BUTTERWORTH HEINEMANN

Computer Integrated Manufacturing Systems Vol. 8, No. 2, pp. 133–142, 1995

0951-5240/95 $10.00 + 0.00

0951-5240(95)00007-0

Adaptive human-computer interfaces for man-machine interaction in computer-integrated systems

Lajos Bálint

Department of Natural Sciences, Hungarian Academy of Sciences, Budapest, Nádor-u.7., H-1051 Hungary (email: h48bal@huella.bitnet)

Human-centred computer-integrated systems are investigated in view of their man-machine interaction requirements. It is shown that adaptive interfaces satisfy all the demands stemming from the requirement of human-centredness. A brief description is given of what kinds of adaptation can be built into the interfaces allowing human-computer interaction in the related computer-integrated systems. Finally, some basic criteria of how to construct efficient, reliable and user-friendly adaptive interfaces, are investigated. After introducing computer-integrated systems and associated human-computer interaction, a general view of the human-computer interfacing problem is given, followed by sections on interface construction, human-centred interfaces, adaptivity/adaptability/adaptation, taxonomy of adaptive interfaces and interface realizations, construction of adaptive interfaces and some trends in applying such interfaces in human-centred computer-integrated systems, especially computer-integrated manufacturing systems.

Keywords: man-machine systems, human-computer interaction, adaptive interfaces, human-centredness, computer-integrated systems

Introduction

Computer-integrated systems (CIS, i.e. special man-machine complexes where human-computer co-operation is the basis of advanced system functionality and computer-based integration of the system functions is determining the key but 'built-in' role of the humans) assume intensive human-computer interaction[1,2].

However, realizing efficient, reliable and cost-effective human-computer interaction is not a trivial technical task, because of the many discrepancies between machine and human properties, and because of the wide variety of characteristics belonging to the machine and human parties of the related man-machine systems (MMS)[3,4]. Even simple cases, 'systems' consisting of a single human (operator or user) and a single computer (PC or terminal) only pose difficult design problems[4,5]. Complex computer-integrated systems, comprising many humans and many machines (computers), are far more complicated and require much more sophisticated interaction sessions.

Among the above-mentioned computer-integrated systems, computer-integrated manufacturing systems (CIMS) are targets of one of the topics in the research devoted to the important theory and practice of systems development and operation[6]. CIMs keep all the general features of computer-integrated systems, but additional specific properties and requirements stem from the tight coupling between machine and human during system operation, and from the severe criteria regarding accuracy, real-time functional demands and strict operational reliability. However, throughout the text below, CIS will be mentioned and investigated, in a general context, even if some of the statements refer first to CIMS or computer-integrated manufacturing/production systems.

The problem of CIS realization became yet more difficult with the introduction of the 'human-centredness' aspect of the design and operation of computer-integrated systems[7,8]. Because of the related new requirements, advanced interfaces should not only facilitate adequate information transfer between humans and machines, but special demands also stem from the characteristics of the human collaborators. These special demands comprise, among others, taking into account the aspects of acceptance, involvement, satisfaction, convenience, comfort, stress and fatigue[9].

The key component in the interaction processes is the human-computer interface (HCI), or simply 'interface' (IF). This is the special component in the man-machine system which enables humans and machines to exchange information[1,2]. Its basic function is to create a bridge between humans (as information sources and receivers) and machines (as information drains and transmitters, respectively). These interfaces are also responsible for the above human-centredness.

Human centred computer-integrated systems prohibit the application of traditional (firm, static, definite, inflexible) interfaces, because these IF types do not allow even the simplest matching between machine and specific human characteristics, nor do they make possible the adjustment of the IF properties to changing circumstances (system state, human disposition, etc.).

The right IF types for interactions in human centred computer-integrated systems are equipped with adaptive properties[10,11]. *Adaptive interfaces* can be well matched to the different and/or changing requirements, or they automatically adapt themselves to the different and/or changing circumstances. However, crucial questions promptly arise what kinds of adaptivity and what methods of adaption are to be applied? These questions are investigated briefly, but in a relatively broad context, in the following.

The structure of the contribution is straightforward in the sense that the forthcoming text starts with a more general study of the related interaction problems, and goes on to elaborate on the more specific questions of adaptive interfaces in CIS and CIMS.

First, human-computer interaction is investigated as the basic cooperative activity between man and machine. Next, a general view of the human-computer interfacing problem is outlined. This part of the paper is followed by sections on interface construction, human centred interfaces and adaptivity/adaptability/adaptation. Then, there follows a taxonomy of adaptive interfaces and, after an evaluation of their possible realizations with respect to adaptivity needs, there comes a section on how to construct such adaptive interfaces. The contribution concludes with a brief look at what is probably the near future of adaptive human-computer interfaces in human-centred computer-integrated systems.

The terms RISC and ASIC, as used in this paper, must not be confused with identical acronyms used for particular hardware components.

Interfaces in human-computer interaction

Human-computer interfaces are special devices in man-machine (human-computer) systems. They serve as a bridge between man and machine (between human and computer) and, by their basic function, they allow *information exchange* between human and machine participants of the MMS operation.

This section summarizes some elementary aspects of how the above basic function is to be dealt with, and how it can be regarded when interfaces are to be analysed and realized. Machine and human properties, interaction criteria, as well as interaction modes and media, are briefly examined by keeping in mind the need for some guiding principles of how to handle the interfacing problem.

When investigating human-computer interfaces, the following starting points are to be taken into account with respect to machines and humans ('components' or 'participants') as *transmitters*, *receivers* and *processors of information* in a man-machine system:

- the *machine* (the computer) is designed by intention so that it is deterministically performing well-defined operations, and interaction is served by proper hardware components (keyboard, mouse, visual display unit, etc.), and by specific software components, as well as databases/knowledge-bases,
- the *human* is behaving, by its nature, non-deterministically, based on such properties as:
  - physiological attributes (eyes, ears, fingers, etc.),
  - intellectual characteristics (capacity, recognition, learning, decision, etc.),
  - knowledge basis (knowing the environment, the system, himself/herself, etc.), and
  - psychological states (concentration, vigilance, fatigue, patience, etc.).

The interfaces should allow *adequate interaction* between the different participants of the man-machine system. Through the interaction processes, they should:

- facilitate satisfactory monitoring of machines by humans,
- support human intervention in machine operations,
- aid human decision-making by providing, e.g. system state diagnosis and intervention possibilities, and thus
- establish error-free or error-tolerating operation of the full system, and in this way
- guarantee efficient and reliable system performance.

To formulate how these requirements can be fulfilled, we have to make a distinction between the different possibilities of what *content and form* the communicated information may have.

With respect to the content of the transferred information (associated to the *mode* of interfacing), the following three of the many aspects have particular importance:

- What kind of information is transferred with respect to the assumed knowledge of the receiver about the MMS operation (i.e. what is transmitted and what isn't transmitted by the human to the machine and by the machine to the human)?
- What kind of abstraction is used in shaping the transferred information (i.e. how familiar is the human with the system model and system functions)?
- What kind of symbolic and physical coding is applied to the data belonging to the transferred information (determining, e.g., what kinds of information losses or errors might be caused/prevented by the coding in the information transfer)?

Concerning the form of information transfer (being in close relationship with the *medium* of the communication through the HCI), again, three important aspects should be emphasized:

- What kind of information transfer is used towards the human side of the IF (i.e. what generic form (numeric, alpha-numeric, textual, graphic, pictorial, audio, speech, tactile has the transferred information?
- What parameters (speed, intensity, extent, variety, etc.) belong to the above forms of the informational transfer?
- What frame or embedding exists for the above information transfer, i.e. what underlying background is provided by the system (especially by the HCI) for making the information transfer efficient and reliable?

It should be mentioned that advanced interfaces apply *multimode/multimedia* communication for facilitating man-machine interaction. However, the many different interface solutions may be characterized by different techniques underlying the satisfaction of general and specific requirements.

Classification of human-computer interfaces

As far as classification of human-computer interfaces is concerned, there are many ways in which to categorize them. In the following, a classification scheme is outlined by which MMS system tasks, system structure, operation principles, environment, kinds of machines and kinds of human are all considered as classification aspects. In this way, we arrive at a state where adaptivity of the interfaces themselves may be selected as the next aspect for further investigation into related IF solutions.

The structure of this section is in accordance with the above goal. First, it goes into some detail concerning the interface essentials; then it surveys the man-machine systems and the interaction process, with respect to system functions, intervention types, human and/or machine dominance, time-related conditions, importance of adaptivity and the role of personal shaping factors. Finally, characteristic application areas are classified in view of the concise survey given.

Interface functions, interface properties and interface realizations

The interface functions, the requirements against the interface properties and the different interface realizations may be investigated in view of:

- the rough classifications of MMS complexes, i.e.:
  - system functions
  - interaction types
  - interface orientations
  - task requirements
  - application demands and
  - personal shaping factors;
- the main application areas (task domains); and
- the cross reference between MMS classes and application areas.

(The latter will also serve as a straightforward basis for investigating the construction of human-computer interfaces.)

Man-machine systems and HCI

Man-machine systems may be categorized, as a first step, by considering the system *function* itself. Here, three main categories are to be distinguished:

1. Machines might do the job without human involvement, but feasibility is questionable.
2. Humans, might do the job without machines, but efficiency/reliability is questionable.
3. Man-machine cooperation is a must (no pure machine- or human-based execution is possible).

As far as man-machine cooperation is concerned, the different situations are characterized, with respect to the types of the *interventions*, by (at least dominantly) either:

1. event-driven, or
2. action-driven

intervention occurrences. Here, another, but in principle, similar distinction can also be applied, based on whether:

1. the machine, or
2. the human is (dominantly) supervising the processes of interactions. Of course,
3. a mix of the two extremes is also possible (mix of event- and action-driven operation, i.e. mixed or alternating supervision).

With respect to the *orientation* of the interface, the IF may be:

1. either process-oriented (where communication is the only attribute being adjusted to user requirements and capabilities), or
2. user-oriented (where the basic system processes themelves are designed directly by taking into account human performances and preferences), again, allowing,
3. the mix of the two types, but postulating at least dominance of either orientation.

Regarding the related *requirements*, the basic system function (process) is either

1. timing sensitive (i.e. sensitive to timeliness of the interventions), or
2. tolerant with respect to the speed of interactions.

while, on the other hand, extreme reliability (virtually error-free operation is either

(A) a must, or
(B) simply an advantage.

Depending on the *demand* posed by the properties of the application, adaptivity is either:

1. a total necessity, or
2. merely a potential convenience factor in system operation.

Among personal shaping factors (PSF, determining/influencing where, when and how adaptivity is to act with respect to the humans involved), a rough classification may distinguish between:

1. Internal factors, i.e. human characteristics: properties (physical, mental and psychological attributes) as well as knowledge, skill, intentions, motivations, etc.
2. External factors, including general (environmental, organizational and task-relational) circumstances and specific settings (related to task-specific demands as well as support).
3. Stressors, that is, physical, physiological, and mental/psychological effects on the human, being in relation to his or her task and role in the MMS; first of all anomalies concerning the circumstances, the task and the interface, itself.

Characteristic application areas

When making a distinction among different situations and different cases (system functions, intervention types, interface orientations, task requirements, adaptivity demand and personal shaping factors), the different *application areas* (task domains) covered by the MMS are of key importance.

Let us limit our investigation regarding these areas to the separation of only four fundamental classes, depending on whether a *computer-aided human activity* is under consideration (simply a recreational one or, more demandingly, a creative one), or there is a *man-machine cooperation*-based complex system process (simply by built-in computers or, more sophisticatedly, by computer-integration). The four classes are:

- CHE: computer-based human entertainment, e.g. computer video games, multimedia performances, etc.
- CHA: computer-aided creative human activities, e.g. computer-aided design, computer-aided instruction, etc.
- COA: computerized office administration, e.g. collective text processing, database manipulations, etc.
- CIS: computer-integrated systems, e.g. computer-integrated industrial (manufacturing/production) systems, etc.

Although the above four classes do not constitute an exhaustive classification of the widest variety of application areas, they form a useful set of different cases allowing categorization by *different properties*, so that almost any specific application can be associated to one of the four classes by taking into account the specific properties of the very application. Thus, if one wants to determine what interface construction to use (and especially, what adaptivity to build in), this categorization serves as a good basis for making the appropriate decisions.

About interface construction

This section first outlines how the interface construction problem is to be dealt with in view of system requirements and application categories. Next, some basic principles of interface construction are given, including specification, modelling and design. Special attention is paid to conceptual, functional and physical aspects, to hierachy and modularity, to a reduction of specification demands, and to re-usability of the interface components.

Constructing human-computer interfaces is just one specific area of constructing technical products. Thus, the general aspects of a 'good product' are applicable here too, namely *efficiency* of the operation, *reliability* of the device and *cost* of producing and using the interface[12].

However, because of the many specific requirements regarding human-computer interaction in man-machine systems, we introduce the above-mentioned cross-reference between MMS classes and application areas so that the derivation of the specific construction (design and realization) aspects is well assisted.

As we have seen in the previous section, ideal human-computer interfaces are capable of aiding man-machine interaction under very *complex circumstances* belonging to general and specific effects (PSF). This capability can be achieved only if:

(a) optimum control of the general external factors is established by (possibly automatic) controlling functions built into the interface;
(b) minimum requirements against human properties are posed by the specific external factors belonging to the specific interface functions;
(c) optimum (and not always minimum) stressors are continuously produced by the interface for continuously obtaining an optimum stress state of the humans; and
(d) the effect of the internal factors is eliminated (or at least minimized) by the interface for achieving reliable and error-free system operation independently of possible human uncertainties.

Interface construction vs. application and system categories

Table 1 summarizes how the four groups of MMS application areas are generally related to our earlier categorization of system functions, intervention types, interface orientations, task requirements, adaptivity demand and personal shaping factors (number codes refer to the codes introduced in the sub-section on 'Man-machine systems and human-computer interaction').

Although, in accordance with *Table 1*, the aspects of constructing interfaces vary in a wide spectrum, the basic principles are of a general nature. Here, a well applicable background of IF construction is provided by the *abstraction hierarchy* used in modelling human-computer interaction, and by the *formal models* applied in interface design. They may take care of the hierarchy

Table 1 MMS application areas vs. system characteristics

	CHA	CHE	COA	CIS
System functions	3	2	2	1
Intervention types	2	3	3	1
Interface orientations	2	3	2	1
Task requirements	2A	1B	2B	1A
Adaptivity demand	2	2	1	1
Personal shaping factors	1	2, 3	1, 2	1–3

levels, the interaction *modes and media*, the *modular structure* of the interfaces, and also the integrated consideration of the *conceptual-functional-physical* aspects of *interface-construction*.

Principles of interface construction

HCI *specification and design* (together with the associated modelling principles) should aim at an integral consideration of the many conceptual, functional, structural and physical aspects. Thus, an HCI specification consists of conceptual, functional and physical criteria[13].

Conceptual specification works with a formalism of events and their relations or consequences (event-oriented specification).

Functional specification of a HCI works with information (messages) coming from the machine to the human and from the human to the machine. These messages may belong to actions, reactions, feedback etc. The specification of the HCI is prescribing the appropriate relations between the messages/message types. These relations are associated with functions while the interface functions are associated with components of the interface. The behaviour of these components is described by their internal states. The components and states are described at this stage by their functional characteristics.

The *physical specifications* of the interface prescribe the static and dynamic properties of the interface in association with the interface functions (i.e. they prescribe where the different functions are to take place spatially, and how the different messages are to be handled in time).

During HCI design, the conceptual specifications are transformed into functional ones, and the functional specifications into physical ones. Both steps work by utilizing the structural properties of the interface, namely vertical *hierarchy* (or abstraction levels) and horizontal *modularity* (in accordance with the different modules belonging to different modes and media).

Abstract formal modelling of the interfaces help in a systematic integral consideration of the conceptual, functional, physical and structural aspects. The model works with sets of events, actions, reactions, feedback, components, states, physical parameter vectors, etc. by introducing special algebraic structures and operator structures, and by using algebraic operations, transformations and mappings, working on the elements of the different sets.

HCI design goes on by performing *conceptual, functional* and *physical design* procedures (through direct or inverse application of the relevant transformations and mappings, generally in an iterative-interactive way, and by using appropriate databases and knowledge-bases). However, the complexity of the interfaces requires special techniques if design procedures are to work well[14].

Functional complexity can be decreased, on the one hand, by using restrictions in the applied (allowed) hierarchical levels, modes and media, as well as in the variety of events, circumstances, etc. to be handled by the interface. Such limitations in the construction of the interfaces result in *reduced interaction scheme construction* (RISC interfaces or interface components).

On the other hand, as far as the HCI modules are concerned, we should distinguish between *universal interface components* (UNIC) and *application-specific interface components* (ASIC). UNICs are characterized by only general intelligence belonging to standard general MMS functions, while ASICs contain additional specific intelligence, especially with regard to a specific MMS application. Thus, the standard module bank used by the construction process contains a set of UNIC modules (they determine the class of reusable interface functions and models) and, during the construction process, a set of application-specific (ASIC) components is generated for the specific tasks belonging to the specific MMS application.

Since universal modules are inexpensive and can be used in different tasks, but are suboptimum solutions in any specific tasks, while application-specific modules are optimum solutions in specific cases but cannot be re-used in further applications, and thus are very expensive, a third class of modules is introduced. The modules in this class are called *application-oriented interface components* (AOIC). They combine the benefits of both the UNIC and ASIC modules but without their drawbacks. They have general intelligence and a certain amount of specific intelligence. But here, the specific intelligence concerns not just a specific application, but a full specific class of applications, i.e. a specific application area. Such AOICs in the module bank allow us to use a radically reduced number of ASIC modules while keeping the possibility of constructing optimum HCIs with reduced difficulties and reduced costs.

It is to be noted that UNICs are application-independent and easily programmable, while ASICs require extensive sophisticated programming which is not re-usable. AOICs involve a high amount of software, too, but here the result of programming is appropriate not just in a specific MMS solution, but in a wider class of applications (*re-usability*). Such an AOIC is to be further programmed to obtain an application-specific solution, but this additional programming phase is much simpler than the development of ASIC software (thus, although this additional software is not re-usable, it is relatively inexpensive).

By using RISC and UNIC/ASIC/AOIC methods and principles, interface construction may become an efficient and straightforward process.

Human-centred interfaces in computer-integrated systems

The basic function of the interfaces in computer-integrated systems is to facilitate information transfer between the 'integrating' computers and the participating humans (the latter being normally simple information sources and sinks, and not being the supervisors of the integrated system functions). Thus, quite evidently, human factors are in most cases almost entirely out of consideration during the traditional design process of interfaces (and similarly, of the whole CIS). This is because, with the advent of computer integration of the complex system functions (first in sophisticated manufacturing and production systems), the belief in the overall 'miracle' stemming from introducing computer integration overshadowed the key role of the humans, and there was little care taken in human factors, apart from consideration of some comfort and ergonomic aspects.

Adequate and reliable human-computer interaction is extremely important in computer-integrated manufacturing systems. However, human-computer interfaces play a crucial role in the operation of these systems, not only because of their importance in assuring adequacy and reliability, but also in providing human comfort, an optimum stress state and job satisfaction.

This is because with the advancement of computer integration of man-machine systems, it turned out that the underestimated role of humans should be revised: it became quite clear that *human factors* and *human aspects* also play a crucial role in adequate, efficient and reliable system functioning. Moreover, even operation costs depend upon how these human aspects are taken into account in system conceptualization and design. Such recognitions led to the principle of human-centredness, and that is why human centred systems design came into the focal point of constructing computer-integrated systems.

With the evolution of human centredness in CIS (and especially CIMS) design, the emphasis shifted from purely functional aspects to a combination of functional requirements and human aspects. This shift did not leave the design of human-computer interfaces unchanged. The answer of HCI theory and practice was the introduction of *adaptivity* into the interface operation.

Adaptation, adaptability, adaptivity

This section is devoted to the brief characterization of the adaptivity concept as applied to human-computer interfaces in computer-integrated systems. First, the basic tasks of adaptive interfaces are summarized, then adaptivity as a specific property is analysed in some detail, together with a short explanation of why a taxonomy should be introduced, before going into details on how to realize adaptive IFs.

Adaptivity, as an interface property, comprises a wide spectrum of characteristics. These involve, among others, the capabilities of the interface to:

- adjust the forms of information transfer,
- transform the information content,
- alter/merge modes of information flow, and
- exchange/combine communication media.

On the other hand, it should be borne in mind that the word 'adaptivity' covers several different things. Among these we can make a distinction between the terms adaptability, adaptivity, and (self-) adaptation:

- *Adaptability* is an interface property, meaning that the interface can be adjusted to certain circumstances or some of its parameters can be tailored to certain prescribed specifications.
- *Adaptivity* is a property expressing that the interface can itself convert or modify itself, with the aim of getting matched to changing circumstances, requirements or needs.
- *Adapting* means the process of utilizing adaptability through forcing the interface to vary (i.e. to perform the adjusting or tailoring steps mentioned) by taking into account prescribed rules and guidelines.
- *Adaptation* (self-adaptation) is an intrinsic process going on inside the interface automatically, resulting in step-by-step or continuous conversion or modification of the interface properties and operation.

The above distinction between the different methods of how to achieve human centred interface operation allows the introduction of a quasi-exhausting taxonomy of the (outline) adaptive interfaces. However, before getting into detail with respect to the taxonomy mentioned, some introductory words are still necessary.

It is an important aspect that, in general, in introducing adaptivity into a human-computer interaction scheme, the normal method of *information exchange* (based on demand and supply of information by both parties, human and machine) is altered: the rules of the original push-pull (demand/supply) mechanism of information transfer become *flexible* relative to the original rigidity. The most important parameters of an adaptive interface declare what amount of flexibility is provided concerning the content and form of the bidirectional information transfer.

It should always be kept in mind as well that, as pointed out above, human-computer interfaces are to be investigated in a broad sense to get a complete picture of where, when and how it is possible/necessary to utilize adaptivity and what kinds of interface design techniques can be used to take into account the adaptivity aspects themselves.

We have seen that a classification of manned technical systems and human personal shaping factors (PSF) lead to taxonomies of human-computer interaction schemes. Consideration of the many different application tasks categories, system structure classes, operational solution principles, environmental aspects and participating 'collaborator' varieties (types of machines as well as kinds and nature of humans) allow correct examination of the separate targets to which the adaptivity aspects are to be applied. Benefits and costs, advantages and disadvantages, promises and pitfalls can also be evaluated by aiming at a general framework of how to determine which interface functions are to be equipped with adaptive capabilities, and of how to select the proper methods of adaptivity with respect to system design efforts and reliable/efficient system operation.

And now, with the aspects mentioned above in mind, a taxonomy of adaptive interfaces is outlined before evaluating the adaptivity realizations in view of adaptivity needs, and before summarizing some basic facts with respect to constructing adaptive interfaces.

Taxonomy of adaptive interfaces

On the basis of the previous sections, an application-oriented taxonomy of adaptivity types is introduced in the following. This taxonomy will make it possible to investigate a cross-reference between different adaptivity types and different MMS application areas, and by these means a set of basic points in constructing adaptive interfaces for human centred computer-integrated systems can be derived.

As far as adaptive behaviour is concerned, the classification of *Table 2* may be considered as a straightforward step towards the elaboration of the quasi-exhaustive taxonomy mentioned (the list of items in *Table 2*, although collected with the intention of achieving full coverage of the many possibilities, can not be considered as a complete set of adaptivity solutions; with the evolution of adaptive interfaces, the taxative register may become more extensive).

It should be mentioned that the different adaptivity types can be characterized and categorized with respect to practical applicability, re-usability, cost and demand for research. These aspects will be briefly examined in the following.

It is also to be noted that allowing *human control* even in the case of virtually fully automatic adaptation

Table 2 Classification of adaptivity types and adaptive interfaces

1. User adapts to interface
    1.1. Adaptation without human-computer cooperation (interface doesn't help user adaptation)
        1.1.1. Advance user adaptation
            1.1.1.1. Simulator-based training
            1.1.1.2. On-system training
        1.1.2. Real-time user adaptation
            1.1.2.1. Out-of-process training (by-process testbed)
            1.1.2.2. On-process training (operation on MMS)
    1.2. Adaptation supported by human-computer cooperation (interface helps user adaptation)
        1.2.1. Advance user adaptation
            1.2.1.1. Teaching-based (interface teaches user)
            1.2.1.2. Training-based (interface trains user)
        1.2.2. Real-time user adaptation
            1.2.2.1. Interface examines/advises user
            1.2.2.2. Interface psychologically prepares (tunes) user
2. Interface adapts to user
    2.1. Adaptation without human-computer cooperation (user not involved directly in interface adaptation)
        2.1.1. Advance interface adaptation
            2.1.1.1. Conversation-based adaptation (controlled by machine)
            2.1.1.2. Testing-based adaptation (controlled by machine)
        2.1.2. Real-time interface adaptation
            2.1.2.1. Out-of-process interface adaptation (as in 2.1.1. but repetitively during process)
                2.1.2.1.1. Conversation-based adaptation (controlled by machine)
                2.1.2.1.2. Testing-based adaptation (controlled by machine)
            2.1.2.2. On-process interface adaptation (continuous automatic evaluation of user behaviour)
                2.1.2.2.1. Stress-based adaptation (stress state of user is evaluated)
                    2.1.2.2.1.1. Stress is limited
                    2.1.2.2.1.2. Stress is optimized
                2.1.2.2.2. Adequacy-based adaptation (adequacy of user interventions is evaluated)
                    2.1.2.2.2.1. User support is varied
                    2.1.2.2.2.2. User involvement is varied
    2.2. Adaptation supported by human-computer cooperation (user directly involved in interface adaptation)
        2.2.1. Advance interface adaptation
            2.2.1.1. Conversation-based adaptation (controlled by human)
            2.2.1.2. Testing-based adaptation (controlled by human)
        2.2.2. Real-time interface adaptation
            2.2.2.1. Out-of-process interface adaptation (as in 2.2.1. but repetitively during process)
                2.2.2.1.1. Conversation-based adaptation (controlled by human)
                2.2.2.1.2. Testing-based adaptation (controlled by human)
            2.2.2.2. On-process interface adaptation (repetitive human-aided evaluation of user behaviour)
                2.2.2.2.1. Stress-based adaptation (stress state of user is evaluated)
                    2.2.2.2.1.1. Stress is limited
                    2.2.2.2.1.2. Stress is optimized
                2.2.2.2.2. Adequacy-based adaptation (adequacy of user interventions is evaluated)
                    2.2.2.2.2.1. User support is varied
                    2.2.2.2.2.2. User involvement is varied

is extremely important[1,5]. Possibilities are, for instance, informing the human, confirming by the human, human-made decisions regarding machine-provided advice, human selection from among machine-provided alternatives, etc.

Evaluation of adaptivity needs vs. adaptive interface realizations

The classification of *Table 2* may be considered as a list of different solutions for realizing interaction schemes and interfaces involving some kinds of adaptivity. Each solution can be characterized by its respective difficulty of construction and cost of implementation. It is to be noted that a pure application of only one solution rarely occurs. Thus, the complexity and cost of a specific solution for built-in adaptivity in a man-machine system is always a function of the applied elementary procedures for adaptive operation and of the method by which these procedures are combined.

The different solutions belonging to the separate classes in the classification mean different complexities of design, realization, installation, training, maintenance, modification, upgrade, etc.

The nature of adaptivity methods and tools also varies together with complexity. Specific (i.e. application-oriented) nature is decreasing, while general nature (or application independence) is increasing, with getting ahead among the different complexity levels covered by the list. This effect is resulting in an economic balance with respect to cost aspects: although the cost of more advanced interfaces is relatively high, their re-usability is generally also better, thus overall average costs in case of multiple applications or a number of products may be kept within appropriate limits.

Construction of adaptive interfaces

The first part of this section on constructing adaptive interfaces examines what the different application classes require from the construction process, and how they are related to the re-usability and complexity of the interface modules. Next, the design process and the different methods of interface construction are briefly investigated.

On the basis of *Table 1* (i.e. the cross-references between MMS characteristics and application areas), the adaptivity classes and MMS application area classes may be related to each other, as is shown by *Table 3* (an 'x' represents the possibility of and demand for applying the related adaptivity type). Note that some alterations may occur depending on the skill and intelligence of the human user and operator, as well as on machine capabilities. It is also to be noted that *Table 3* illustrates the present situation; with the development of interfacing techniques, more and more x's may appear in future.

From *Table 3*, some interesting facts may be derived concerning the up-to-date state of applying adaptivity in HCIs. First, it can be seen that *CIS allow and require the highest level and variety of adaptivity*. Next is CHA, while COAs need and allow much less. The lowest level of need and possibility of introducing adaptivity is with CHE. As far as CISs are concerned, the main properties requiring a high degree and variety of adaptivity types are:

- bi-directional human-computer communication
- communication between computers and non-computing machinery also involved
- communication between humans and non-computing machinery may also be involved
- real-time processes (timing!)

Table 3 Adaptivity classes vs. MMS application areas

	CHA	CHE	COA	CIS
Simulator-based advance user adaptation without HC cooperation (1.1.1.1)				×
On-system advance user adaptation without HC cooperation (1.1.1.2)	×		×	×
Out-of-process real-time user adaptation without HC cooperation (1.1.2.1)	×		×	×
On-process real-time user adaptation without HC cooperation (1.1.2.2)	×	×	×	×
HC cooperation-based advance user teaching (1.2.1.1)	×		×	
HC cooperation-based advance user training (1.2.1.2)	×		×	×
HC cooperation-based real-time user examination (1.2.2.1)	×	×	×	×
HC cooperation-based real-time user training (1.2.2.2)				×
Advance IF adaptation by interviewing the user (2.1.1.1)	×	×	×	×
Advance IF adaptation by testing the user (2.1.1.2)			×	×
Query-based out-of process repetitive real-time IF adaptation (2.1.2.1.1)	×		×	
Testing-based out-of process repetitive real-time IF adaptation (2.1.2.1.2)				×
Analysis-based real-time stress limitation by observing the user (2.1.2.2.1.1)		×		×
Analysis-based real-time stress optimization by observing the user (2.1.2.2.1.2)				×
Analysis-based real-time variation of user support by event observation (2.1.2.2.2.1)	×	×	×	×
Analysis-based real-time variation of user involvement by event observation (2.1.2.2.2.2)	×			×
Conversation-based advance IF-adaptation by human control (2.2.1.1)	×	×	×	
Collaborative testing-based advance IF-adaptation (2.2.1.2)				×
Discussion-based out-of-process repetitive real-time IF adaptation (2.2.2.1.1)	×			×
Self-testing-based repetitive real-time IF adaptation (2.2.2.1.2)				×
Analysis-based real-time stress limitation by consulting the user (2.2.2.2.1.1)				×
Analysis-based real-time stress optimization by consulting the user (2.2.2.2.1.2)		×		×
Analysis-based real-time variation of user support by event evaluation (2.2.2.2.2.1)	×	×	×	×
Analysis-based real-time variation of user involvment by event evaluation (2.2.2.2.2.2)	×	×	×	

- high reliability requirements
- crucial (large-scale) economy aspects
- extreme complexity
- sophisticated processes.

Second, it can also be recognized that on-process training, user examination/advising, conversation-based advance adjustment and two kinds of variable user support (1.1.2.2, 1.2.2.1, 2.1.1.1, 2.1.2.2.2.1 and 2.2.2.2.2.1 in *Table 3*) are the only adaptivity types applicable/applied in virtually all application areas. These are the ones requiring most practical attention and, at the same time, these are the ones allowing *maximum re-usability* of the adaptivity modules in the interfaces (maximum possibility of using universal interface modules instead of application-oriented or application-specific ones).

Third, it is to be observed that there are several adaptivity types used just in one specific application area (see 1.1.1.1, 1.2.2.2, 2.1.2.1.2, 2.1.2.2.1.2, 2.2.1.2, 2.2.2.1.2 and 2.2.2.2.1.1. in *Table 3*). Generally, they do not attract too much practical attention, but, on the other hand, they require *specific* and *complex interface modules* (and thus, lack of their re-usability here means very *high relative cost*). What is more important, they need a relatively high amount of research until appropriate development can lead to successful practical application.

The *design* of adaptive interfaces is to be performed in all 'regular' cases in two steps. First, the normal push-pull mechanisms of information transfer (demand and supply of information for both parties in the MMS) are to be realized. Second, the necessary deviations from this normal mechanism are to be added (with respect to the required adaptivity types) at the appropriate hierarchical levels in the structural interface construction. Here, the design process should take into account the appropriate modes and media, in accordance with the occurring contents and forms of information transfer. This second step may be considered as a *perturbation method*: the elementary subfunctions (and their structural equivalents, the elementary modules) of the interface are perturbed by introducing alternative and/or tunable possibilities (and corresponding human-requested, semi-automatic or automatic selection/adjustment algorithms) for information transfer into the original (nominal) design. Of course, there are some cases where it is impossible to use the perturbation principle, and a single-step design method must be applied because of the very strong coupling between normal and adaptive interface operation. However, these cases may be considered as 'irregular', requiring a totally specific method of design.

By taking into account the aspects mentioned, we can recognize that the different construction principles applicable to the design of adaptive interfaces for computer-integrated systems lead to three easily distinguishable *different methods of IF construction*:

- Custom design (applying ASICs only, and using specific built-in adaptivity algorithms.
- Variational design (applying a combination of UNIC/AOIC/ASIC components and using advanced or real-time selection from among working states for achieving adaptivity).
- Parametrized design (applying UNIC and AOIC components only and using advance or real-time fitting and calibration of the IF parameters to achieve adaptivity).

All three principles allow the utilization of both (combined) methodologies of:

- modular construction (the interface is built up of elementary IF subfunctions/modules); and
- perturbational construction (based on alternative states/working modes and the tunability of I/O characteristics).

Finally, it should be mentioned that more complex (more advanced) adaptivity schemes require and/or allow:

- more levels of interaction
- more media and modes (content and form) of interaction
- involvement of more PSFs (internal and external factors, stressors, etc.) into adaptation
- wider choice of skill-based, rule-based and even knowledge-based interaction elements, and
- more modularity/hierarchy in the interface structure and functions.

Future of human centred interfaces in computer-integrated systems

Let us try to look ahead to the possible future perspectives of adaptive interfacing (again, bearing in mind human-centredness aspects of computer-integrated systems). The basis of this *review* is again the taxonomy introduced in the previous sections. Special attention is paid here, too, to computer-integrated manufacturing/production system applications.

Moving up and down in the list of the adaptivity classes also means a walk through the *evolution phases* of adaptive interfacing. Some of the items in the list are mainly future possibilities, while most items belong to normal everyday practice in today's man-machine systems, even if their complexity and cost vary across a wide range.

While today's practice is using predominantly the less sophisticated *basic adaptivity categories* of the classification tree of *Table 2* (in contrast to the more complex *real-time automatic adaptation* types that are not often applied at present), today's research is covering almost homogeneously the full list. However, one can hardly imagine what methods of adaptivity will still be achieved in forthcoming years/decades.

As far as the many PSFs are concerned, adaptive support is most frequently matched to the needs of efficient, reliable and friendly operation of the entire man-machine complex, but with the evolution of adaptive interfacing, more and more (external and/or internal, as well as stressor) factors will also be taken into consideration.

Computer-integrated manufacturing systems and computer-integrated production systems are extremely susceptible to problems due to the lack of appropriate adaptive support. This is because the adequacy, effectiveness, reliability and friendliness of system operation are the most crucial elements of these classes of computer-integrated systems. Thus, advanced forms of man-machine interaction and, especially, application of sophisticated adaptive interfaces are of key importance in future CIMS and CIPS.

It should also be mentioned that the role of formal interaction-modelling, the importance of an abstract/structural interface hierarchy, the integration of interaction modes and media, the sophistication of interface modularity and the exploitation of the advantages in combining/integrating conceptual-functional-physical design aspects will probably be increasing in future. However, we are just at the outset of adaptive interfacing, and of using human centred computer-integrated systems. Thus, a look at the future is only able to emphasize probable outcomes for the *near* future. Many more may still follow, at the moment being outside of our imagination.

Summary

Adaptive human-computer interfaces serve as a solid basis for achieving human centredness in computer-integrated systems. Although both the theory and practice of using adaptive interfaces in computer-integrated systems, and especially computer-integrated manufacturing/production systems, are just at their beginning, it is quite clear that future computer-integrated systems will exploit extensively the advantages of those adaptive interfaces that have been briefly investigated in the above study.

Acknowledgement

The author is indebted to the Hungarian National Research Fund (OTKA T-4468) for kindly supporting the work on man-machine systems, human-computer interaction and adaptive interfaces.

References

1 **Rasmussen, J and Andersen, H B** *Human-Computer Interaction*, Lawrence Erlbaum, Hillsdale, NJ (1991)
2 **Helander, M (ed)** *Handbook of Human-Computer Interaction*, North-Holland, Amsterdam (1988)
3 **Rouse, W B (ed)** *Advances in Man-Machine System Research*, JAI Press, Greenwich, CT (1984)
4 **Lupton, T (ed)** *Human Factors, Man, Machine and New Technology*, Springer-Verlag, Berlin (1987)
5 **Card, S K, Moran, I P and Newell, A** *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum, Hillsdale, NJ (1986)
6 **Weatherall, A** *Computer Integrated Manufacturing*, Butterworth-Heinemann, Oxford (1988)
7 **Norman, D and Draper, S W (eds)** *User Centered Design; New Perspectives in Human-Computer Interaction*, Lawrence Erlbaum, Hillsdale, NJ (1986)
8 **Schmid, F, Hancke, T and Wu, B** *The Human-Centred Approach to the Implementation of Computer Integrated Systems*, Report, Brunel University, UK (1992)
9 **Bayley, R W** *Human Performance Engineering: A Guide for System Designers*, Prentice Hall, Englewood Cliffs, NJ (1982)
10 **Browne, D, Totterdell, P and Norman, M (eds)** *Adaptive User Interfaces*, Academic Press, London (1990)
11 **Malinowski, U, Kühme, T, Dietrich, H and Schneider-Hufshmidt, M** 'A taxonomy of adaptive user interfaces', in Monk, A, Diaper, D and Harrison, M D (eds), *People and Computers VII*; (*Proc. HCI '92 Conf.*), Cambridge University Press, Cambridge (1992)
12 **Bálint, L** 'The role of models in handling complexity, flexibility and reliability of human-computer interfaces, *Proc. 3rd IEEE Ann. Euro. Comput. Conf. "COMPEURO'89"*, Hamburg, Germany (1989)
13 **Bálint, L** 'Approaching optimum HCI properties in industrial process control applications', *Proc. 1st Int. Conf. Applic. Ind. Electr. Syst.*, Jerusalem, Israel (1990)
14 **Bálint, L** 'New aspects of constructing HCIs in hybrid automation: borrowing key VLSI concepts can enhance efficiency and decrease cost', *Proc. 2nd Int. Conf. Human Aspects of Advanced Manuf. Hybrid Automation*, Honolulu, HI (1990)