

## Redesign of technical systems

S.J.M. van Eldonk, L.K. Alberts, R.R. Bakker, F. Dikker, P.M. Wognum

*Department of Computer Science, University of Twente, P.O. Box 217, NL-7500 AE Enschede, Netherlands*

---

### Abstract

The paper describes a systematic approach to support the redesign process. Redesign is the adaptation of a technical system in order to meet new specifications. The approach presented is based on techniques developed in model-based diagnosis research. The essence of the approach is to find the part of the system which causes the discrepancy between a formal specification of the system to be designed and the description of the existing technical system. Furthermore, new specifications are generated, describing the new behaviour for the 'faulty' part. These specifications guide the actual design of this part. Both the specification and design description are based on YMIR, an ontology for structuring engineering design knowledge.

*Keywords:* Engineering design; Model-based diagnosis

---

### 1. Introduction

In the practice of technical design, it is customary to use an existing design as a starting point for a new design process [1]. Instead of starting design from scratch, an old design is adapted to meet new specifications. This process, named 'redesign', can be an efficient method for the design of a new product. By the reuse of existing knowledge, techniques or methods, it is possible to achieve a new design with much less effort than by means of design from scratch. Consequently, redesign can result in a major cost reduction. An example is the redesign of a personal stereo into a combined personal tape/radio stereo, which gives it an extra functionality.

Two short comments can be made. Firstly, the term 'redesign' can be used in a negative sense. In this case it concerns the repair of design faults created during the initial design process which have become apparent only later on during the production process. We do not commit ourselves to this view. We look at redesign as the design of a new product by reusing an old one.

Secondly, redesign is no guarantee of success. When a new set of specifications greatly differs from an old design, then redesign does not make sense. The general idea behind redesign is that a minimal adaptation is sufficient to achieve a new design. The adaptation of a bicycle into an automobile has nothing to do with redesign.

Our starting point for redesign is the availability of an

old design, together with new specifications. The central question of this paper is twofold. Firstly, how can we determine which part of the old design can be adapted and which part can remain unaltered. This task is referred to in the literature as diagnosis [2], blame assignment [3] or critique [4]. For the accomplishment of this task we will use techniques from model-based diagnosis (MBD).

Secondly, how can we achieve new specifications for the diagnosed part of the old design? When these specifications are derived we end with a 'normal' design problem, but on a smaller scale. These two tasks together we will address as design analysis, i.e. the reduction of the initial problem into a new problem which is hopefully easier.

The outline of the remainder of this paper is as follows. In the next section we will take a closer look at the redesign task. In Section 3 we will pay attention to the chosen design framework. Next, in Section 4, the way the analysis process is handled will be presented. Section 5 is dedicated to possible extensions to our method for analysis, based on different types of redesign. Finally, in the conclusion, amongst other things, we will point out further directions for research.

### 2. Redesign

#### 2.1. Definition

Redesign is characterized by the following input/

output relation:

### Redesign

- *Input:*
  - old design description,
  - new specification.
- *Output:*
  - new design description.

Some conditions apply: redesign takes place whenever the new specifications are not satisfied by the old design. The redesign is successfully ended when the new design satisfies the new specifications.

In general, redesign can be seen as a subtask of case-based design, namely as a case adaptation task. In this paper we do not consider the process of case retrieval, although within our project others are concerned with this problem [5]. It is, however, not necessary to view redesign as a subtask of case-based design. It is possible for an old design to be available or for an old design to be provided by a human designer.

### 2.2. Assumptions

To position our research more precisely, we will explicitly state some of our assumptions regarding redesign:

- The redesign task takes place in technical domains such as civil, mechanical and electrical engineering. In general, we concentrate on tangible products.
- A design is similar to a design description, as we do not consider the actual production process of the artifact.
- A specification is a formal specification. The stage of formulating global ideas in exact terms is omitted.
- A specification is a purely technical specification: costs, environment, ethics and aesthetics are not considered.

Besides these, we commit ourselves to a description language for the design specification and the design description. This view will be presented in the next section.

### 2.3. Subtasks

To make our ideas clearer, we present the subtasks of the redesign process that we distinguish. These are roughly based on models for redesign that can be found in the literature, e.g. those in [2,6]. We want to distinguish four subtasks: diagnosis, respecification, design and evaluation. These subtasks will be described briefly.

#### Diagnosis

- *Input:*
  - old design description,
  - new specification,

- *Output:*
  - list of candidates.

Assuming that the old specification does not satisfy the new specification, the first subtask is diagnosis. In this subtask it is determined which part of the old design can remain the same and which part is a candidate for adaptation. Beforehand, all the components of a design are candidates for adaptation and the goal of this subtask is to limit this number of candidates.

An example is the redesign of a radio. When the new specifications require a different transmission range, diagnosis should come up with the components concerned with the reception and not with the components concerned with the amplification.

#### Respecification

- *Input:*
  - list of candidates,
  - old design description.
- *Output:*
  - new subspecifications.

Diagnosis ends with a list of candidates for adaptation. For a candidate a new specification has to be derived. This subspecification must satisfy the following requirement. When it turns out that this specification can actually be designed in the next subtask, then the new design, consisting of the unaltered part plus the newly designed part, must satisfy the overall specification. In other words, the derived subspecification must bridge the gap between the old design and the new overall specification.

#### Design

- *Input:*
  - new subspecification.
- *Output:*
  - new partial design description.

As mentioned above, within redesign there is a design subtask. The first phase of redesign is concerned with the determination of a new specification for a part or component of the old design. Therefore the next step is a normal design step. A design solution for this newly derived specification has to be found. This can be accomplished by various methods:

- synthesis ('design from scratch'),
- case-based design,
- redesign,
- design by a human user.

In particular, redesign is an interesting possibility. There is a newly derived specification and an old design exists for the selected component. This means that one can restart the redesign process, but at the level of a component, assuming that a component in itself consists of a whole structure. This will become clearer in the next

section, where we present our modelling framework, which consists of different levels of abstraction.

### Evaluation

- **Input:**
  - new partial design description,
  - new specification,
  - old design description.
- **Output:**
  - judgement.

As the design subtask comes up with a new design for a part of the old design, the ultimate result has to be evaluated, that is, the old design with the newly incorporated part. Normally the part in itself will satisfy the specifications, but there is no guarantee that the same holds true for the whole design. Due to side effects, it is possible that the new design will not satisfy the overall specifications. This evaluation is no different from a ‘normal’ design evaluation, because two parts have been designed independently, and an evaluation is needed to decide whether the whole design satisfies the specification.

Both the design and evaluation subtasks are not specific for the redesign process. In the rest of this paper, we will not pay much attention to these subtasks, but concentrate on diagnosis and respecification: the design analysis.

Further, we not pay attention to the control structure for the subtasks mentioned. This structure can be straightforward; when a subtask fails, backtracking to the previous subtask seems appropriate.

However, first we will describe the design framework on which we base our work.

### 3. Design framework

Before formulating methods and techniques for redesign, it is first necessary to present our view on the description framework for the specifications and the design. We present an overview of *YMIR*, an ontology for engineering design, which has been developed within our group [7].

*YMIR* is based on systems theory/network models (see, for example, [8]) and it aims to describe both the design and the specification in a formal way.

*YMIR* consists of building blocks or components which can be used to compose a design. Such a building block is named a generic system model (GSM). It consists of the relation between so-called implicit and explicit variables. This relation expresses the behaviour of a GSM. These implicit and explicit variables are paired and are always the same for a specific domain, such as force/displacement, current/voltage, and torque/rotation. The relation between such a pair depends on so-called form-related variables. This means that there is

a formal relationship between the form and the behaviour of a component.

As an example, the behaviour of a beam consists of a relation between the displacement of the beam (explicit) and the force (implicit) that is exerted on the beam. The behaviour directly depends on form characteristics of the beam, namely its length, its material and its cross sectional area ( $F = (EA/l)d$ ). Thus form extends the ‘normal’ interpretation of geometrical form.

A function is a specific instantiation of the general behaviour of a design. The application of the beam, giving it a function, means that specific values for the form are acquired.

A collection of building blocks has to be defined for each specific domain, like resistors, capacitors and inductors for electrical engineering, or rods, beams and shafts for civil engineering constructions.

These collections form the primitive GSMs. They can be combined to form complex GSMs. When combining two GSMs, special boundary conditions for the connection apply. These are known as the ‘generalized Kirchhoff’s laws’, and they imply that in a connection the sum of the implicit variables equals zero and all explicit variables are equal. The behaviour of such a complex GSM is determined by the constituent GSMs. The structure of a design consists of the components and connections which are part of the design. Fig. 1 gives an example of a primitive GSM (a resistor) and a complex GSM (two connected resistors). The form-related variable  $R_1$  stands for the resistance of this component and relates the port  $(u_1, i_1)$  with the port  $(u_2, i_2)$ . A connection is possible between the ports of components by obeying the boundary conditions. Complex GSMs can be built recursively from complex and primitive GSMs.

A specification can be seen as a complex GSM, and in particular as a black box the content of which is still unknown. The design process boils down to the instantiation and/or decomposition of the black box until only primitives are obtained. A specification is, in other words, an underspecified design solution (Fig. 2).

In most domains primitives exist at different levels of abstraction. For instance, in the electrical domain there exists various types of amplifiers. At a lower level,

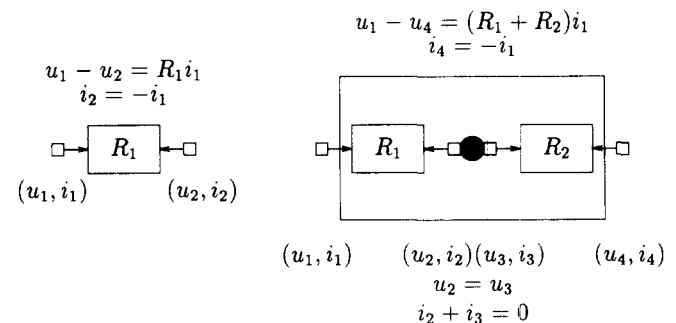


Fig. 1. Example of a primitive GSM and a complex GSM.

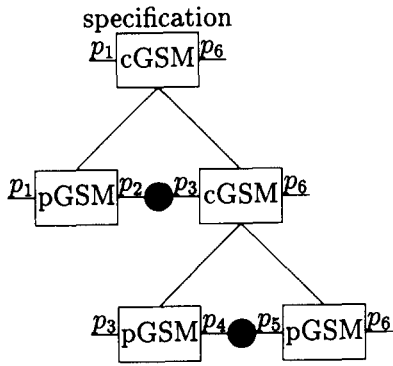


Fig. 2. Design process.

resistors, transistors etc., are considered primitive. In *YMIR* these different levels of abstraction are called technology layers. For each layer a separate set of primitives has to be defined. The design process starts at a high level of abstraction where the specification is satisfied with the primitives of that level. Then, a translation step has to be made to the next layer where the design process continues with the GSMs of that particular level. The design finishes when the lowest level of abstraction is reached. The translation implies that the design description at a high level is translated into a specification at a lower level.

The above-mentioned concepts deliver the basic tools for design. This is not sufficient for realistic design problems. Besides the building blocks, which represent 'first principles', there is a need to represent experiential design knowledge. This knowledge can be represented in the same *YMIR* framework. Experiential knowledge, represented in so-called prototypes, can be seen as complex GSMs, which are partially instantiated. Different types of bridges, like a suspension bridge, a truss bridge or a lift bridge, can be expressed as complex GSMs without full instantiation. In combination with a particular specification this will eventually result in a specific design, giving it a length, width, number of spans, etc.

One of the advantages of the *YMIR* ontology is its basis in systems theory. At the moment the formalization of *YMIR* is done within constraint logic. A formalization in Ontolingua [9] is considered. For more details regarding *YMIR* and its use, refer to [10].

### 3.1. Example of *YMIR*

In the forthcoming sections we address the redesign of an elevator. At this point we will present an initial design, thereby giving an extended example of *YMIR*.

Our design consists of ideal models from the mechanical domain. The elevator consists of a cabin, a contra-weight, a motor, a cable, and a pulley. The motor acts on the contra-weight. This example does not pretend to represent a 'real' elevator, but its complexity is sufficient for our purpose. Fig. 3 shows the model of our elevator.

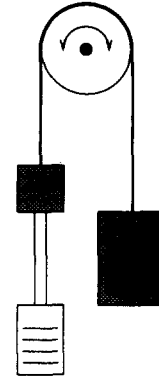


Fig. 3. Model of the elevator.

The elevator cabin and the contra-weight appear as ideal masses  $M_e, M_c$ , the force source (gravity) which acts upon them ( $G_e, G_c$ ), and the friction, represented by ideal dampers  $B_e, B_c$ . The cable consists of an ideal spring and an ideal damper and it is separated into a left-hand part ( $K_l, B_l$ ) and a right-hand part ( $K_r, B_r$ ). The pulley  $P$  is an ideal transformer; it inverts the direction of the velocity. The motor is a velocity source  $V$  which acts upon the contra-weight, although we do not explain the way in which this is done. The structure of the design can be seen in Fig. 4.

As we do not pretend to represent a 'real' elevator, we can make an extra simplification. All components are considered to be independent. We look upon the mass, gravity and friction as independent elements, although they are properties of the same object. The same holds for the cable, which consists of a damper and a spring.

This model is accompanied by a system of equations that express the behaviour of the components and the boundary conditions for the connections:

*Component expressions:*

$$\begin{aligned}
 V_{P_2} &= P * V_{P_1}, p = -1 \text{ (pulley)} \\
 F_{P_2} &= -F_{P_1} \\
 F_{B_e} &= B_e * V_{B_e}, B_e = 4 \text{ (dampers)} \\
 F_{B_c} &= B_c * V_{B_c}, B_c = 15 \\
 F_{B_{l1}} &= B_l * (V_{B_{l1}} - V_{B_{l2}}), B_l = 4 \\
 F_{B_{l2}} &= -F_{B_{l1}} \\
 F_{B_{r1}} &= B_r * (V_{B_{r1}} - V_{B_{r2}}), B_r = 2 \\
 F_{B_{r2}} &= -F_{B_{r1}} \\
 F_{K_{l1}} &= K_l * (V_{K_{l1}} - V_{K_{l2}})/s, K_l = 3 \text{ (springs)} \\
 F_{K_{l2}} &= -F_{K_{l1}} \\
 F_{K_{r1}} &= K_r * (V_{K_{r1}} - V_{K_{r2}})/s, K_r = 3 \\
 F_{K_{r2}} &= -F_{K_{r1}} \\
 F_{M_e} &= M_e * V_{M_e} * s, M_e = 12 \text{ (masses)} \\
 F_{M_c} &= M_c * V_{M_c} * s, M_c = 5
 \end{aligned}$$

*External sources:*

$$\begin{aligned}
 F_{G_c} &= G_c/s, G_c = 50 \text{ (gravity)} \\
 F_{G_e} &= G_e/s, G_e = 120 \\
 V_v &= v \text{ (v source)}
 \end{aligned}$$

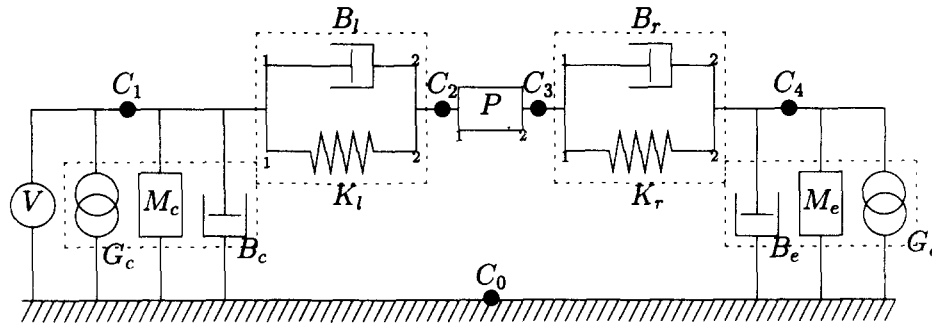


Fig. 4. Design model of the elevator.

**Boundary conditions:**

$$\begin{aligned}
 F_v + F_{M_c} + F_{G_c} + F_{B_c} + F_{B_{l1}} + F_{K_{l1}} &= 0 \\
 V_v = V_{M_c} = V_{G_c} = V_{B_c} = V_{B_{l1}} = V_{K_{l1}} \\
 F_{B_{l2}} + F_{K_{l2}} + F_{P_1} &= 0 \\
 V_{B_{l2}} = V_{K_{l2}} = V_{P_1} \\
 F_{P_2} + F_{B_{r1}} + F_{K_{r1}} &= 0 \\
 V_{P_2} = V_{B_{r1}} = V_{K_{r1}} \\
 F_{M_e} + F_{G_e} + F_{B_e} + F_{B_{r2}} + F_{K_{r2}} &= 0 \\
 V_{M_e} = V_{G_e} = V_{B_e} = V_{B_{r2}} = V_{K_{r2}}
 \end{aligned}$$

The values of the from variables are arbitrary. All equations are expressed after Laplace transformation, resulting in a system of algebraic equations. The general behaviour of the ideal components is quite simple. For a mass,  $f = ma$  holds. This results in  $F = MVs$  in the  $s$  domain. For a damper,  $f = bv$  results in  $F = BV$ , and, for a spring  $f = kd$  results in  $F = KV/s$ .

The velocity of the system is not specified. This can be a very complex function, such as the one shown in Fig. 5, which is composed from a series of step functions and sinusoidal functions. An alternative is to give the motor a constant velocity. The reason for not specifying  $V$  will be made clear later on.

This system contains the instantiated behaviour of the design. The function of the system, being the velocity of the cabin, can be derived:

$$V_{M_e} = -\frac{8vs^2 + 18vs + 240s + 9v}{24s^3 + 38s^2 + 18s + 9}$$

In other words, given this specification, the presented model is a design solution.

**4. Analysis**

As explained, the important subtasks of the redesign process are diagnosis and respecification. For both we will outline a method to achieve the desired outcome. The method will be illustrated with the elevator example of the previous section.

First we will distinguish between different types of redesign. These types are distinguished in the literature, e.g. [6], and they fit neatly into our design framework.

We make this distinction because we assume that these different types of redesign correspond to differences in the complexity of the adaptation.

- *Parametric redesign:* This type implies the adaptation of the form-related variables in a component (GSM). The general functionality remains equal, except that a different instantiation is searched for, for example the change of a resistor value from  $20\ \Omega$  to  $50\ \Omega$ . We assume that a parametric change in the specification leads to a new design which is easy to obtain. This holds only within certain ranges. The change of a specification from  $20\ \Omega$  to  $20\ M\Omega$  will probably lead to a totally different design.
- *Component redesign:* This type occurs when one component in the design is replaced by another component with a different behaviour. In terms of *YMIR* this means the replacement of one type of GSM with another. An example is the replacement of a resistor by a capacitor. We assume that this type of redesign will be more complex than parametric redesign.
- *Structure redesign:* the third type deals with the altering of the structure: the addition, deletion or movement of components within the design. An example is the addition of a buffer to an elevator in case the cable breaks. We assume that this is the most complex type of redesign.

This tripartite method is a heuristic one. There is no guarantee that the actual design of a parametric change will be easier than a structure adaptation. The three types only refer to a difference between the old and new specification, and not to the actual design solution.

The method we describe concentrates on component adaptation, thereby including parametric adaptation. Possibilities of structure redesign and the specific use of parameters are postponed until the next section.

**4.1. Diagnosis**

As mentioned above, diagnosis concerns the localization of components which can be responsible for the discrepancy between the old design and the new specification. The result has to be a set of components which are candidates for redesign and a set of components

which can be kept the same. In other words, there is a conflict between the new specification and the old design, and the task of diagnosis is to restrict the set of candidates as much as possible.

To accomplish this task we want to use techniques from model-based diagnosis (MBD). There is an analogy between 'standard' MBD techniques and the diagnosis of a design in our formalism. By regarding our form of diagnosis as analogous to MBD, we are able to use the well understood techniques from MBD research.

Although there are numerous variants of MBD, there seems to be a consensus about what is considered standard. To explain our use of MBD, we will first very briefly point out this standard view, based on [11] and [12]. Extensive treatment of MBD techniques can be found in [13].

The purpose of MBD is to diagnose the malfunctioning of a certain device. This is accomplished by reasoning about a model of the device with the use of measurements and tests carried out within the 'real' device. The model has available the knowledge of the desired behaviour, possibly extended with knowledge about the faulty behaviour.

The measured values may deviate from the values are predicted by the model. A conflict is a set of components which cannot all be correct. One or more components in a conflict explain the deviating values. These components form a diagnosis of the problem. To discriminate between the hypotheses, additional observations are required. MBD offers methods for probe advice and updates its hypotheses when additional data from tests become available. The iteration of diagnosis and measurements continues until a diagnosis consisting of one or more components is found. Then further action is possible, like the replacement or repair of the malfunctioning component(s).

In [14], we have described how to apply model-based reasoning techniques for the diagnostic problem in redesign. The redesign diagnostic problem looks similar but differs slightly from MBD. The deviation in our case is between the new specification and the old design. The role of the diagnosis model is played by the old design, while the new specifications play the role of measured values, which differ from the desired behaviour. Note that it seems confusing that the new specifications are desired in terms of design, but they play the role of undesired behaviour in terms of diagnosis.

In general, MBD is concerned with numerical values. This is not always the case in design, as we often deal with symbolic mathematical expressions. This difference can have an influence on the complexity of the diagnosis process.

The result of a diagnosis process will often consist of several hypotheses. A further distinction, based on new tests, is, however, not possible. The counterpart of the real device of MBD consists only of the new

specifications. Measurements within the device are not possible. It would mean measuring within the new design, which does not exist.

Therefore, the diagnosis task consists of only one step, instead of a sequence of a diagnosis/test steps for further reduction. This is not as harmful as it seems. In MBD, the set of components that is 'broken' is searched. The replacement of components which function correctly is unwanted. In redesign it is possible to acquire a new solution in different ways. There is no strict separation between 'broken' and 'unbroken' components. Every solution which satisfies the specifications is considered a correct solution. Only 'better' solutions are preferred to 'worse' solutions. Whenever extra knowledge is available, this can be used to select between the possible solutions.

The differences between 'normal' MBD and the way we use it are summarized in Table 1.

#### 4.2. Example

We will now apply diagnosis to the example of the elevator from the previous section. Therefore we transform our design into an MBD model.

We are interested in the relation between the velocity  $V$  of the motor and the velocity  $V_e$  of the elevator cabin. This is expressed as a mathematical expression, but it will be more clear if we look at the graphs of both functions (see Fig. 5).

The velocity of the cabin is smoothed by the system of springs and dampers. A new specification could require a faster damping. As mentioned, we assume that it is possible to formulate a new specification in a mathematical expression.

We have the *YMIR* description of the elevator at our disposal. The diagnosis starts with finding the components that determine the velocity of the cabin, given the velocity of the motor. Therefore, we alter our model to a new model with the implicit (force) and explicit (velocity) variables strictly separated (see Fig. 6). In a normal component there is a relation between the implicit and explicit variables. Therefore, it is represented as one box. Within sources and connections there is no such relation. Therefore, they have separated boxes for  $f$  and  $v$ .

Table 1  
Model-based reasoning

Diagnosis	Redesign
Difference between predictions and observations	Difference between old and new specification
Numerical values	Symbolic values
Further tests	No further tests possible
One solution	More solutions

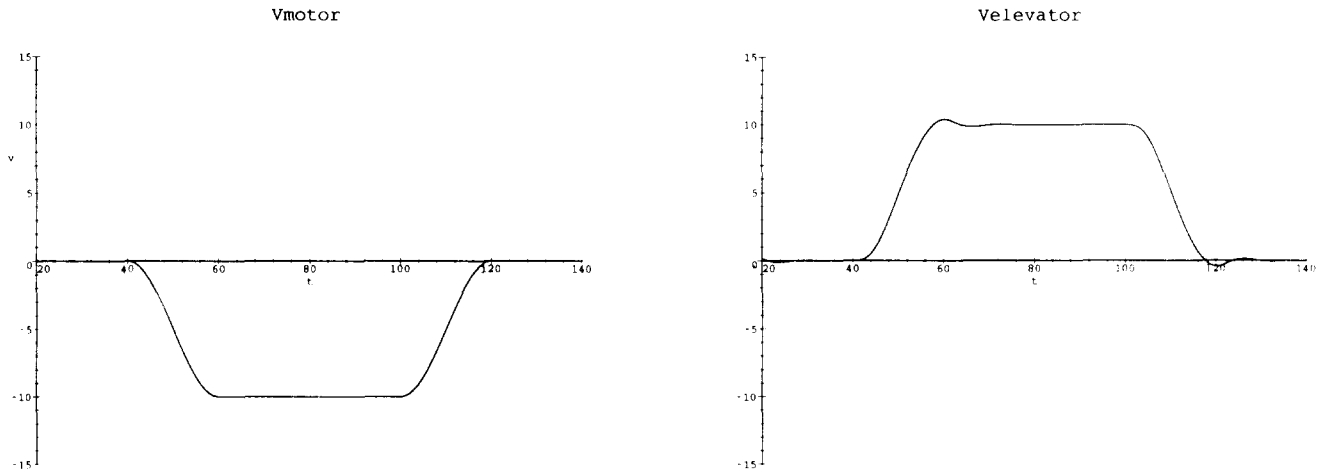


Fig. 5. Graphs of  $V_{motor}$  and  $V_{levator}$ .

The next step is the assignment of a direction between all links. By assigning direction we can construct a set of predecessors, making MBD possible. The assignment of direction does not imply a physical direction; it only gives a direction for calculation. This assignment is done by applying the following steps:

- **Step 1:** Assign a direction to the sources. The  $v$  of a velocity source is outgoing, and the  $f$  is incoming. For a force source, the opposite holds.
- **Step 2:** Determine the consequences of the assignment.
  - In a  $v$  connection one incoming link means that the rest can be derived and are therefore outgoing. This holds because  $v_i = v_j$ .
  - In an  $f$  connection one outgoing link means that the rest are incoming, because  $\sum f_i = 0$ . The outgoing link is determined by the other incoming links.
  - Within a 2-port component there are two possibilities:
    - One  $f$  link and one  $v$  link are incoming; they determine the rest.
    - Two incoming  $v$  links determine the  $f$  links.
  - Within a 1-port component (earth) one link is incoming and one link is outgoing.

The consequences are propagated until no further assignment is possible.

- **Step 3:** Whenever there are undirected links, assume a 'smart' direction and return to Step 2. 'Smart' means the most informative: an incoming link in a  $v$  connection or an outgoing link in an  $f$  connection. Otherwise stop.

This results in a new model (see Fig. 7). These steps are not specific for force and velocity. For other parts of implicit/explicit variables, the same steps can be applied.

The final step is to pass through the network, thereby collecting the components that are necessary for determining  $V_e$ . This can be seen in Fig. 7. The broken lines are passed through. It is not necessary to collect the connections. As we assume no structure adaptation, it is not possible to alter a connection. The particular diagnostic engine that we used here is Focused Sherlock [15]. This engine works in an iterative way. At first the old system is checked by comparing its derived behaviour with the new specifications. As a result a conflict is detected:

$$\langle V, K_l, B_l, P, K_r, B_r, G_e, M_e, B_e \rangle$$

All elements in this conflict are potential redesign solutions. Each solution can now be checked by replacing its function by a yet unknown function. We will discuss this in more detail in Section 4.3. As a result, here, all potential solutions turn out to be feasible from a mathematical point of view, and thus the diagnostic process halts.

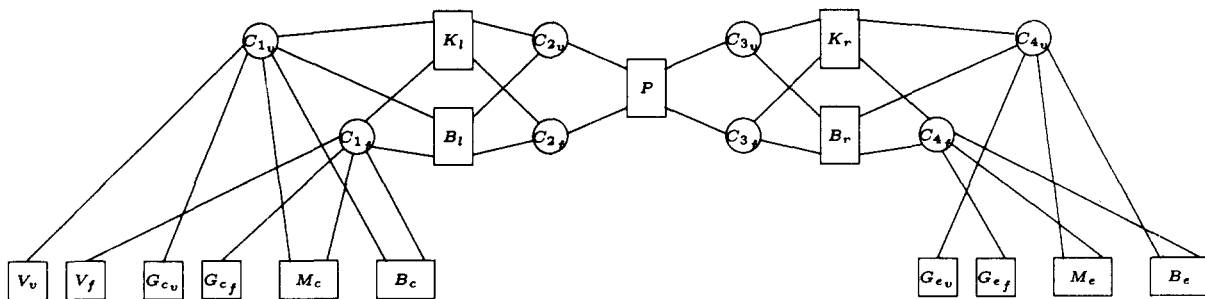


Fig. 6. Model without direction.

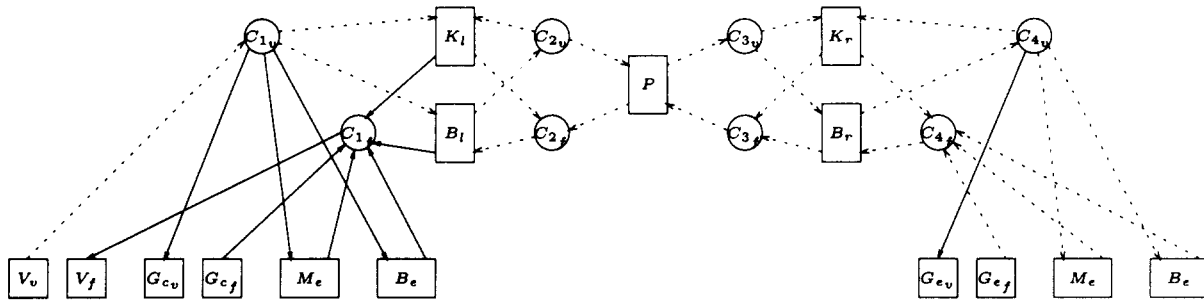


Fig. 7. Model with direction.

That is the conflict that is derived. Every element of this set forms a minimal diagnosis. That is, each element is in itself an explanation of the difference between the old design and the new specifications. Whether redesign is really possible has to be determined.

The choices that were made in Step 3 have no consequences for this set. Every other choice would lead to the same set of components. Within *this* structure it is not possible to alter one of the components outside the conflict to fulfil a change in the specification.

It has to be noted that the new specification itself is not necessary to accomplish this result. The only thing that has to be known is which relation has to be altered, in this case the relation between  $V_e$  and  $V_v$ . The precise contents of the required specification will be addressed in the next subsection, when a new specification has to be derived.

### 4.3. Respecification

We have derived a set of components the elements of which can each be a diagnosis. Now we have to derive new specifications to find new solutions for these components. Several strategies are possible at this point. When we have extra knowledge at our disposal, we could use it for the selection of a suitable diagnosis. For example, we may prefer a spring adaptation to a damper adaptation. Second, we may choose to derive new specifications for all components and then select one for the actual design. Third, we may choose one component, do the respecification and then try to design a new one.

The second method is preferred as the new specifications can differ in complexity and it seems to be a good heuristic to choose simple specifications over complex ones. When the set of candidates is very large, however, this benefit may no longer hold.

Our method for deriving a new specification consists of two steps. Firstly, the old specification of the selected component is eliminated from the whole design description, and replaced by an 'empty' specification, thereby not altering the structure. Such an empty specification is a black box the content of which is yet unknown. The next step is to solve the combined expression, consisting

of the new specification and the old design for the unknown component.

We<sup>1</sup> will show how this works after constructing a new specification in an artificial way.

We modify  $B_1$ 's parameter from 4 to 5. This gives a slightly different specification. We do not take into account the precise motor function and take  $v$  as input. We solve this system and we get the following specification for the cabin:

$$V_e = -\frac{10vs^2 + 21vs + 360s + 9v}{36s^3 + 55s^2 + 21s + 9}$$

The conflict given above delivers the diagnoses. In the real world, we would derive specifications for all diagnoses, selecting a specification afterwards. In this example, we start with  $B_1$  because we know the desired outcome. Afterwards we will compute some alternatives.

We replace  $F_{B_{11}} = 4 * (V_{B_{11}} - V_{B_{12}})$  with  $F_{B_{11}} = X * (V_{B_{11}} - V_{B_{12}})$  and try to determine the value for  $X$ . The force/velocity relation remains the same, as we demand that the structure be unchanged. The expression for  $V_e$  is

$$V_e = -\frac{2vs^2X + 6vs + 3vsX - 240s + 120sX + 9v}{-24s^3 + 12s^3X - 30s^2 + 17s^2X + 6s + 3Xs + 9}$$

We acquire the solution by equalizing the new specification and the old one, adapted by inserting  $X$ . This results in the equation

$$\begin{aligned} & -\frac{10vs^2 + 21vs + 360s + 9v}{36s^3 + 55s^2 + 21s + 9} \\ & = -\frac{2vs^2X + 6vs + 3vsX - 240s + 120sX + 9v}{-24s^3 + 12s^3X - 30s^2 + 17s^2X + 6s + 3Xs + 9} \end{aligned}$$

Now we find our solution by solving the expression for  $X$ . As expected we find that  $X = 5$ . We have derived a new specification for the damper:  $F_{B_{11}} = 5 * (V_{B_{11}} - V_{B_{12}})$ . It can be seen that this result is independent of the specific velocity of the motor ( $V_v$ ). Thus it turns out to be possible to neglect this specific function, as long as respecifying  $V_v$  is not the aim.

<sup>1</sup> Whenever 'we' is used in combination with a mathematical process, 'we' means Maple.



Now we will try  $K_1$ . The ‘empty’ specification becomes  $F_{K_{11}} = X * (V_{K_{11}} - V_{K_{12}})$  instead of  $F_{K_{11}} = 3 * (V_{K_{11}} - V_{K_{12}})/s$ . Note the disappearance of the  $s$  term. We do not require that the new specification matches an ideal spring; we only require the relation between  $F_{K_{11}}$ ,  $V_{K_{11}}$  and  $V_{K_{12}}$ .

Again we obtain an expression containing  $X$ :

$$V_e = \frac{8vs^2 + 2s^2vX + 12vs + 3vsX + 240s + 120sX - 360}{24s^3 + 12s^3X + 2s^2 + 17s^2X - 33s + 3sX}$$

Equalizing it to the new specification and solving it for  $X$  leads to  $X = 1 + 3/s$ .

This is the new specification for  $K_1$ . It is the design subtask which must find out how to accomplish this specification. As a side step, we will look at the consequences for the design subtask. For this example, it turns out that replacing the spring with a parallel spring/damper gives the required result. The 1 is the difference between the old damper and the new specification, and the  $3/s$  relates to the old spring. Thus, in the design subtask, we would replace a component with a new structure, but that is not the same as structure redesign.

The difference is curious. The immediate change of structure by adding an extra damper is considered structure adaptation, although there is no essential difference from the case of replacing the spring by the combination of a spring and damper (see Fig. 8). Our method does not derive specifications for structure adaptation, but the newly derived specifications can be satisfied by a new structure.

The benefit of determining all specifications beforehand, before selecting a component, becomes clear in the next example, when we try to respecify  $B_e$ . This leads to the following value for  $X$ :

$$X = \frac{3(16vs^5 + 168vs^4 + 306vs^3 + 135vs^2 + 3440s^3 - 480s^2 - 360s)}{(10vs^2 + 21vs + 360s + 9v)}$$

This new specification seems very difficult to design. As mentioned before, it is preferable to derive specification for all diagnoses and select the least complex ones.

The respecification subtasks deliver new specifications for the diagnoses. As explained in the design subtask the actual design has still to take place. This can be done, as explained, in several ways, by design from scratch, by case-based design, by redesign, or by a human designer. When it is possible to come up with a design solution, the new design has to be evaluated by checking whether the new design really satisfies the desired specification.

## 5. Extensions

The previous section described a basic method for the redesign of an old design. This section will present some

extension, which could mean an improvement of the presented ideas. First we will concentrate on parametric redesign, and then we will pay attention to structure redesign.

### 5.1. Parametric redesign

In our example of the previous section, it became clear that we did not use all the information. The respecification of the spring  $K_1$  did not demand that the new specification was similar in structure. The result was a new specification which could be designed by a new structure.

If we had used the extra demand of keeping the structure of the specification the same, then it would not be possible to come up with a new specification. A new specification can only be derived when it turns out that there is an exact match between the old and new structure, as in the case of the adaptation of  $B_1$ .

One may consider first checking whether a parametric adaptation is possible before checking for a component adaptation. This is done with the assumption in mind that a parametric adaptation is probably easier to obtain than a component adaptation.

This idea can be extended to more components at the same time. We may try to find a new instantiation for all parameters at the same time. In this way we would also check the possibility of the parametric adaptation of more components.

We formulate a new specification by changing some parameters. The new parameters are  $B_e = 11, K_1 = 4, B_1 = 5, B_c = 7, G_c = 80, M_c = 8$

$$V_e = -\frac{10s^2v + 23sv + 360s + 12v + 120}{36s^3 + 55s^2 + 34s + 12}$$

We first have to derive the uninstantiated expression for  $V_e$ :

$$V_e = -\frac{(-B_1PB_r)s^2v + (-PB_1K_r - PB_rK_1)sv}{(PB_rM_e + B_1M_e)s^3} + \frac{(B_1G_e + PB_rG_e)s + (-PK_rK_1)v}{(B_1B_r + PK_rM_e + B_1B_e + K_1M_e + PB_rB_e)s^2} + \frac{(K_1G_e + PK_rG_e)}{+(PK_rB_e + K_1B_e + K_1B_r + B_1K_r)s + (K_1K_r)}$$

A pure parametric adaptation is only possible if the structure and the type of components remain the same. A solution can be found by matching all the variables from the above expression with the parameters from the new specification. This results in the following system:

$$\begin{aligned} -B_1PB_r &= 10 \\ -PB_1K_r - PB_rK_1 &= 23 \\ B_1G_e + PB_rG_e &= 360 \\ -PK_rK_1 &= 12 \\ K_1G_e + PK_rG_e &= 120 \end{aligned}$$

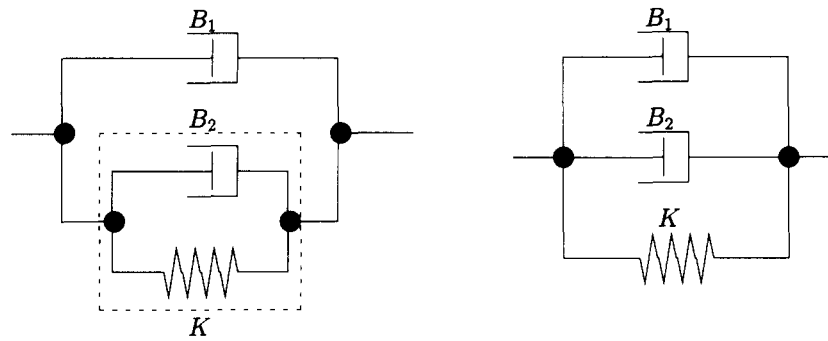


Fig. 8. Difference between component and structure adaptation.

$$PB_r M_e + B_1 M_e = 36$$

$$B_1 B_r + PK_r M_e + B_1 B_e + K_1 M_e + PB_r B_e = 55$$

$$PK_r B_e + K_1 B_e + K_1 B_r + B_1 K_r = 34$$

$$K_1 K_r = 12$$

The solution corresponding to the changes in the parameters is found:  $B_e = 11, P = -1, B_r = 2, K_1 = 4, B_1 = 5, G_e = 120, K_r = 3, M_e = 12$ .

The parameters that are not derived can remain the same. When more instantiations are found, the one closest to the old solution is preferred, as this means the least adaptation. If no solution can be derived, then the other method is tried, resulting in component adaptation.

### 5.2. Structure

The idea of structure redesign is somehow contrary to the general philosophy behind redesign, which is to adapt an old case with minimal effort.

If structure redesign is introduced, lots of adaptations are possible, and it is hardly possible to distinguish interesting redesigns from uninteresting ones. We have mentioned one solution in Section 4.3. In component adaptation we have postponed structure adaptation until the actual design phase.

This is not completely satisfying. There seems to be special structure adaptations which are interesting and we want to incorporate these in the redesign process.

A simple example is the redesign of a bridge, in the case in which the length has to be increased. In addition to changing the thickness of the piers (parametric redesign), or using another type of pier (component redesign), a structure redesign is desirable, for example changing the number of piers (see Fig. 9).

Guidance for structure adaptation can be found in the idea that certain components are repeated in a design. Therefore interesting structure adaptations could be additional repetitions of the same component. To accomplish this, we could parameterize the structure by taking into account the number of components (e.g. piers). To comply with our formal view on design, it is,

however, necessary that this extra structure parameter should be related to the behaviour. As an example we show the expressions for serial and parallel resistors (Fig. 10).

Therefore, given a parallel structure of five resistors with resistances of  $90 \Omega$ , resulting in  $u_1 - u_2 = 18i_1$  and a new specification of  $u_1 - u_2 = 15i_1$ , a simple parametric adaptation is possible: taking an  $R$  of  $75$ , as  $75/5 = 15$ . The structure adaptation results in an extra resistor, as  $90/6 = 15$ .

This seems possible in simple cases, but it is an open question as to what will happen in complex cases when a whole structure is repeated.

## 6. Conclusions

We have presented our view on the redesign process. We have stated that redesign is an important research subject as design is often, in fact, redesign.

The crux of redesign lies in two subtasks: diagnosis and respecification. We have presented a method for both tasks that is inspired by MBD and by the use of our design framework *YMIR*. In addition, we have presented extensions to our method that are related to different types of redesign. Our method worked for the presented example of the redesign of an elevator. This gives no guarantee for the extension to 'real' elevators or even any technical system. The complexity of the diagnosis algorithm itself is similar to Focused Sherlock [15]. The use of symbolic expressions can be a hindrance. Refer to [16] for possible solutions.

This work is embedded in different projects. Within our group much work has been done on MBD [17]. Another project focuses on engineering design, one aspect of which concerns modeling the product related knowledge. One of the results of this part of the project is the *YMIR* ontology [7]. Ongoing research concentrates

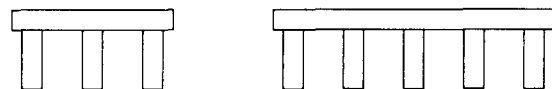


Fig. 9. Redesign of a bridge.

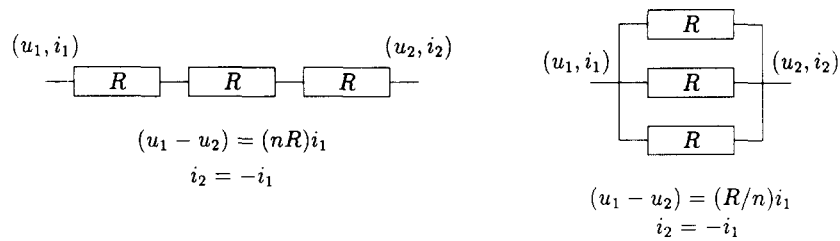


Fig. 10. Resistors in series and in parallel.

on various aspects of design, such as redesign, case-based reasoning and process modeling. The third embedding is in the *REVISE* project, which concentrates on different aspects of redesign: not only technical redesign, but also the redesign of control structures, simulation software and knowledge-based systems. One of the overall goals of this project is to formulate an ontology for redesign.

Our future work will concentrate on the following points:

- We want to develop the example further. We wish to perform redesign on a more realistic elevator. We will base this on the VT domain [18,19], which was used in Sisyphus-II [20]. This does not mean that we will follow the specific VT approach with fixes. The VT domain will be more a source of inspiration.
- A related issue is the introduction of more technology layers, as described in *YMIR*, and their influence on the redesign process.
- We have not discussed the control structure of the redesign process. We want to integrate all subtasks into one model with a matching control structure.
- We will address efficiency questions. The described parametric variant can provide a more efficient method, but it has to be determined under which condition this holds.
- Finally, we want to incorporate specific adaptation knowledge. The idea that repeating structures are interesting is an example of adaptation knowledge. Such knowledge can be used to guide the redesign process. We have to collect such knowledge, both general and domain dependent.

### Acknowledgements

This research is part of the *STEVIN* project, which is a research program aimed at supporting the design of technical systems at the Knowledge-Based Systems Group at the University of Twente, Netherlands.

The investigations were supported by the Netherlands Computer Science Research Foundation (*SION*) with financial support from the Netherlands Organization for Scientific Research (*NWO*), as Project 612-322-316.

(Evolutionary Design in Knowledge-Based Systems), the *REVISE* project.

### References

- [1] J. Mostow, Design by derivational analogy: issues in the automated replay of design plans, *Artificial Intelligence*, 40 (1989) 119–184.
- [2] A. Goel and B. Chandrasekaran, Functional representation of designs and redesign problem solving, in *Proc. IJCAI'89: 4th Int. Joint Conf. on AI*, 1989, pp. 1388–1394.
- [3] E. Stroulia, M. Shankar and A. Goel, A model-based approach to blame-assignment in design, in J.S. Gero (ed.) *Proc. AID '92: 2nd Int. Conf. on AI in Design*, 1992, pp. 519–537.
- [4] B. Chandrasekaran, Design problem solving: a task analysis, *AI Magazine*, 4 (1990) 59–71.
- [5] F. Tempelman, P.M. Wognum and N.J.I. Mars, Case-based reasoning in literature and Stevin, *Memorandum UT-KBS-94-12: Technical Report*, Department of Computer Science, University of Twente, Netherlands, 1994.
- [6] J.R. Dixon, M.J. Guenette, R.K. Irani, E.H. Nielsen, M.F. Orelup and R.V. Welch, Computer-based models of design processes: the evaluation of designs for redesign, in *Preprints NSF Engineering Design Research Conf., University of Massachusetts, USA, 1989*, pp. 491–506.
- [7] L.K. Alberts, *YMIR: An ontology for engineering design*, PhD Thesis, University of Twente, Netherlands, 1993.
- [8] H.R. Martens and D.R. Allen, *Introduction to Systems Theory*, Charles E Merrill Publishing, USA, 1969.
- [9] T.R. Gruber, Ontolingua: a mechanism to support portable ontologies, *Technical Report KSL 91-66*, Knowledge Systems Laboratory, Stanford University, USA, 1992.
- [10] L.K. Alberts and F. Dikker, Integrating standards and synthesis knowledge using the YMIR ontology, in J.S. Gero (ed.), *Proc. AID '94: Third Int. Conf. Artificial Intelligence in Design, August 15–18, 1994, Lausanne, Switzerland*, pp. 517–534.
- [11] J. de Kleer and B.C. Williams, Diagnosing multiple faults, *Artificial Intelligence*, 32 (1987) 97–130.
- [12] R. Davis and W. Hamscher, Model-based reasoning: troubleshooting, in H.E. Shrobe (ed.), *Exploring Artificial Intelligence*, Morgan Kaufmann, 1988, pp. 297–346.
- [13] W. Hamscher, L. Console and J. de Kleer, *Readings in Model-Based Diagnosis*, Morgan Kaufmann, USA, 1992.
- [14] R.R. Bakker, S.J.M. van Eldonk, P.M. Wognum and N.J.I. Mars, The use of model-based diagnosis in redesign, in *Proc. ECAI '94*, John Wiley, 1994, pp. 647–651.
- [15] J. de Kleer, Focusing on probable diagnoses, in *Proc. 9th National Conf. on AI*, MIT Press, USA 1991, pp. 842–848.
- [16] L.K. Alberts, R.R. Bakker, D. Beekman and P.M. Wognum, Model-based redesign of technical systems, *Memorandum UT-KBS-92-45*, University of Twente, Netherlands, 1992.
- [17] R.R. Bakker, P.C.A. van den Bempt, N.J.I. Mars, D.-J. Out and

- D.C. van Soest, Issues in practical model-based diagnosis, *Future Generation Computer Systems*, 9 (1993) 329–337.
- [18] S. Marcus, J. Stout and J. McDermott, VT: an expert elevator design that uses knowledge-based backtracking, *AI Magazine*, Spring (1988) 95–111.
- [19] G. Yost, Configuring elevator systems, *Technical Report*, Digital Equipment Corporation, 1992.
- [20] A.T. Schreiber and W.P. Birmingham (eds.), *Proc. KAW '94: Knowledge Acquisition Workshop, Banff, Canada, 1994*.