Modeling the User's Processing Resources: Pragmatic Simplicity Meets Psychological Complexity

Anthony Jameson*
Department of Computer Science, University of Saarbrücken
P.O. Box 151150, D-66041 Saarbrücken, Federal Republic of Germany
jameson@cs.uni-sb.de

Abstract

A number of methods have been proposed for reasoning about the demands that particular tasks place on a user's limited cognitive processing resources. On the other hand, the notion of "cognitive resources" has been strongly criticized by psychologists. This paper addresses the question of whether these criticisms are applicable to user modeling. It first introduces and illustrates a framework for reasoning about resource-limited processing. It then considers the main criticisms of psychological resource theories in turn. It argues that these criticisms can be taken into account, with the help of the framework presented here, by designers of on-line user modeling systems.

1 Introduction

Throughout the history of research on user and student modeling, the key variables modeled have usually concerned the user's knowledge, beliefs, goals, preferences, interests, and proficiencies. Recently, however, some models have begun to take into account explicitly the obvious fact that a human's capacity to perform tasks at any given moment is limited: The user cannot at a given time perform an arbitrarily complex task—or an arbitrary number of tasks. These performance limitations are often seen as the consequences of limitations of *cognitive processing resources*; the meaning of this term will be examined in Section 2.

Limitations of processing resources have been taken into account for decades by the designers of interactive systems, who have attempted to minimize unnecessary processing by the user by (a) having the system take over as many tasks as possible and (b) optimizing the work that remains for the user. And more specifically in the area of user and student modeling, one of the purposes of modeling the user's knowledge is to allow the system to adapt so as not to force the user to engage in more complex processing than necessary. But it is a fairly new idea for the system to model explicitly the extent to which particular tasks place demands on the user's cognitive resources.

1.1 Approaches to Off-Line Modeling of Cognitive Resources

In the field of human-computer interaction, researchers have not attempted to allow a system to reason on-line about the user's cognitive capacity; but some have supplied designers with formalisms and conceptual frameworks within which they can perform such reasoning in a systematic way when designing systems. Because of space limitations, only a few prominent examples can be mentioned here.

In late 1970's Card et al. (see, e.g., Card et al., 1983) presented the Model Human Processor, a simplified conceptualization of the human cognitive architecture that allowed designers to anticipate the processing that a user would have to engage in in order to perform elementary tasks. The model includes three processors (perceptual, cognitive, and motor); long-term and working memory; and visual and auditory image stores. It summarizes their basic properties (e.g., cycle duration, storage

*This research was supported by the German Science Foundation (DFG) in its Special Collaborative Research Program on Resource-Adaptive Cognitive Processes, SFB 378, Project B2, READY.

capacity, decay rate). The model also formulates 10 principles of operation. Using this model, designers can in some instances identify in advance cases in which a system would probably place too heavy demands on the user. Though the Model Human Processor is probably the most detailed model designed so far for its purpose, it does reflect a good deal of conscious simplification, as the authors note. A key idea underlying this approach, which will be adopted in this paper, is the following one: The goal of an engineering-oriented model of the human cognitive system is not to make precise predictions or to explain why the system functions the way it does; rather, the goal is to support approximate reasoning about particular user behaviors.

From a somewhat different perspective, a number of researchers have presented formalisms for cognitive task analysis that make it possible to predict the information-processing demands that a system will place on a user who is performing a given task. For example, the formalism may include a specification of the information that the user has to store in working memory at each point during the course of the execution of a given task (see Olson & Olson, 1990, for a discussion of some formalisms of this sort). Instead of taking into account diverse cognitive resources, as the Model Human Processor does, these analysis methods typically focus on working memory as an especially important limited resource.

1.2 Approaches to On-Line Modeling of Cognitive Resources

Within research on on-line user modeling, the most explicit reference to cognitive resources is found in the READY project (see, e.g., Schäfer & Weyrath, 1997; Berthold, in this volume). READY makes uncertain inferences about the working memory of the user, for example, predicting whether $\mathcal{U}$ will be able to understand a particular complex instruction while simultaneously carrying out an action. So far, READY has used a very simple, global conception of working memory, on the grounds that the limited amount of evidence that is available does not allow detailed inferences about $\mathcal{U}$'s working memory anyway.

1.3 Preview of the Rest of This Paper

Despite their differences, the approaches just sketched share the assumption that it is possible to predict and explain limitations of a user's performance in terms of limitations of processing resources. Similar assumptions have also been made by psychological researchers in various forms over the past few decades (see, e.g., Herrmann & Grabowski, 1994, for a recent example). On the other hand, there has also been severe criticism of various forms of "resource theory". The question therefore arises: Does this criticism is also apply to approaches to on-line user modeling that make use of the concept of *resources*?

In order to answer this question, we first have to introduce explicitly an appropriate conception of *resource*. Section 2 presents such a framework, and Section 3 illustrates the framework with two examples that involve cognitive resources. Using the concepts of this framework, Section 4 discusses the criticisms that have been leveled at psychological resource theories and assesses their significance for user modeling. Finally, Section 5 summarizes the consequences of this analysis for user modeling.

2 A Theoretical Framework for Reasoning About Resource Allocation

2.1 Desired Properties of the Framework

If we want to deal systematically with the above issues, it will be useful to have a theoretical framework for reasoning about resource allocation. There already exist various formalizations of resource-related concepts: for example, in the literatures on operations research, project management, planning under resource constraints, and economics. While we can borrow some ideas from these existing approaches, we require a framework with some properties not in general exhibited by the previous frameworks:

Inclusiveness. Many definitions of resource-related concepts introduce restrictions that exclude some entities that might in some ways be viewed as resources. For example, one can specify that a resource must be continuously quantifiable (which excludes any individual person), or that it must be consumable (which excludes information). These restrictions may make sense within a particular context. But since one of our questions is that of what entities can meaningfully be treated as resources in the first place, it is better for us to choose a conceptualization that includes everything that might from some point of view be considered a resource. Then, restrictions can be introduced to define subclasses of the general class *resource*, and the utility of the various subclasses relative to that of the broadest class can be examined explicitly.

Neutrality With Respect to the Type of Application. There are various contexts in which one can reason about resources, including the following:

- planning to accomplish tasks under resource constraints;
- predicting how another agent will plan under resource constraints and/or recognizing the plans of such an agent;
- explaining the observed behavior of an agent in terms of resource limitations, even though the behavior is not the result of a planning process.

It is possible to formalize resource-related concepts in a language that is appropriate for one of these purposes—for example, a language for hierarchical planning for the first type of application. But clarification of the central concepts is best served by a general framework that defines concepts and relationships in a way that supports their use for various types of reasoning.

Variable Degrees of Granularity and Precision. There are several reasons why it can be desirable to be able to choose the degree of precision and accuracy with which one reasons about resources:

- Tractability
  In some contexts fine-grained, precise reasoning can be computationally intractable. For example, hierarchical planning systems start with coarse-grained reasoning and then refine the results using a more differentiated model. Similarly, when planning how to deal with resources a system may start with a simplified model of them and then use a more precise model when refining the initial plan.
- Dealing with uncertainty
  Lack of information may force one to use a simplified model. For example, when software firm A decides how to allocate its 10 programmers, it may take into account each programmer's specific capabilities. But if firm B is trying to predict what firm A will be able to accomplish, B may know only that A has 10 programmers. Since B does not know their specific properties, B may as well view A's programmers as a homogeneous resource.

2.2 The Framework

Figure 1 illustrates the basic concepts graphically. Throughout this paper, we assume the following restricted situation: Within a given time slice, a set of tasks $\{T_j\}$ is to be performed. There is a set $\{R_i\}$ of resources available, each of which is to be allocated among some subset of the tasks. The success of the performance of each task will depend on the resources allocated to it, among other things. The figure depicts one typical model of resource-dependent processing within such a situation.

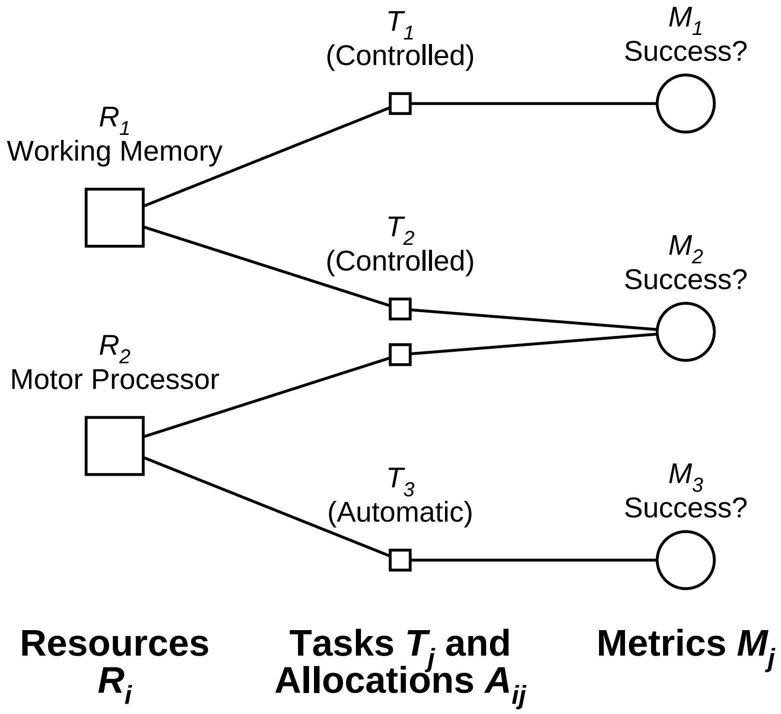

Figure 1: Graphical of key concepts in a framework for reasoning about resources.
(The lines on the left indicate which resources can be allocated to which tasks. The small squares in the middle represent the specific allocations. The lines on the right reflect the fact that the performance metric for a task is a function of the resources that were allocated to it. The specific concepts in the left-hand and middle columns refer to the example that is discussed in Section 3.1.)

Whenever a concrete model is to be formulated for a particular purpose, a number of aspects have to be specified. These are discussed in turn in the following paragraphs:

Expressions for Describing Allocations

In each instance where a resource R_i can be allocated to a task T_j (as indicated by a line on the left-hand side of the figure), it must be possible to characterize each possible allocation A_{ij}. For different resources, these expressions may take quite different forms. So for each resource, the (possibly infinite) set of possible *allocation expressions* must be specified, as must the semantics of the expressions. Here are some examples of such sets, expressed in terms of the form that a single element of the set can take. In each case, the semantics of the expressions are described verbally. For concreteness, we consider the case where each R_i is a hotel that may be allocated in some way to the task T_j of holding a particular conference.

- A Boolean value
  R_i is viewed as being either allocated to T_j (`true`) or not (`false`). (For example, it might be specified simply which hotels are to serve which conferences.)
- A nonnegative real number
  A_{ij} might represent the amount of R_i allocated to T_j on some specified scale (e.g., the number of rooms of hotel i that will be used for conference j).
  Alternatively, the scale might refer to a dimension of the tasks, such as the number of attendees of conference j who will be housed in hotel i.
- Assuming that R_i can be viewed as a set: A subset S_i of R_i.
  For example, if hotel i is viewed as a set of rooms, A_{ij} might specify which particular rooms are to be used for conference j. This representation is more fine-grained than one that represents

only the *number* of such elements that are allocated to T_j. Such a specification may make sense if the identity of these elements is important (e.g., if the rooms differ in size and price).

- Assuming that T_j is associated with some set E_j of entities that have to be handled: A subset of E_j.
  For example, A_{ij} might specify which of the events E_j that constitute conference j will take place in hotel i. Here again, such a fine-grained representation can be worthwhile if the elements of E_j have different properties.
- Assuming again that R_i can be viewed as a set: A set of pairs of the form (S_{ik}, x), where S_{ik} is a subset of R_i and x is a nonnegative number.
  Here, each subset of R_i that is allocated to T_j is viewed as being allocated to a certain degree. (For example, one group of rooms S_{ik} in hotel i might be allocated for half of the time slice to conference j and for the other half to conference $j+1$.) This type of specification is a recursive application of the second type given above to *subsets* of R_i.

Specifications of Allocation Constraints

A central aspect of the concept of a *resource* is that there are some limitations on the ways in which it can be allocated. A *specifiable allocation* of R_i to the tasks T_j is a vector $(A_{i1}, \ldots, A_{in})$ in which each A_{ij} is an allocation expression of the language. In general, only a subset of the set of specifiable allocations can actually be realized; the *allocation constraints* define this subset. Here are some examples of allocation constraints:

For Boolean allocation expressions:

- At most N tasks can have the allocation `true`.
  Here, R_i is a resource that can be used by at most N tasks within a given time slice. For example, R_i may be a processor that can work on only two tasks in any given time slice.

For numerical allocation expressions:

- $\sum_j A_{ij} \leq Q_i$ for some number Q_i
  Here, R_i is a divisible resource of which a certain total quantity Q_i is available.
- $\max_j(A_{ij}) \leq Q_i$
  Here, the quantity Q_i of R_i can be made available to all tasks at once. For example, R_i might be 20 minutes of sunshine that can be used by solar cells that do not compete with each other. This example illustrates that competition among tasks for R_i—and/or consumption of R_i—are not necessary for it to be meaningful to view R_i as a "resource". As long as there are constraints on the allocation of R_i and the allocation influences the performance metrics, the idea of a resource is applicable.
- $\sqrt{\sum_j A_{ij}^2} \leq X_i$
  Here, there is some competition among tasks for R_i, but it does not take the simple form of having to divide up a certain fixed quantity. So X_i cannot be viewed as the "total amount" of R_i that is available.

For specifications of subsets of R_i:

- There can be no overlap among the subsets specified by the A_{ij}
  Here again, R_i is a resource that has to be divided up. The difference from the previous related example ($\sum_j A_{ij} \leq Q_i$) is just that the description of each allocation is more specific.

For any type of allocation expression:

In the worst case, there may be no parsimonious way of specifying the possible allocations. But it may still be possible to list them explicitly. For example, suppose that the 5 tasks $\{T_1, \ldots, T_5\}$ require the storage of 5 objects in a store R_i. It may be known on the basis of previous experience exactly which subsets of the objects can be stored successfully simultaneously; but it may be difficult or impossible to characterize this set of subsets in any other way than by enumeration, since the successful storage of a subset of objects may depend in a complex way on various properties of the objects and their relationships to each other. (Consider, for example, the problem of determining which of a set of animals of different species can be transported together in a single truck.)

Functions Specifying Allocation Costs

Among the allocations of R_i that are possible, there may be differences in costs (whereby various different metrics may be used for cost, or cost may even be viewed as having more than one quantifiable aspect). For example, allocations involving relatively large quantities or subsets of R_i may be relatively expensive. Since high cost may trade off against high values of the performance metrics, the cost of resources will often play an important role in reasoning about them. On the other hand, with resources that are aspects of the human cognitive architecture, the resources can in general simply being viewed as being available for allocation, and the concept of *cost* is less important. Accordingly, resource costs will not be discussed in this paper, though the concept may sometimes prove useful even when cognitive resources are involved.

Performance Profiles

For each task T_j, there must be a specification of how the expected success of its performance, indexed by some metric M_j, depends on the resources allocated to it. The expressions for the degree of success may vary in complexity, as the following examples illustrate:

- one of the symbols `success` and `failure`;
- the probability of success;
- a symbolic or numerical assessment of the degree of success on some scale;
- a vector of such symbolic or numerical assessments that correspond to several metrics (e.g., quality of result, speed of completion).
  In this case, it may be clearer to specify several simpler performance profiles, one for each criterion.

Of course, the success of the performance of a task will depend not only on the resources allocated to it but also on other factors, including aspects of the context in which the task is performed. In some cases it will be worthwhile to specify some of these factors as additional parameters of the performance profile.

2.3 Remarks on the Framework

A number of complications are intentionally omitted from this picture:

1. The performance of T_j is assumed to be dependent on the resources that are allocated to it but otherwise to be independent of the performance of the other tasks. For example, the results of T_j cannot enhance or interfere with those of another task T_k.

2. The issue of the temporal order and duration of the tasks does not arise, because only one time slice is involved. Once these issues are taken into account, techniques such as critical-path analysis (see, e.g., Olson & Olson, 1990) become relevant.

3. It is not considered how the allocation constraints, the allocation cost functions, and the performance profiles are determined. In some cases, considerable computation or reasoning may be required to estimate (even with considerable uncertainty), for example, whether a particular allocation of R_i is possible and how much it would cost; or how the performance of T_j would rate on the performance metrics if it were given a particular allocation of resources. These types of computation may figure prominently in a system's reasoning, but they are put in the background here by our assumption that the relationships in question exist.

4. It is not considered how particular decisions might be made. For example, to determine an optimal allocation of resources, a system might iterate over all possible combinations of allocations

of all resources, determining for each combination the cost of the allocation and the values on the performance metrics for all tasks, combining these quantities to arrive at an overall evaluation of the allocation, and seeing which allocation yields the best overall result. But if the model is reasonably complex, this brute-force approach may well be prohibitively expensive, in which case more sophisticated approaches are required. Given our focus on the definition of concepts rather than on the solution of particular types of problems, it makes sense to omit consideration of such issues.

3 Two Example Models

Although the framework introduced in the previous section is intended to be applicable to all sorts of resources, its use will be illustrated here with two examples involving cognitive resources. The examples introduce some issues that will be discussed in the subsequent sections.

3.1 A Model With Coarse Granularity

The specific model in Figure 1 reflects the following simple theory, which is sometimes viewed as a useful approximation: There are two types of tasks that people can perform: those that require *controlled processing* (here called *C-tasks*) and those that involve *automatic processing* (*A-tasks*) (see Shiffrin & Schneider, 1977, for an early, much more differentiated discussion of this distinction). C-tasks place demands on a limited working memory; consequently, only one such task can be performed at a time. Since automatic processing does not require working memory, an A-task can be performed simultaneously with a C-task, subject to other constraints.

The figure depicts a case in which a user $\mathcal{U}$ is considering three tasks that it would be desirable to perform within the next time slice. T_1 (e.g., evaluating a manuscript) and T_2 (e.g., writing an e-mail message) both require controlled processing. By contrast, T_3 (e.g., manually sealing a large number of envelopes) can be performed automatically. T_2 and T_3 both require motor output (e.g., $\mathcal{U}$'s hands have to be used). Thus T_1 and T_2 compete for working memory while T_2 and T_3 compete for the motor processor. One way of formulating the allocation constraints and performance profiles is as follows:

Allocation constraints for working memory (R_1) and for the motor processor (R_2):

- Syntax of allocation expressions: `true`, `false`
- Semantics of allocation expressions: A_{ij} = `true` iff T_j uses R_i
- Allocation constraints: For each i, at most one A_{ij} is `true`

Note that no distinction is being made with regard to the exact extent to which each of the tasks places demands on the user's working memory or motor processor.

Performance profiles:

- $M_1 = \texttt{success}$ iff $A_{11} = \texttt{true}$
- $M_2 = \texttt{success}$ iff $A_{12} = A_{22} = \texttt{true}$
- $M_3 = \texttt{success}$ iff $A_{23} = \texttt{true}$

Note that the only distinction being made concerning success is whether a given task can be performed or not. For example, it might be assumed that the user's ability to evaluate the manuscript is in no way impaired if he or she simultaneously works on sealing the envelopes.

The simple consequence of these specifications is that either of two subsets of the set of tasks can be performed successfully: $\{T_1, T_3\}$ and $\{T_2\}$ (as well as subsets of these sets). Which subset $\mathcal{U}$ will actually perform depends on the way in which $\mathcal{U}$ arrives at a decision; for example, $\mathcal{U}$ might take the importance of successful performance of the various tasks into account. What the analysis shows is how the possible outcomes are constrained by resource limitations.

This analysis is clearly coarse-grained and approximate: Each of the two resources could be represented in more detail, and the allocation constraints could do more justice to reality. For example,

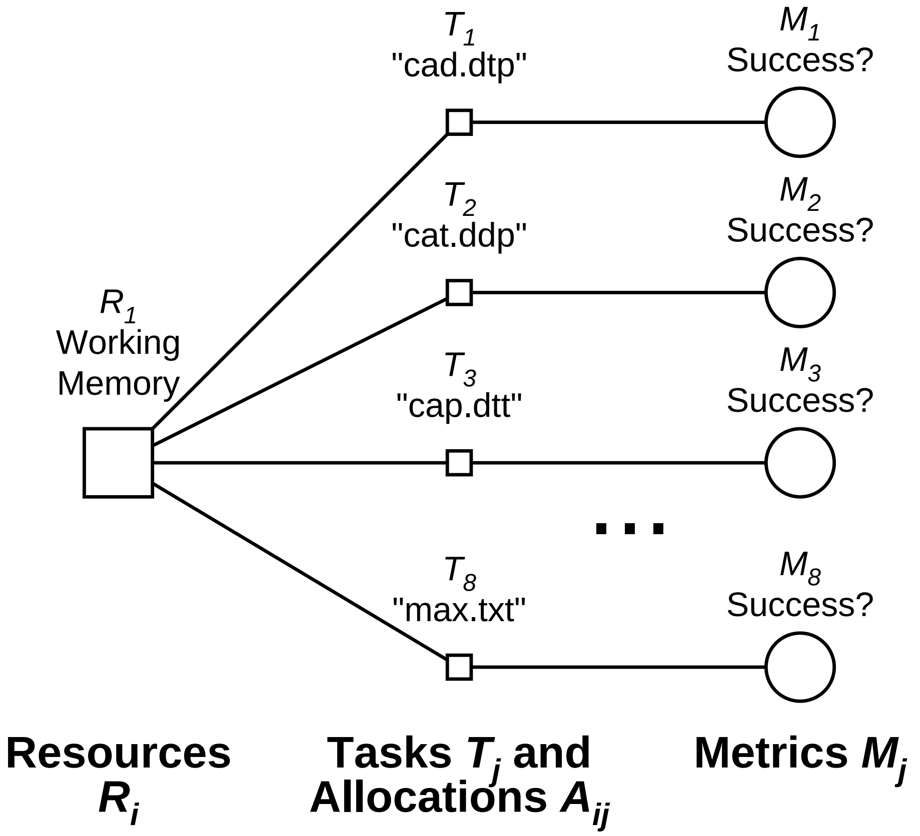

Figure 2: A relatively fine-grained model of the problem of using working memory to store file names. (Explanation in text and in Figure 1.)

it is often possible to perform two tasks that require some working memory simultaneously; or to perform two tasks that require motor processing (e.g., using the hands and speaking). Within a finer-grained model, it would probably also be worthwhile to use more differentiated performance metrics, for example reflecting the quality of the results of the tasks T_1 and T_2.

The next example gives a more fine-grained analysis of a different situation.

3.2 A Fine-Grained Model

Consider a user who sees a list of eight file names, who wants to remember as many as possible of them for a few seconds, and who can store them only in working memory. The analysis in Figure 2 views the remembering of each file name as a separate task. There is now only one resource, working memory. The form of the allocation expressions is the same as in the previous example, but their meaning is different.

Allocation constraints for working memory (R_1):

- Syntax of allocation expressions: $\{\texttt{true}, \texttt{false}\}$
- Semantics of allocation expressions: $A_{1j} = \texttt{true}$ iff T_j uses R_1 (i.e., the file name associated with T_j is stored in working memory)
- Allocation constraints:
  These can be specified at different levels of granularity:
  - At most N of the A_{1j} are `true`.

- For the set of j such that A_{1j} is `true`, the total number of syllables in the corresponding file names is at most N.
- For the set of j such that A_{1j} is `true`, the total number of information chunks in the corresponding file names is at most N.
- ...
- The following subsets of $\{T_1, \ldots, T_8\}$ can all have $A_{1j} = \texttt{true}$ (as can their subsets):
  - $\{T_1, T_2, T_3, T_4, T_5\}$ (five different-sounding file names)
  - $\{T_6, T_7, T_8\}$ (three similar-sounding names)
  - ... [further subsets]

Performance profiles:

- $M_j = \texttt{success}$ iff $A_{1j} = \texttt{true}$

The most important issue in this analysis is how to formulate the allocation constraints. There is no simple principle on the basis of which one can predict which contents can be held in working memory. Formulations such as the first three alternatives given above, which quantify the total storage capacity and the size of each element, may be convenient approximations; but there are other factors determining what can be stored that cannot be described in this way. For example, the number of names that can be stored is relatively small if the names sound similar, because their phonological codes tend to interfere with one another (cf., e.g., Card et al., 1983, pp. 79–80). The least parsimonious way of formulating the allocation constraints would be to enumerate all of the possible allocations. A more desirable way would be to specify a decision procedure for determining whether a given allocation is possible, whereby the procedure takes into account factors like the relationships among the individual elements. Which specification is most appropriate depends on the specific context and on the purpose for which the model is to be used.

Note, however, that even in the worst case of an enumerative specification, it still makes sense to view working memory as a limited resource. For example, one can explain why $\mathcal{U}$ did not remember all of the file names, even though it might have been highly desirable for him to do so, by saying that there was no possible allocation of working memory that would have enabled him to accomplish all of the tasks successfully.

4 Discussion of Criticisms of Resource Theories

Especially during the 1970's and 1980's, a number of psychological theorists formulated *resource theories*, which aimed to explain performance limitations in terms of limitations of cognitive resources. These formulations have been subjected to considerable empirically and theoretically based criticism. The research and arguments in question have been ably summarized by a number of authors (e.g., Meyer & Kieras, 1997, Neumann, 1992, and Rummer, 1996, chap. 3), so it is not necessary here to recapitulate the discussion in any detail. In the next three subsections, three of the main criticisms will be discussed in terms of the framework introduced in Section 2 and the implications of each criticism for research on on-line user modeling will be considered.

4.1 Criticism 1: Unsatisfactory Accounts of Structural Interference Effects

One type of empirical result that resource theories had difficulty with concerns cases in which simultaneous processing of different sets of stimuli or responses depends on the specific nature of the stimuli or responses. An example of such a case was discussed in connection with Figure 2. Another example is the fact that performing two mental arithmetic tasks simultaneously is in general much more difficult than performing one such task while reciting a poem (see Neumann, 1992, pp. 85–86, for a discussion of this and similar results). To a certain extent, one can account for such constraints by postulating a more differentiated set of resources (e.g., a visuo-spatial scratchpad in which images can be stored and a phonological loop in which words can be rehearsed; see, e.g., Baddeley, 1986). This approach was used by resource theorists; it led to a rather disturbing increase in the number of resources postulated, and in many cases its success was limited: Consider trying to explain interference among verbally presented names in working memory by postulating specialized subsystems of working memory (e.g., one for names beginning with "a", one for those beginning with "b", etc.).

Another way of accounting for specific interference effects within a resource framework is to retain the original grain size of the resource in question but to specify the allocations and the allocation constraints in a more fine-grained fashion. This is what was proposed in connection with the example in Figure 2. In this way, the number of resources is not increased and the complexity of the allocation constraints can be dealt with as accurately as is practically possible and necessary.

A possible objection to this strategy concerns its explanatory power: If you specify a complex set of allocation constraints for a resource R_i, you may be able to predict *which* allocations of R_i are possible, but you may have no explanation of *why* these particular constraints exist—at least not an explanation that itself refers to resource limitations. (Note, by contrast, that if the allocation constraint simply specifies a total quantity of R_i that can be distributed among tasks, the constraint not only predicts possible allocations but also explains why exactly these allocations are possible, and it does this without invoking any further considerations.)

From a practical point of view, however, it doesn't really matter how the allocation constraints are to be explained. It is of course inconvenient in almost any context if the allocation constraints are complex rather than simple; but this fact does not speak against the use of resource concepts.

4.2 Criticism 2: Proliferation of Resources and of Degrees of Granularity

Some early resource theories tried to explain a broad range of phenomena with models of the general degree of simplicity shown by the example given in Figure 1. As has already been mentioned in connection with that example, empirical results often made it necessary to add further resources to the picture—either by taking into account different resources that had previously been left out (e.g., a perceptual processor as well as a motor processor) or by viewing a given resource as consisting of subsystems that could themselves be viewed as resources (e.g., distinguishing between processors for the hands and the feet).

Part of the problem was the postulation of new resources to explain structural interference effects, as discussed in the previous section. But even if this move is avoided in the way sketched above, it does remain difficult or impossible to define once and for all a set of resources and allocation constraints that can be used to explain all new results concerning performance limitations. Instead, it often seems necessary to formulate a new model for new situations, choosing the appropriate set of resources to consider and an appropriate level of granularity at which to define the resources and the tasks.

This state of affairs is unsatisfactory from the point of view of psychological theorizing. What one desires in this context is a unified framework that can explain and predict observations—not a method for generating a great variety of partial models. Even though this ideal has in many cases proven difficult to attain, it does represent a characteristic goal of basic psychological research.

From a pragmatic, engineering point of view, the unavailability of a single, constant framework makes matters more complicated and inconvenient, but it does not render the enterprise pointless. Any user modeling system is likely to deal with some limited universe of tasks. The designers will have some idea, when building the system, what sorts of resources are potentially relevant, what types of data are likely to be available, what sorts of inferences have to be made, and how accurate the inferences have to be. On the basis of this information, the designer can choose an appropriate model (or set of models), making judicious choices about matters such as the grain size of allocation constraints. If the resulting system makes incorrect inferences, these can be viewed as either the inevitable consequences of approximate reasoning or a demonstration of the need to revise the model that is being used. As long as the system on the whole ultimately performs its tasks satisfactorily, such incorrect inferences do not pose the sort of problem that they do in the context of psychological theory formation.

4.3 Criticism 3: Lack of Evidence for Biological Bases for Allocation Constraints

One idea of resource theories that is often criticized is the notion that allocation constraints are due to biological limitations—such as the hardware limitations that give each computer a limited amount of RAM memory. To use an analogy: The fact that cars only permit one person at a time to drive does not necessarily reflect the fact that it would be expensive to provide two persons with the ability to control the car's movements. Even if such cars could easily be constructed, they might well be prohibited by law, since they would create an unacceptable danger that the two drivers would form different plans and perform incompatible actions (e.g., simultaneously accelerating sharply and shifting into reverse gear). Similarly, a person's inability to form detailed plans for several different actions simultaneously may be due to either executive control processes or inhibitory mechanisms that have the function of preventing incompatible commands from being sent to a limited set of effector organs (cf. Meyer & Kieras, 1997; Neumann, 1992).

Here again, the criticism addresses the explanatory value of resource concepts. If indeed the "capacity limit" of one driver per car is ultimately due to safety considerations rather than to limitations of space or machinery, then resource concepts cannot be used to *explain* why this limit exists. But for the purposes of predicting and explaining the behavior of cars, it can be useful to know that there is at most one driver. For example, we will then not be inclined to postulate a larger number of drivers as an explanation for observed behavior—whether successful or unsuccessful— that might conceivably have been produced by several persons guiding the car simultaneously.

5 Conclusions Concerning the Modeling of the User's Cognitive Resources

We have seen that it is possible to define resource-related concepts in such a way that they can be applied flexibly to a wide variety of aspects of the human cognitive architecture (as well as to other types of resources). The relatively inclusive definitions make it possible to apply the concept of a *resource* without having to make unrealistic assumptions about the entities in question.

Designers of user modeling systems may find the framework useful in that it allows them to specify a conception of the aspects of the cognitive architecture that they want to reason about which is independent of the formalism used for their reasoning. In particular, by considering alternative conceptions, they can make well-considered decisions about which one is most appropriate for their purpose.

Criticisms of psychological resource theories have highlighted the complexity of the factors that limit human performance, making it clear that any attempt to predict and explain such limitations in terms of resource concepts will involve error-prone approximation. But the claim that resource concepts should be abandoned applies to the goal of psychological theorizing: to provide a parsimonious theoretical explanation of what types of performance limitations exist and why they exist. The arguments advanced were not directed at the sort of enterprise that some user modelers are now engaged in: practically oriented attempts to predict and explain, in an approximative way, particular instances of performance limitation. In sum, in spite of the criticisms discussed above, the concept of *cognitive resources* may in a few years be found to be alive and well and widely employed in the area of user modeling.

References

Baddeley, A. D. (1986). *Working memory*. Oxford: Oxford University Press.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Erlbaum.

Herrmann, T., & Grabowski, J. (1994). *Sprechen: Psychologie der Sprachproduktion*. Heidelberg: Spektrum Akademischer Verlag.

Meyer, D. E., & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance: Part 1. basic mechanisms. *Psychological Review, 104,* 3–65.

Neumann, O. (1992). Theorien der Aufmerksamkeit: von Metaphern zu Mechanismen. *Psychologische Rundschau, 43,* 83–101.

Olson, J. R., & Olson, G. M. (1990). The growth of cognitive modeling in human-computer interaction since GOMS. *Human-Computer Interaction, 5,* 221–265.

Rummer, R. (1996). *Kognitive Beanspruchung beim Sprechen*. Weinheim: Beltz.

Schäfer, R., & Weyrath, T. (1997). Assessing temporally variable user properties with dynamic Bayesian networks. In A. Jameson, C. Paris, & C. Tasso (Eds.), *User modeling: Proceedings of the Sixth International Conference, UM97*. Vienna, New York: Springer Wien New York. Available electronically from http://um.org.

Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological Review, 84,* 127–190.