタの設計方針

Fig.1 Concept of the usability measurement method.

あり、ビデオ記録で得られるようなユーザの思考や感情といった情報は欠落している。したがって、現実の製品評価における実用性、すなわちどの程度詳細な評価結果が得られ、それがどれくらいユーザビリティ改善に寄与できるかを実証的に検討する必要がある。これまでの研究では方式の提案に重点が置かれ、上記の検討は十分ではなかった。以下本論文では、UI テスタの機能的概要を紹介するとともに、実際の製品開発に適用した結果と、従来評価方式との比較結果について報告する。

2. UI テスタの設計方針

一般に、ユーザによる操作対象システムの理解や解釈（メンタルモデル）と、開発者がシステムを設計する際に有するイメージ（デザインモデル）との食い違いが、ユーザビリティ上の問題を生むとされている^{14),15)}。両モデルを厳密に記述することは困難だが、ユーザ対話履歴にメンタルモデルが反映されており、かつデザインモデルは標準の対話手順として具現化されていると考えると、ユーザ対話履歴を標準対話手順と比較しながら分析することで、コンピュータによるユーザビリティ分析が可能になると考えられる（図1）。ただし、一般に対話履歴をベースにしたユーザビリティ分析には、「対話履歴データの読み取りが難しい」「対話パターンとユーザビリティ問題点を関連づけることが難しい」「複数ユーザのデータを処理する機能がなく、個人差の影響の排除が難しい」、などの困難がある。UI テスタでは以上の点を考慮し、次のような設計方針のもとで開発を進めた¹⁶⁾。

- 誤操作^{☆☆}を含んだユーザの対話履歴を計測対象と

☆☆ 本論文では、便宜的に装置の設計どおりの対話手順を「標準対話」、ユーザが行った操作でこれらと異なる部分を「誤対話」と称している。これは、もちろん装置が提供する手順の方が「正しい」という意味ではなく、むしろ、これらを含んだユーザ操作の中にUI改善の手がかりがあると考えている。

☆ UI=ユーザインタフェース

し、これを評価可能（標準対話手順と比較可能）な形式にコード化する。

- まず評価者が対話状況を容易に把握できるように上記データを視覚化する。特に対話履歴と標準対話手順との差異が明確になるような視覚化機能を提供する。
- 多数ユーザの対話履歴を分析することで、ユーザビリティ問題点を示唆する対話ボタンを選択的に抽出する。

UI テスタの開発にあたっては、ファクシミリを対象として選択した。このタイプの UI では、タスクを達成するための標準の対話手順が 1 通りに規定できるため誤対話の判定が比較的容易であり、上記設計方針の効果検証に適していると判断したからである。また、標準対話手順自体もある程度複雑であり、ユーザビリティ問題点を含んだ誤対話が多く抽出できると予想されたことも理由の 1 つである。したがって、以下本稿で展開する議論は、たとえば GUI など対話手順を柔軟に変更できるものや、逆に画面変化がほとんどないような単純な UI に対してはそのまま適用できないことに注意を要する。

UI テスタでは実機に接続する形態をとらず、操作パネル設計/シミュレーションツールと結合して対話履歴データを記録する仕組みを実現した。結果的に評価用実機プロトタイプが不要となり、製品開発の早期段階で評価/改善が可能となった¹⁷⁾。

このようなツールを利用するうえでの問題点としては、まず実機の操作感覚とのズレ、特にボタン形状など物理的操作感覚の相違が予想される。ハードウェアの身体的適合性などを評価するためには不適切である。ただし、UI テスタでは上記に述べたように対話手順などソフトウェア側面の評価に焦点を当てており、この影響は少ないものと考えられる。

また、シミュレータを利用した評価ツールを考えると、シミュレーションデータを作成する工数が問題になる可能性がある。ただし、今回用いたツールはもともファクシミリ開発用に実用化されていたものであり、通常、仕様設計の完了時には UI 動作が細部まで忠実にシミュレートできる状態になっている。したがって本論文で評価工数を議論する場合には、シミュレーションデータ作成の工数は考慮しなかった。評価用に新たにデータを作成しなければならないようなケースでは、その工数も含めてテスト効率を議論すべきであると考ええる。

以下 3 章では、UI テスタで提供するユーザビリティ評価機能について説明する。

3. ユーザビリティ評価機能

2 章で述べた設計方針を具体化するため、まずシミュレータ上で実行されたユーザの対話履歴（操作と画面の内容を時間順にコード化したもの）を実時間で記録する機能を実現した。この対話履歴データをベースに、標準対話手順との差異を明示する対話構造ダイアグラムと、多数ユーザの対話履歴を分析するための共通誤対話分析機能を提供した。

3.1 対話構造ダイアグラム

一般にファクシミリの UI では、ある特定のタスクに対する対話手順は画面の遷移を表現した 1 本のパスで表すことができる。この性質を利用して、あらかじめ入力された標準の対話履歴をベースに、ユーザの誤対話部分だけを別のアークとノードで表現する対話構造ダイアグラム（DSD）方式を提案した¹⁸⁾。表示例を図 2 に示す。

DSD では対象システムの画面状態がノードで、ユーザの操作が各ノードを結ぶアークで表現される。図において「初期状態」、「機能選択」などは画面状態の属性を表しており、図面上で段差として区別することで対話状況の把握を容易にしている。図 2 (a)~(d) の各部について以下に説明する。

- (a) 帯状の表示で標準の対話手順を示す。本手順と各ノードの属性は評価者があらかじめ入力する。複数回現れる画面には「A」「B」などのラベルを割り当て、同じものであることを表現する。大ノードは全面的に画面が書き換わった場合、小ノードは画面の一部だけが変化したことを示している。標準対話手順以外のノード、アークはすべてユーザによる誤対話を表現している。
- (b1), (b2) ある時点において標準の対話手順と異なる操作を実行し、その結果異なる画面に直接ジャンプしたことを示すものである。誤操作したことによりそれ以前の状態に戻る場合 (b1) と、必要な画面を飛ばしてしまう場合 (b2) が考えられる。
- (c1), (c2) やはりある時点においてユーザが誤操作を実行し、他の標準対話上の状態に遷移したことを示すアークである。ただし、(b1), (b2) が誤操作直後に他の標準対話画面にジャンプしたことを表現しているのに対し、(c1), (c2) ではその間に少なくとも 1 つ以上の標準対話にない画面を経たことを示している（標準対話に戻るまでの誤対話は、誤対話アーク上に追加されたノード 1 個で代表して表現されている）。(c1) は標準対話手順の同じ画面に復帰した場合、(c2) は異なる画面にた

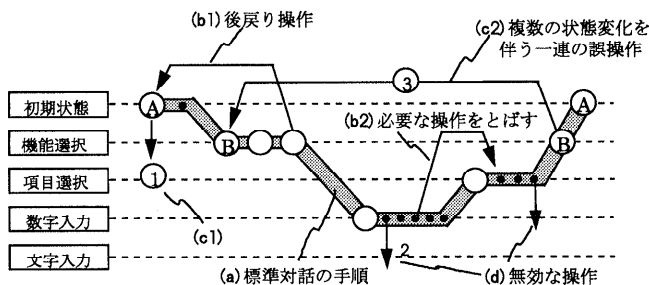


図2 対話構造ダイアグラム。太い帯で結ばれた大小のノードで標準の対話手順を表す。ユーザの誤操作は (b)(c)(d) のような付加的なアークやノードで表現する。

Fig. 2 Dialog structure diagram. A standard operations sequence is represented as nodes tied with broad arcs. Erroneous user operations are drawn with fine arcs referred as (b)(c)(d) in the figure.

どり着いた場合を示している。なお、誤対話アーク上のノードの中の数字 ((c2) の例では「3」) は、このアークとノードで表現される誤対話が3回出現したことを示している。

- (d) 状態変化をともしない誤対話を表現している。装置に受け付けられなかった誤対話、あるいは操作しても画面の変化がなかった場合を示している。矢印アークの横に配置した数字 (図2の例では「2」) で、同様の誤対話が起こった回数を示している。

以上、対話構造ダイアグラムの表現方式を説明した。タスクを達成するための標準対話手順が1通りであり、かつユーザが何らかの形でそのタスクを実行できた (すなわち、先頭ノードから対話を開始し最終ノードにたどり着いた) 場合には、あらゆる誤対話パターンが図2(b)~(d)のいずれかで表現できる。

上記(c)において、いったん誤対話に陥ってから標準の手順に戻るまでの一連の対話を、単一のノードで代表して表現することを説明した。これは、いったん間違った画面に入って操作が分からなくなった場合、一般に非常に多くの画面を探索するケースが多く、これを詳細に表現するとダイアグラムが過度に複雑になるからである¹⁸⁾。本表現法はその弊害を防ぐものである (誤対話の詳細を参照したい場合は、このノードをダブルクリックすることで別画面に状態遷移図が表示される)。また経験的にユーザビリティ評価では、「どの画面から」「どんな誤操作をして」標準対話から逸脱したかに注目することが多く、いったん異なる画面に迷い込んでしまった後の対話は問題点抽出にさほど重要でないことも、本表現法を採用した理由である¹⁸⁾。すなわち、画面をできるだけシンプルにし、誤対話の発生箇所を発見しやすくすることが、DSDの狙いである。

3.2 共通誤対話分析機能

誤対話はユーザビリティの問題点を示唆する重要な指標である。ただし、単独のデータでこれを確定的に議論することは困難であり、あえて行えば評価結果が恣意的なものになる危険性が高い。実際の誤対話ボタンの中には個人差によるものや偶然に生じたものなどが含まれており、これらを区別することが難しいからである。これは対話履歴ベースの評価方式を考えるうえで本質的な問題である。この問題を解決する1手段として、UI テスタでは共通誤対話分析機能を提供する¹⁹⁾。すなわち、複数ユーザ間で共通的に観測された誤対話は重要度が高いと見なし、これらを選択的に抽出する。

実際には同一のタスクを複数のユーザに実行してもらい、それぞれをファイルとして記録する。分析時に複数のデータファイルを指定し、さらに共通度 (%) を指定することで、その値以上の割合のユーザに共通して現れた誤対話ボタンだけをDSD上に表示する。図3は共通度を変化させたときのDSDの変化例を示したものである。評価者は表示の変化を観察しながら、共通度をインタラクティブに調整することで、必要と思われる誤対話ボタンを絞り込んでいくことができる。図3の例では共通度が10%のときには多数(42個)観測された誤対話ボタンが、30%では12個、60%では1個に絞り込まれている。なお、図3において各アーク横に示された()内の数値は、同じ誤対話ボタンを示したユーザの数を表している。

3.3 その他の機能

その他、評価の際に補助的に用いることを想定した機能として、対話履歴を状態遷移で表現した状態遷移ダイアグラム¹⁶⁾、操作時間をグラフ化する共通操作時間グラフ²⁰⁾などを提供している。前者是对話の全般的な複雑さや繰返しボタンの発見に有効であり、後者は

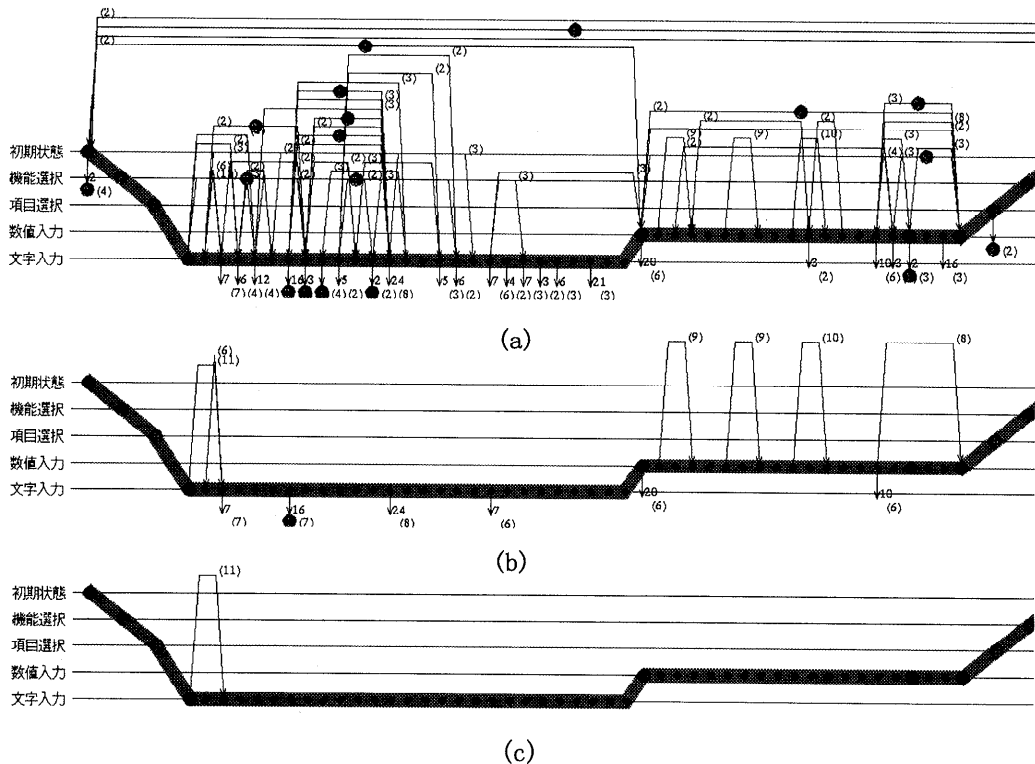


図3 DSDの変化。共通度 = 10% (a), 30% (b), 60% (c)。

Fig. 3 Change of DSD form. Degree of commonness = 10% (a), 30% (b), 60% (c).

操作に時間がかかる箇所を発見するのに有効である。

4. 評価機能の検証

この章では、UI テスタの評価機能の検証を試みる。まず、実際に開発中にあるファクシミリに適用し、製品開発プロセスにおける実用性を検証する。特に、評価者がDSDと共通誤対話分析機能を利用し、UI問題点を示すボタンを実際に抽出できるかどうか注目する。さらに、UI テスタによる評価結果と従来評価技法により抽出された問題点とを比較することで、誤対話ボタン抽出方式による評価能力を検証する。

4.1 製品開発プロセスへの適用

4.1.1 テスト環境

UI テスタを用い、実際に開発段階にあったあるファクシミリ製品のユーザビリティを評価した。まず設計ツールで操作パネル部分の設計を行い、コンピュータ端末上でUI動作をシミュレートする環境を用意した。以下に実験の条件を示す。

被験者： 18人（年齢20～30歳。事務職2人，研究職16人。いずれもファクシミリを日常業務で利用するが、対象機種種の操作経験なし）

タスク： 以下の3種類を作業直前に各被験者に指示。

() 内は手順数（押下ボタン数）を示す。

(1) 内蔵カレンダー/タイマの設定 (17)

(2) 発信元の名称と電話番号の登録 (40)

(3) 特殊（ワンタッチ）キーへの相手先電話番号や通信モードの設定 (53)

要領： 被験者には評価対象とタスクの指示のみが与えられ、マニュアルの参照は不可とした。各タスクが終了するごとに、対話履歴データをファイルとして記録した。また、シミュレータの制御のため実験者が付き添ったが、操作に関するアドバイスはいっさい行わなかった。

念のため実験状況をビデオ記録したが、データ分析には利用しなかった。各被験者とも自力で全タスクを完了した。標準の作業時間（＝対話手順を熟知した実験者の作業時間）と、被験者の作業時間の平均値と最大値を以下に示す（単位はいずれも秒）。

（タスク1）標準：53，平均：100，最大：173

（タスク2）標準：100，平均：325，最大：655

（タスク3）標準：132，平均：380，最大：839

最大でも15分（900秒）を超えたセッションはなく、各タスクとも被験者にとってさほど大きな負荷になるものではなかったことが分かる。

表1 評価機能検証実験で抽出された問題点

Table 1 Problems detected from the feasibility study.

タスク2:
①発信元の名称登録時に違うボタンを選択。 ②漢字入力時に確定操作を誤る/怠る。 ③文字が辞書登録されていないことが伝わらない。 ④電話番号登録を開始するボタンがわからない。 ⑤電話番号のポーズ記号「-」の入力を誤る/怠る。
タスク3:
⑥宛先を登録するためのボタンがわからない。 ⑦半角入力モードにするためのボタンがわからない。 ⑧記号文字「-」の入力方法がわからない。 ⑨操作不可の領域を押下する。

4.1.2 適用結果

ユーザ対話履歴を分析した結果、タスク2, 3から表1に示す9項目の問題点を抽出することができた。対話履歴データの収集には従来とほぼ同じ1週間程度を要したが、分析は18人分のデータを1日で完了することができた。

本問題点を対象機種の開発部門と検討した結果、UIの改善が必要と認められた5項目について設計を変更した(他の項目も問題点として確認されたが、旧モデルとの一貫性を優先するなどの理由から修正されなかった)。設計変更後のモデルについて、上記実験に参加しなかった被験者6人で追実験を行い改善効果を確認した。その後、実機試作を経て製品として出荷した。従来方式では困難であった当該機種への評価結果のフィードバックを、時間的な余裕を持って実現することができた。

4.1.3 問題箇所の抽出

ユーザビリティ評価の未経験者1人に評価者としてデータ分析を依頼した。評価者はUIテストの開発者の1人であり、その機能や操作法は熟知していた。データ分析は評価者の裁量で自由に作業を進めてもらった。当該者に対する作業後のインタビューにより、以下のような評価作業手順を確認した。この作業を1日(約8時間)で完了することができた。

- U1) タスク1に対して、全被験者の対話履歴ファイルを指定し、共通誤対話ボタンを抽出する。
- U2) 共通度を操作することで分析対象となる誤対話ボタンを絞り込み、その発生箇所(=画面)と誤操作内容(=誤って操作したボタン)を特定する。
- U3) 上記発生箇所に相当するファクシミリの画面と誤操作の内容を検討し、UI上の問題点に相当すると思われるものを評価結果として記録する。

U4) タスク2, 3に対してU1)~U3)を繰り返す。

一方、プロトコル解析で評価する場合には、一般的には次のような手順が必要となる³⁾。各手順を今回の実験に適用した場合に、経験的に予想される所要工数も目安として示す(1人が担当した場合)。

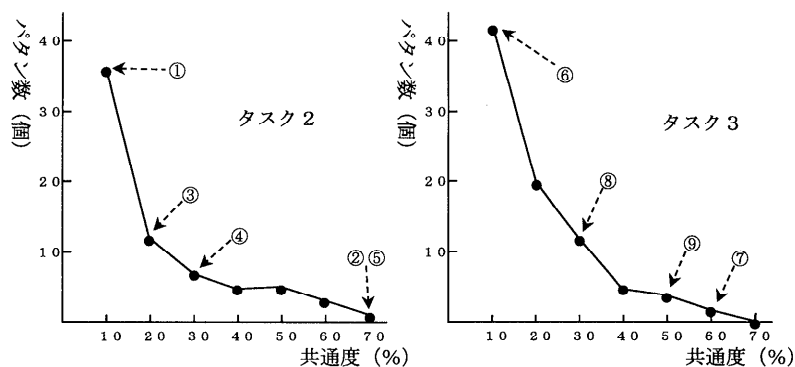
- P1) デイクテーション: 1~3週間
記録テープから発話内容を書き取る。
- P2) セグメンテーション: 2日
操作上のトラブルが含まれている期間を抽出する。
- P3) トラブル抽出: 3日
上記トラブルの内容を特定する。
- P4) 問題点抽出: 1日

P3)の原因を検討し、UI上の問題点を抽出する。したがってデータ分析全体としては約2~4週間を要する作業量である²¹⁾。実際の作業は評価者の裁量による部分が大きく、上記の所要工数は必ずしも正確ではない。ただし、一般にVTR分析型の評価手法ではデータ分析にその記録時間の3~10倍の時間を要するとする報告もあり²⁾。今回データ記録に1週間を要したことを考えると、過小な見積りではないと考えられる。

これらのことから、「UIテストによりプロトコル解析で2~4週間要していたデータ分析期間を1日に短縮することができた」といえる。この理由としては、まずプロトコル解析で最も工数を要するデイクテーション作業が不要になった点があげられる。また、P2~P4に関しても各被験者データごとに作業が必要になるため、その数に比例して負担が増える。UIテストの場合は多数データを一括して処理するため、被験者数が多いことに起因する工数の増加はほとんどない。

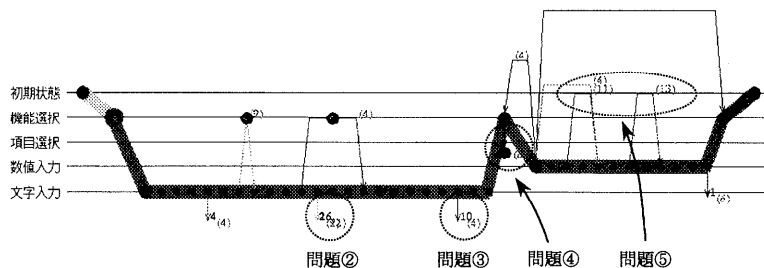
タスク2, 3の対話履歴データに対し、共通度を変化させたときの共通誤対話パターン数と、表1の問題点を示す各ボタンがどの時点で現れたかを図4(a)に示す。また参考までに、図4(b)には(タスク=2, 共通度20%)のときのDSDと問題点を示すパターンの分布を示す。

図4(a)から、共通度をいったん高い値(60%以上)に設定し、これを下げていきながら新しく現れたボタンを順次分析していくことによって効率良く評価できることが分かる。共通度が10%(=18人中2人)の場合にはボタン数が40前後にもなり、各ボタンの内容を調べていくにはかなりの労力を必要とする。ただし、ここまで共通度を下げることで発見できた問題点も存在する(①, ⑥)ことから、詳細な分析ではこれらにも着目する必要があることが分かる。



(a) 共通度を変化させたときの誤対話パターン数の変化

①～⑨は表1の各問題点に対応する誤対話パターンが共通度を下げていったときにどの時点で表示されるかを示したもの。



(b) 問題点を示す誤対話アークの例 (共通度=20% タスク2)

図4 共通度と共通誤対話パターン

Fig. 4 Degree of commonness (DOC) and common erroneous patterns.

4.2 評価能力の検証実験

4.2.1 検証方式

1章でも述べたように、UI テスタの対象とする対話履歴データにはユーザの心理的態度に関する情報が欠落しており、本来検出されるべき問題点が見落とされる可能性も十分考えられる。ここでは、UI テスタの誤対話パターンの検出による問題点抽出能力がどの程度であるかを推定する。

この検討には、比較のための基準として「本来抽出すべき問題点」を明らかにする必要がある。しかしながら、現実にはユーザビリティ問題には客観的な基準がなく、どの問題点を抽出すべきであるかを明確に規定することは難しい。そこで本研究では、すでに利用実績が多く実践的な技法として評価が高いヒューリスティック評価技法を同じファクシミリ製品に適用し、これで得られた問題点を基準と見なすことで評価能力の検証を試みる。

ヒューリスティック評価では、たとえば「ユーザの用語を使用する」「一貫性を保つ」といった10項目の原則を基準とし、複数の専門家が評価対象を個別に吟味し、その結果を持ち寄り集中的に問題点を抽出す

る技法である^{22)~24)}。ユーザが参加しないので厳密にはユーザビリティ評価技法とはいえないが、プロトコル解析の3/4程度の問題点は発見できるとされている²³⁾。

以下に筆者らが実施したヒューリスティック評価の条件を示す。

- 参加者：評価経験者4名（経験年数：7, 3, 2, 2年）
- 方法：
 - (1) 各参加者が個別に評価対象機を操作する。
 - (2) 10項目の原則²²⁾を基準に各自が問題点を洗い出す。比較のためタスクは4.1.1項の3つに限定する。
 - (3) 各評価者は2パス評価（1回目は全体的流れ、2回目に個々のUI要素を検討）を心がける。
 - (4) 各評価結果を持ち寄り、討論により問題点を確定する。
- 評価期間：試行に約1週間、討論とまとめに1日。

4.2.2 UI テスタの評価範囲

ヒューリスティック評価で抽出された24項目の問題

点に、表1で示したUIテストで検出できた問題点のうち8項目が含まれた(UIテストで検出できヒューリスティックで検出できなかったものが1項目あった)。すなわち、両結果を総合すると25項目の問題点が発見されたことになり、以下ではこの数字を基準として評価能力を検討する。25 - 9 = 16項目はUIテストの評価者が発見できなかった問題点であるが、ヒューリスティック評価の後に対話履歴データを見直したところ、このうち6項目には何らかの誤対話ボタンが見られた。したがって、25項目の問題点をまとめると次のようになる。

- A) ヒューリスティック評価で抽出でき、UIテストでも確実に抽出できたもの ... 8
- B) ヒューリスティック評価で抽出でき、誤対話ボタンが観測されたもの ... 6
- C) ヒューリスティック評価で抽出できたが、誤対話ボタンとしては観測されなかったもの ... 10
- D) UIテストで抽出できたが、ヒューリスティック評価では見落とされたもの ... 1

A)のケースで「確実に抽出できた」という表現は、4.1の実験で評価の非専門家でも抽出できたことを根拠としている。参考までに、B)、C)に相当する問題点の内容を表2に示す(D)に相当する問題点は表1の⑥、A)に相当するものはそれ以外の8個の問題点である)。

上記の結果からヒューリスティック評価との比較においては、

- UIテストの誤対話分析機能により、非専門家でも1/3程度(≒ (A+D)/25)の問題点は抽出できる。
- 全体の3/5程度(≒ (A+B+D)/25)の問題点は誤対話ボタンとして観測できる。

という2つの見解が得られる。仮に文献23)に従いヒューリスティック技法の問題点発見能力をプロトコル解析の3/4程度であると仮定すると、上記2観点の数字はそれぞれ1/4、1/2程度ということになる(プロトコル解析技法と比較した場合)。

今回の実験は単一の製品だけを対象にしたものであり、一般的かつ客観的な結論を導出することは無理がある。前者の見解をUIテストを製品評価に適用する際の目安に、後者を対話履歴分析機能を強化していく際の目標値として活用していく。

C)は「ユーザビリティ上の問題ではあるが、誤対話を誘発するような性質のものではない」問題点の存在を示している。たとえば「操作方法に一貫性がないが、ユーザは類推できた」「表示のデザインが悪いが

表2 ヒューリスティック評価で抽出された問題点

Table 2 Problems detected with the heuristic evaluation.

B) 誤対話ボタンが観測されたもの

- ①モード表示領域とボタンの区別が紛らわしい。
- ②現在時刻入力画面で、入力手順のガイダンスが欲しい。
- ③矢印キーがどの表示オブジェクトに対し有効かが不明確。
- ④[現在時刻]は[現在日時]というラベルの方が正確。
- ⑤時刻登録が正常に行われたかどうかのフィードバックが欲しい。
- ⑥「プログラム」「自局短ダイヤル」の言葉の意味がわかりにくい。

C) 誤対話ボタンが観測されなかったもの

- ⑦[確定]キーを押下しづらい。[変換]キーの隣に配置すべき。
- ⑧電話番号と発信元情報は同じ画面で登録できた方がよい。
- ⑨ボタンイメージの中に文字を入力することがわかりにくい。
- ⑩同機能に[削除]と[1文字削除]の2通りのキーラベルが使われている。
- ⑪自局電話番号登録やワンタッチキー登録では[取消]キーが無い。
- ⑫[クリア]の意味が「現在時刻登録」と「発信元登録」との場面で異なる。
- ⑬未登録の[ワンタッチ]キーは未登録であることを明示すべき。
- ⑭カナ入力モードの時はキーラベルもカナ表記になるべきである。
- ⑮表示データの区切りが狭すぎて見づらい。
- ⑯[ワンタッチ削除]を押した後に確認が無い。

意味は通じる」といった問題がこれにあたる。たとえ誤操作に至らなかったとしても、主観的な印象に影響が大きい場合も考えられ、これらが必ずしも重要度の低い問題点であるとはいえない。この種の問題点を抽出できないことが誤対話ボタンに着目した分析方式の本質的な弱点である。他の評価方式と補完するなどの対策を検討する必要がある。

4.3 まとめ

以上、対話履歴分析をベースとしたユーザビリティ評価ツールUIテストの概要を説明し、その評価機能を製品開発プロセスへの適用と従来評価技法との比較という2つの側面から検証した。実際に開発過程にあったファクシミリ製品に対して出荷前ユーザビリティ改善を実現したこと、評価の非専門家が短期間にデータを処理できたことから、現場での実用性を確認することができた。

評価能力を検証するため、ヒューリスティック技法によるものと評価結果を比較した。推定値も含めて3つの評価技法の能力と工数を表3に示す。本実験で対象としたのは1機種だけであり、厳密な議論を進めるには無理があるが、大まかな見解としてUIテストによる評価の特長を以下の2点にまとめることができる。

- プロトコル解析に比べた場合、データ分析工数の短縮効果大きい。
- 所要工数はヒューリスティック評価とほぼ同様であるが、専門家の参加は前提としない。

反面、評価能力はプロトコル解析の1/4程度になることから、現実には評価技法を選択する局面では、上記特長とのトレードオフを慎重に検討する必要がある。ただし一般に(特に国内メーカーでは)評価期間と専門家の確保がユーザビリティ評価活動の障壁になること

表3 3種類の評価技法の比較

Table 3 A comparison of three usability evaluation methods.

評価技法	評価期間			発見率**
	データ収集*	分析期間	専門家(人)	
プロトコル解析	1週間	1～3週間	1～3	1
ヒューリスティック評価	1週間	1日	4	3/4
UIテスト	1週間	1日	不要	1/4

*ヒューリスティック評価の場合は専門家の試行期間
 **プロトコル解析のそれを1とした場合

が多いため、UIテストが有効な場面は少なくないと考えられる。

問題点全体の1/2程度が誤対話ボタンとして観測されたという結果は、今後の評価機能強化のための目標値としてとらえたい。すなわち、今後、注目すべき誤対話ボタンを強調するような適当な分析支援機能を付加することで、評価の非専門家でも半分程度の問題点は抽出できるようにしていく。その他に、今後の課題を以下に列挙する。

- 本論文で提案した共通誤対話分析機能を適用するためには、タスクを達成するための手順が単一のルートで記述できることと、かつ対話の進行が、画面/表示パネルの状態をノード、操作内容をアークとする状態遷移図として表現できる（すなわちDSDでユーザ対話履歴が記述できる）ことが条件になる。逆に、たとえば一般のグラフィカルユーザインタフェースなど正解の手順を1通りに規定しにくいようなUIに対しては、異なる視覚化あるいは分析機能を開発する必要がある。
- 本システムでは対話履歴そのものがユーザのメンタルモデルを表現していると仮定したが、その関連を詳細に検討したわけではない。ユーザビリティ評価に必要なメンタルモデルの記述およびその応用方式など、理論的な側面も並行して検討する必要がある。
- 現行の機能では、評価者に提示されるのは「問題のある箇所の候補（誤対話ボタン）」だけである。非専門家の利用を前提にする以上、問題点や改善案まで示唆するような機能も必要になると考えられる。さらに、他の評価技法やチェックリスト、ガイドラインなども含めた評価技術体系を確立し、その中でツールの利用方法も見直していく必要がある。

本システムは専門の評価者がいない製品開発現場で応用されることを想定して開発したものである。した

がって、そういった環境で定常的に利用されるようになって初めて目的を達成したといえる。現在、上記課題も含めて実用化のための研究を継続している。

参考文献

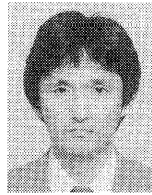
- 1) Tyldesley, D.A.: Employing Usability Engineering in the Development of Office Products, *The Computer Journal*, Vol.31, No.5, pp.431-436 (1988).
- 2) Nielsen, J.: *Usability Engineering*, Academic Press (1993).
- 3) 旭 敏之, 井関 治, 宮井 均: 製品開発におけるインタフェース評価—認知科学との接点として, 認知科学の発展, Vol.5, 講談社 (1992).
- 4) France, A.: Data Collection Techniques 1: Knowledge Elicitation and Inspection Methods, *Usability Testing and System Evaluation*, pp.113-148, Chapman & Hall (1994).
- 5) Asahi, T. and Miyai, H.: A Usability Testing Method Employing the "Trouble Model", *Proc. Human Factors Society 34th Annual Meeting*, pp.1233-1237 (1990).
- 6) Hoiem, D. and Sullivan, K.: Designing and Using Integrated Data Collection and Analysis Tools: Challenges and Considerations, *Behaviour & Information Technology*, Vol.13, No.1/2, pp.160-170 (1994).
- 7) Weiler, P., Cordes, R., Hammontree, M., Hoiem, D. and Thompson, M.: Software for the Usability Lab: A Sampling of Current Tools, *Human Factors in Computing Systems, Proc. INTER-CHI '93*, pp.57-60 (1993).
- 8) Siochi, A. and Ehrich, R.: Computer Analysis of User Interface Based on Repetition in Transcripts of User Sessions, *ACM Trans. Information Systems*, Vol.9, No.4, pp.309-335 (1991).
- 9) Kishi, N.: SimUI: Graphical User Interface Evaluation Using Playback, *Proc. 16th Annual International Computer Software and Applications Conference*, pp.121-127 (1992).
- 10) 来住伸子: X-Windowにおけるユーザ操作記録および再生ツールの再設計, インタラクティブシステムとソフトウェアII, 日本ソフトウェア科学会 WISS '94, pp.95-104 (1994).
- 11) 池本浩幸, 今村大輔: GUIの操作性評価ツール, 東芝レビュー, Vol.51, No.2, pp.45-47 (1996).
- 12) Cuomo, D.: Understanding the Applicability of Sequential Data Analysis Techniques for Analyzing Usability Data, *Behavior & Information Technology*, Vol.13, No.1/2, pp.171-182 (1994).
- 13) Guzdial, M., Santos, P., Badre, S., Hudson, S. and Gray, M.: Analyzing and Visualizing Log

Files: A Computational Science of Usability, Technical Report 94-08, GVU Center, College of Computing, Georgia Institute of Technology (1994).

- 14) Norman, D.A. and Draper S.W.: *User Centered System Design*, Lawrence Erlbaum Associates (1986).
- 15) Kieras, D. and Polson, G.: An Approach to the Formal Analysis of User Complexity, *International Journal of Man-Machine Studies*, Vol.22, pp.365-394 (1985).
- 16) 旭 敏之, 井関 治: 使いやすさ評価システム「UI テスタ」の提案, 情報処理学会ヒューマンインタフェース研究会資料, HI-38-1 (1991).
- 17) 旭 敏之, 岡田英彦, 井関 治, 松田良一: 製品の使いにくい部分を出荷前に修正するためのソフトウェアツールを開発, 日経エレクトロニクス, No.609, pp.111-120 (1994).
- 18) 旭 敏之, 井関 治, 岡田英彦: 「使いやすさ」の計測: UI テスタ, 第8回ヒューマンインタフェースシンポジウム論文集, pp.287-290 (1993).
- 19) 岡田英彦, 大津祐司, 旭 敏之, 井関 治: UI テスタにおける共通誤対話の分析, 情報処理学会HI 研究会報告, HI-50-4, pp.25-31, (1992).
- 20) 岡田英彦, 旭 敏之, 井関 治: UI テスタによる共通操作時間モデルの分析, 第10回ヒューマンインタフェースシンポジウム論文集, pp.649-652, (1994).
- 21) 旭 敏之: プロトコル・データによる製品の評価, 海保, 原田 (編), プロトコル分析入門, 第8章, 新曜社 (1993).
- 22) Nielsen, J., Molich, R.: Heuristic Evaluation of User Interfaces, Human Factors in Computing Systems, *Proc. CHI '90*, pp.249-255 (1990).
- 23) Nielsen, J.: Finding Usability Problems Through Heuristic Evaluation, Human Factors in Computing Systems, *Proc. CHI '92*, pp.373-380 (1992).
- 24) Nielsen, J.: Enhancing the Explanatory Power of Usability Heuristics, *Proc. CHI '94*, pp.152-158 (1994).

(平成8年8月26日受付)

(平成9年4月3日採録)



旭 敏之 (正会員)

1959年生。1984年大阪大学基礎工学部制御工学科修士課程修了。同年日本電気(株)入社。1992～1993年メリーランド大学客員研究員。現在, NEC 関西 C&C 研究所研究専門課長, 兼奈良先端科学技術大学院大学情報科学研究科博士課程在学。ユーザビリティ評価, HI におけるメディア応用の研究に従事。計測自動制御学会 HI 部会, UPS 各会員。



岡田 英彦 (正会員)

1970年生。1992年大阪府立大学工学部経営工学科卒業。同年日本電気(株)入社。現在, 関西 C&C 研究所に所属。ユーザインタフェース評価の研究に従事。本学会研究(1995年), 全国大会奨励賞(1996年後期)受賞。日本ファジィ学会会員。



井関 治 (正会員)

1954年生。1977年青山学院大学理工学部電気電子工学科卒業。1979年同大学院理工学研究科電気電子工学専攻修士課程修了。同年日本電気(株)入社。現在, (株) NEC デザインヒューマンインタフェースデザイン部デザイン課長, 兼 NEC 関西 C&C 研究所研究専門課長。ヒューマンインタフェースデザイン, ユーザビリティ評価, 福祉インタフェース等の研究開発に従事。1984～1985年メリーランド大学客員研究員。計測自動制御学会ヒューマンインタフェース部会, ACM 各会員。