

誰のための自動化？

稲垣 敏之

1. はじめに

人は誤るを常とする。人はエラーを起こさないよう努力する。それでもエラーは起こる。エラーの形態、発生状況も様々である。

閉じかけたドアを無理に開けてエレベータに乗り込んで階数ボタンを押した途端、それが目的階ではなく途中の階のボタンであったことに気がつく。やがてその階でエレベータは停止するが、誰も降りず、誰も乗り込まない。乗り合わせた人々の視線を感じて、何やら罪を犯したかのようないたたまれない気になる。「押し間違えた階数ボタンをクリアする undo 機能がついていたら…」と恨めしく思う。このような経験をお持ちの方もおられるのではないだろうか。

われわれのまわりは、便利になった様々な機械がある。実際に使っている（あるいは使いこなせる）人がいるのだろうかと思われるほどの多種多様の機能を誇るものもある。リモコン操作ができるものも多い。活用するか死蔵するかは別として、多くの家庭には同じようなデザインのリモコンが2個や3個はある。

しかし、これらの機器のなかには、ユーザーの立場に立って設計したのだろうかと考えてしまうものもある。筆者の家のエアコンは「リモコンで操作できる」ようになっている。あるときリモコンの電池が用をなさなくなった。予備の電池がなかったの

でエアコン本体の強制スイッチを入れた。さて室温を設定しようとしたが、本体のどこを探しても室温調節のつまみがない。そのエアコンは「リモコンで操作しなければならない」ものだったのである。

人間はエラーを起こすが、絶対に故障しない機械も存在しない。機械に不都合が生じたなら、人間はそれを何らかの工夫で補おうとする。人間の工夫が入る余地を残さない設計は、よい設計と言えるだろうか。

われわれの社会には多くの自動化システムがある。技術の進展につれて機械の故障が少なくなり、人間のエラーが目立つようになる。いつエラーを起こすかわからない人間に代えて、その役割を自動化システムで代替しようとする動きは自然なものであった。

自動化には、人間が行うには危険で不向きな「極限作業」をロボットに代行させようという面もあるが、人間の関与を減らしてシステム全体の信頼性・効率を高めたいとするところに、システムの自動化への大きな動機づけがあった。

しかし、人間（オペレータ）はいまだにこれらのシステムの中に存在する。オペレータの存在が不可欠であるときえ言われている。なぜだろうか。そこには、システムの設計時の予想を越える状況が発生したときにはオペレータが対処してくれるとの期待がある。

必ずしも信頼性が高いとはいえない人間

を排除しようと導入された自動化システムだが、オペレータにシステム設計の不備を補うことを要求する。かくして、人間は自動化システムにとって欠くことのできない存在となる。これを「自動化の皮肉」¹⁾という。

自動化システムが人間の関与を必要とするなら、人間と自動化システムの調和のとれた関係をいかに実現するかが重要な課題となる。

本稿では、ヒューマン・エラーの様々な形態を人間の認知レベルとの関連から概観し、現在の自動化システムが抱える問題点を明らかにする。また、マン・マシン・システムをモデル化した監視制御 (supervisory control)²⁾ の概念を述べ、人間と自動化システムの協調と調和のありかたについて、自動化レベル (levels of automation) の観点から考察する。

2. 人間の行為とヒューマン・エラー

ヒューマン・エラーには様々な形態があり、その分類法も一通りではない。代表的なものに、(1) オミSSIONとコミSSIONへの分類を基本とする方式^{3),4)}、(2) スリップとミスエイクへの分類を基本とする方式^{4),5)} などがある。本稿では後者の分類法にしたがう。

スリップは、ほとんど定型化された動作をあまり意識せずに行ううちに、当初の意図とは異なったことをしてしまうエラーである。スリップにはいくつかの形態がある⁵⁾。以下に2通りの例を掲げる。

(例1) 筆者はある日、講義をすることになっている教室へ向かっているとき、学生に配付する資料のうちの1枚をコピーしてこなかったことに気がついた。急いでコピー機のある事務室へ戻ろうとエレベータ

に乗り、階数ボタンを押した。エレベータを降りて事務室へ入ろうとしたが、ドアは開かないし、廊下の様子も何やら違う。「おかしい」と思いつつもドアを開けようとしているうちに、そこは事務室のある9階ではなく、8階であることに気がついた。エレベータに乗ったとき、時間に気をとられていたため、つい自分の部屋のある8階のボタンを押していたのであった。

例1のエラーは、「乗っ取り型エラー」とよばれ、本来意図していた行為が日常的に頻繁に実行される行為によって置換されてしまうものである。

(例2) 1992年1月、エアバス A320機がストラスブル空港16km手前の山に墜落する事故があった。A320機は統合型デジタル電子技術に基づくグラス・コックピット機である。事故後の調査では、パイロットが、空港への進入角度を指示する「降下角度」と、1分間の降下高度を指示する「降下率」を間違えて入力したのではないかと疑いがもたれている。

A320機では、降下角度と降下率は、モードをボタンで切り替えて同じ装置から入力する設計になっている。パイロットは降下角度のつもりで「3.3 (度)」と入力したが、降下率入力モードになっていたため、コンピュータは「33 (00フィート/分)」と解釈したと考えられている⁶⁾。

例2のエラーは「モードエラー」とよばれ、いくつかの操作モードを持つ装置において、あるモードでの行為が、別のモードではまったく異なる意味として解釈される場合に起こる。

一方、ミスエイクとは、意図したとおりに行為が実行されるが、意図そのものが不適切であったり誤っているような場合のエラーをいう。すなわち、ミスエイクは思考

のエラーである。

ミステイクの典型例は、手許にある様々な情報に基づいてシステムの状態を判定しようとするとき、因果関係の解釈を誤ったり、本来考慮に入れなければならない要因のすべてを網羅できなかつたときに起こる。

(例3) スリーマイル島原子力発電所の事故において、加圧器水位計は「1次冷却水の水位が高い」ことを示し、他の計器は「炉心圧力は低い」ことを示したときがあった。ふつう、これらの現象は両立しない。加圧器水位が上昇を続けると、いずれソリッド(完全満水)になる。オペレータはソリッドを恐れ(そのように訓練されていた)、緊急炉心冷却装置への信号をバイパスさせるという、(結果から見れば)誤った措置をとった。実際は、加圧器逃がし弁の開固着のために1次冷却水は失われつつあり、加圧器逃がし弁の方へ向かう1次冷却水によって、水位が高く見せかけられたのであった。

オペレータは、加圧器逃がし弁が開いているとは考えもしなかつた。パネル上の表示ランプが「閉」を示していたからである。しかし、この「閉」表示は「弁を閉じるための信号が送られた」ことを示すものであり、「弁が閉じている」ことを示すものではなかつた。オペレータはそのようなインタフェース設計になっていることを知っていたが、緊張が高まった状況では心理的に視野が狭くなり、「弁が開いている可能性はないか」と考えてみることはできなかつた。

ミステイクの背景には、システムの複雑さ、マン・マシン・インタフェースの不適切な設計、不完全な訓練、社会的・時間的圧力など、様々な問題がある。

ところで Rasmussen^{8),9)} は、人間の行為をスキル・ベース、ルール・ベース、知識

ベースの3つの認知的階層に分けて捉らえることを提唱した。

スキル・ベースの行為とは、感覚系を通して情報が入力されたとき、ほとんど意識することなく円滑に行われる日常的あるいは熟練の域に達した行為である。

ルール・ベースの行為とは、「HならばC」の形態で記述されたプロダクション・ルールに基づくものをいう。ここでは眼前の状況がどのプロダクション・ルールの前件(H)に該当するかの判断が行われ、選ばれたルールの後件(C)が実行される。

馴染みがない状況に対しては、過去の経験やノウハウに基づく適切なプロダクション・ルールが存在しない。そこで仮説を立て、現在の状況が説明できるかどうかを検証する。仮説が妥当でないと判断されれば別の仮説を立てる。こうして現在の状況が同定されたなら、それに基づいて何をどのようにすべきかを決定する。このような、対象に関する概念的・構造的モデルに基づく行為を知識ベースの行為とよぶ。

ルール・ベースと知識ベースの行為には「思考」が関与するが、スキル・ベースの行為にはその直接的関与は見られない。このことから、行為に関する誤りとしてのヒューマン・エラーについても、思考が関与しているかどうかを区別して議論することができる。

Reason⁴⁾ は、ヒューマン・エラーを単にスリップとミステイクに2分するのは不十分とし、思考の直接的関与がないスキル・ベース・スリップ、思考が関与するルール・ベース・ミステイクと知識ベース・ミステイクの3種類に分類することを提案している。

先の例1や例2はスキル・ベース・スリップである。例3にはルール・ベース・ミ

ステイクと知識ベース・ミステイクが混在している。

3. 技術中心の自動化と人間中心の自動化

現在の多くのシステムは高度に自動化されている。しかし、システムの中から人間がいなくなったわけではない。オペレータは、自動化システムが目的どおりに正しく作動しているかを常に監視し、異常が発見された場合には適切な措置を施すうえで重要な存在である。すなわち、オペレータはシステム設計者の補完的任務を負う¹⁾。

複雑なシステムであればあるほど、その設計には多くの困難が伴う。設計者は予めシステムが運用される間に起こりうる様々な状況を想定し、そのすべてに対処できるように設計しなければならない。しかし、起こりうるすべての状況を予想することは一般に不可能であり、実際、「予期しなかった」状況も発生する。必然的に、設計者の予想を越えた状況に対処する役割を果たすものとして、オペレータには重大な任務が与えられる。設計段階では予想できなかった状況においてオペレータがシステムを救った例は、原子力プラント⁴⁾や航空機¹⁰⁾などでも報告されている。

こうして、人間の関与を排除しようとしつつも、本質的な点で人間の存在を仮定しなければならない、「自動化の皮肉」が現出する。いまや「自動化するか、否か」を議論してもそれほど有益ではない。「いかに自動化するか」を考察することが課題となる。

人間と自動化システムが共存するシステムを構成するにあたって重要となるのは、人間と自動化システムとの関係をどのように捉らえるかである。すなわち、「人間と自動化システムを同等なものとして位置づける」

か、「人間を主とし、自動化システムを従と位置づける」かによって、様相は大きく異なる。

人間と自動化システムを同列に考えると、「自動化できるところは自動化し、どのように自動化すればよいか明確でないところ、あるいは自動化のコストが高すぎるところは人間に任せる」ことになる。これを「技術中心の自動化」(technology-centered automation)¹¹⁾という。

技術中心の自動化では、困難な仕事を人間に割り当てるにもかかわらず、人間の特性を考慮したシステム設計は十分にはなされない。すでに航空機、生産システム、原子力プラント、銀行情報システムなど様々な分野で、技術中心の自動化による人間と機械の不整合の問題が指摘されている¹¹⁾。

これに対し、人間を主、機械を従と規定すると、「自動化システムは、人間にとって透明度が高い(transparent)²¹⁾なもの、かつ使いやすいものでなければならない。また、人間が必要とする支援を提供できるものでなければならない」という、「人間中心の自動化」(human-centered automation)¹¹⁾の考え方に至る。その基本となる「人間中心の設計」(human-centered design)の概念と実際例については、本号 Rouse 氏の解説記事²²⁾を参照されたい。

4. 監視制御モデル

人間と自動化システムが共存するシステムは、図1に示す監視制御モデル^{2),12)}で記述できる。ここでは、コンピュータを2種類のコンピュータ、すなわち、ヒューマン・インタラクティブ・コンピュータ (HIC) とタスク・インタラクティブ・コンピュータ (TIC) に分け、各々の役割分担を明確にしている。

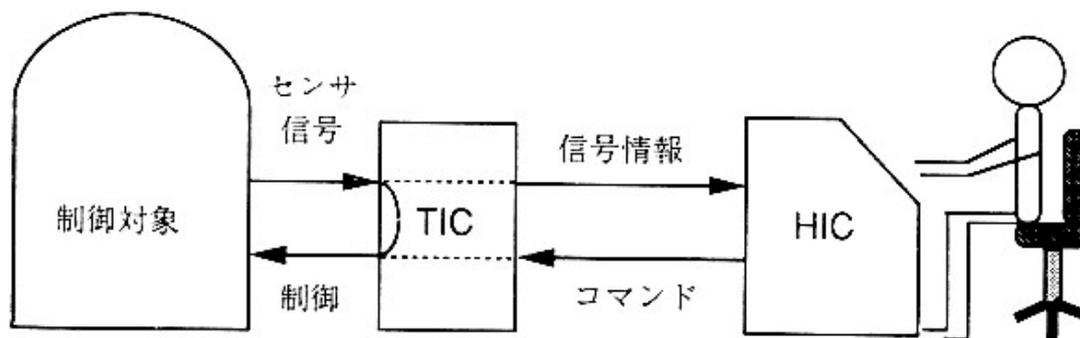


図1 監視制御系モデル

HICは中央制御室やコクピットなど、オペレータの身近にあり、(a) オペレータ・コマンドを解釈し、コマンド実行手順をTICに伝達、(b) TICからの制御対象状態に関する種々の信号を解釈し、オペレータへ情報提示、(c) オペレータ意思決定支援、などを司る。すなわち、HICは制御対象のモデル/知識ベースを持つ。

TICは一般に制御対象の近くにあり、HICから指示されたタスクの達成を目標に、対象への制御を行う。ここで、TICと制御対象の間にフィードバック・ループが構成されているところにポイントがある。

本来、監視制御モデルは、月面上の移動体を地球上から制御するテレオペレーションを想定して提唱されたが、生産プロセス、原子力プラント、航空機など、現在のマン・マシン・システムの多くはこのモデルで記述することができる。オペレータは制御対象から離れた中央制御室あるいはコクピットにおいて、自らの感覚器官を通して直接的に制御対象の状態を感知することはできず、ディスプレイや計器盤を通して、対象システムの状態を間接的に知ることになる。この状況は、テレオペレーションと本質的に変わらない。

監視制御系におけるオペレータの主なタ

スクは、つぎの5種類に分けられる。

- (1) 何をどのように行うべきかを定める。これを「計画」とよぶ。
- (2) 計画内容をHICに指示する。これは「教示」とよばれ、コマンドの送信あるいはプログラミングによって行われる。
- (3) 適切に、かつ予定どおりに自動化システムによる制御が行われているか、また、異常が発生していないかどうかを「監視」する。
- (4) 目標が達せられたとき、あるいは異常事態が検出されたとき、直ちに「介入」して自動制御モードを解除し、新たな目標を設定してHICに指示する。
- (5) 次回同じことをするのなら、どのようにするのがよいか、経験をもとに将来へ向けた「学習」を行う。

これら5種類の行為の時間的流れを図2に示す。

ループ(a)は、自動化システムの監視を続けているうちにある現象に気がついたとき、それが何によるものかを知ろうとして、改めて注意を向けて監視を続ける様子を表す。このループでは、注目すべき現象を見落とすスキル・ベース・スリップや、眼前の現象を誤った現象にマッチングさせるルール・ベース・ミスエイクなどが起こりうる。

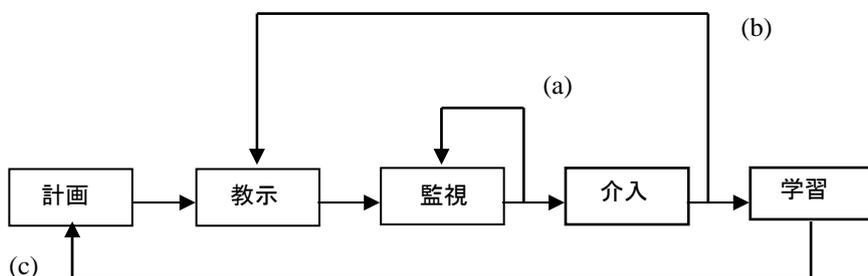


図2 オペレータの行為の多重ループ構造

ループ(b)は、何らかの異常を検知して自動制御モードを解除した後、どのように事態に対処すべきか新たな目標を自動化システム指示し、それが実行される過程を監視する様子を表す。ここで異常診断に誤りがあれば、ルール・ベース・ミステイクあるいは知識ベース・ミステイクが発生する。

ループ(c)は、経験から学んだことが次の計画立案に活かされる様子を表す。計画立案および学習は知識ベースの行為であることから、このループで発生するエラーは知識ベース・ミステイクが主となる。

5. 人間と機械の責任分担とその柔軟化

人間と自動化システム（特にコンピュータ）の役割分担の形態は、監視制御モデルにおける「自動化レベル」の観点から考察することができる。

自動化レベルとしては、ふつう、つぎの10通りのレベルが考えられている^{12),13)}。

- (1) コンピュータの支援なしにすべてを人間が行う。
- (2) コンピュータはすべての選択肢を提示する。人間はそのうちのひとつを選択して実行する。
- (3) コンピュータは可能な選択肢をすべて

人間に提示するとともに、その中のひとつを選んで提案する。それを実行するか否かは人間が決定する。

(4) コンピュータは可能な選択肢の中からひとつを選び、それを人間に提案する。それを実行するか否かは人間が決定する。

(5) コンピュータはひとつの案を人間に提示する。人間が了承すれば、コンピュータはそれを実行する。

(6) コンピュータはひとつの案を人間に提示する。人間が一定時間以内に実行中止を指令しない限り、コンピュータはその案を実行する。

(7) コンピュータがすべてを行い、何を実行したか人間に報告する。

(8) コンピュータがすべてを行う。人間に問われれば、何を実行したか人間に報告する。

(9) コンピュータがすべてを行う。何を実行したか人間に報告するのは、報告すべきであるとコンピュータが判断したときに限られる。

(10) コンピュータがすべてを行う。

上に示したリストは、考えられる自動化レベルをすべて網羅したものではない。現実には即した詳細な状況設定を行えば、上に示したもの以外の自動化レベルを見出すことも可能である。

現在の多くのシステムでは、「決定に関する権限・最終責任は人間にある」^{13),14)}とされる。これは人間中心の自動化システムにとっては大前提となり、人間と機械を同列に扱う技術中心の自動化システムにおいてすら、このことが仮定されている。

「決定の権限および最終責任を人間に与える」のは当然であり、コンピュータに主導権を与えたレベル(6)以上の自動化など考える必要もないと思われるかもしれない。しかし、ほんとうにそうだろうか。

既に見たように、人間は様々な状況のもとで様々なエラーを犯す。それらのうちには人間に責任を帰して良いエラーもあるが、「被告席に座るのはデザインである」⁵⁾との指摘もあるように、マン・マシン・インタフェースを含めたシステム設計に起因するエラーもある。時間的余裕がない場合にはエラーの発生傾向はさらに助長される。そこで、緊急度の高さなど、状況によって自動化レベルを柔軟に切り換えることができる設計が考えられないだろうか。筆者らは、システム安全性の観点から、状況に応じて自動化レベルを切り換えることの有効性を簡単な数学モデルを用いて証明したが¹⁵⁾、実際のシステムにもその例を見ることができる。

運輸省や自動車メーカーが21世紀初頭の実用化を目指している「先進安全自動車」では、衝突の危険が予測されるにもかかわらずドライバーの適切な操作がない場合は、自動的に車を止めたり最適な回避手段をとるシステムの開発を目標にしている¹⁶⁾。この衝突回避装置の自動化レベルは(7)である。

同様の例は航空機にも見られる。周知のように、ウィンドシアは航空機の安全運航に危害を及ぼす。ウィンドシアが検出された

ときは、音声警告を発してパイロットに知らせる設計(自動化レベル(4))になっているものもあるが、グラス・コクピット機のひとつであるマグダネル・ダグラスMD11は、ウィンドシアが検出されたときはコンピュータが自動的に危険回避コースをとるように設計されている¹²⁾。これは自動化レベルとしては(7)に相当する。

このように危険回避操作がルール・ベース的行為として記述できるものは、必ずしも「それを実行する時点」においてコンピュータは人間の了承を求めなくてもよいのではないだろうか。必要がある場合に対してのこのようなコンピュータの行為は「事前に」・許可できるものであり、人間中心の自動化には矛盾しない。

予期しない状況はまれにしか起こらないが、いったんそれが起こると、人間のワークロードは急激に増加し、認知レベルも、それまでのスキル・ベースあるいはルール・ベースの認知レベルから、より高度かつ抽象的な知識レベル³⁾と移行しなければならない。しかし、それを迅速・円滑に行うことは容易ではない。

このようなとき、自動化システムは単に「警報を出して事足れり」とするのではなく、困難な状況に追い込まれている人間の肩代わりをする行動をとってもよいのではないだろうか。それができてはじめて人間と自動化システムの調和と協調が実現されるように筆者には思われる。

6. あとがき

現在の自動化は「完全な自動化に見せかけた不完全な自動化」¹¹⁾である。事故が起これば、「状況判断を誤り初歩的なミスを行った」オペレータはその責を問われる。しかし、「オペレータの判断に誤りがあった」

と言えるのは、われわれがすべての経過と結果を知っているからであり、オペレータの「誤判断」の中には、彼らが置かれた状況では「合理的」と考えざるを得ないものも多い。

「決定に関する権限・最終責任は人間にある」とされるが、オペレータの責任を問うに相応しい環境はまだ整えられていない。人間の特性に合致したシステム設計やマン・マシン・インタフェース¹⁷⁾、オペレータ支援技術、人間と機械の役割分担の適正化^{18),19),20)}などについて、さらに研究を積み重ねる必要がある。

本稿では、「決定に関する権限・最終責任は人間にある」ことを前提としつつもその形式的解釈にこだわらず、自動化レベルを状況に応じて切替えることの可能性を考察したが、レベル(6)以上の自動化の実現には、センサシステムの高信頼性が前提となる。また、「オペレータのエラーをコンピュータが検出・補償できるか」という問題にも関連する。人間と機械の調和と協調を目指すには、まだ多くの課題が山積しているようである。

参 考 文 献

- 1) L. Bainbridge: Ironies of Automation. In J. Rasmussen et al (eds.), *New Technology and Human Error*, Wiley, 271/283 (1987)
- 2) T. Sheridan: Supervisory Control. In G. Salvendy (ed.), *Handbook of Human Factors*, Wiley, 1243/1268 (1987)
- 3) 幸田, 井上: ヒューマン・エラーの評価方法, 計測と制御, 30-7, 623/630 (1991)
- 4) J. Reason: *Human Error*, Cambridge University Press (1990)
- 5) D.A. Norman: *The Psychology of Everyday Things*, Basic Books (1988) あるいは(野島久雄訳) 誰のためのデザイン?, 新曜社(1990)
- 6) 赤木昭夫: 安全の鍵をにぎるインタフェースと人間, 予防時報 170, 5 (1992)
- 7) 柳田邦男: 恐怖の2時間18分, 文芸春秋 (1983)
- 8) J. Rasmussen: *Skills, Rules and Knowledge: Signals, Signs and Symbols and Other Distinctions in Human Performance Models*, IEEE Trans. SMC, 13-3, 257/267 (1983)
- 9) J. Rasmussen: *Information Processing and Human-Machine Interaction*, North-Holland (1986)
- 10) 加藤寛一郎: 墜落, 講談社 (1990)
- 11) D. Woods: *The Effects of Automation on Human's Role: Experience from Non-Aviation Industries*. In S.D. Norman, H.W. Orlady (eds.), *Flight Deck Automation: Promises and Realities*, NASA CP-10036, 61/85 (1989)
- 12) T.B. Sheridan: *Telerobotics, Automation, and Human Supervisory Control*, MIT Press (1992)
- 13) S.K. Card: *Human Factors and Artificial Intelligence*. In Hancock and Chignell (eds.), *Intelligent Interfaces*, North-Holland, 27/48 (1989) あるいは(認知科学研究会訳) 知的インタフェース, 海文堂 (1991)
- 14) G. Johannsen: *Fault Management, Knowledge Support, and Responsibility in Man-Machine Systems*. In Wise and Debons (eds.), *Information Systems: Failure Analysis*, Springer-Verlag, 205/209 (1987)

- 15) T. Inagaki, G. Johannsen: Human-Computer Interaction and Cooperation for Supervisory Control of Large-Complex Systems. In Pichler and Moreno Diaz (eds.), Computer Aided System Theory - EUROCAST'91 (Lecture Notes in Computer Science 585), Springer-Verlag, 281/294 (1992)
- 16) 「21世紀の安全車」, 6月30日朝日新聞 (1992)
- 17) K.J. Vicente, J. Rasmussen: Ecological Interface Design: Theoretical Foundations, IEEE Trans. SMC, 22-4, 589/606 (1992)
- 18) Balancing Automation and Human Action in Nuclear Power Plants, IAEA (1991)
- 19) G. Johannsen: Towards a New Quality of Automation in Complex Man-Machine Systems, Automatica, 28-2, 355/375 (1992)
- 20) A.L. Levis, N. Moray: Task Allocation Problems and Discrete Event Systems, Preprints IFAC Symposium on Analysis, Design and Evaluation of Man-Machine Systems (1992)
- 21) 海保, 原田, 黒須: 認知的インタフェース, 新曜社 (1991)
- 22) W.B. Rouse: Human-Centered Design: Concept and Methodology, 計測と制御, 32-3 (1993)