Controlling Qualitative Resolution

Jean-Luc Dormoy
Electricité de France Research Center
Clamart, France

LAFORIA - Pierre & Marie Curie Paris University

Abstract

We proposed earlier in [Dormoy & Raiman, 1988] a new way of reasoning about a device, we called "Assembling a Device". Starting from a component description (namely confluences), the qualitative resolution rule provides task-oriented global relations which link the physical quantities involved in a device to some selected reference variables. This rule is complete: given any task to be performed (simulation, postdiction,...), it discovers an assemblage, i.e. a set of relations reducing the task to a straightforward propagation. We might thus expect to apply qualitative reasoning to large-scale systems.
Unfortunately, the number of potential applications of the resolution rule is likely to increase exponentially as it is being fired. This behavior has to be related to the NP-completeness of the problem which consists of solving a set of confluences. In this paper, we present a heuristic for controlling the resolution rule, i.e. for choosing between its potential applications, and a collection of simple rules for speeding it up. This heuristic has a combinatorial form, but it is based on a simple commonsense idea. At the same time, it is borne out by mathematical results. Theoretically, a qualitative model can be out of its scope, but we have not yet hit upon a physical system with this kind of pathology.

1 Introduction

In [Dormoy & Raiman, 1988], we proposed a new way of reasoning about a device, called "Assembling a Device". Starting from a component description (namely confluences), the qualitative resolution rule provides task-oriented global relations which link the physical quantities involved in a device to some selected reference variables. This rule is complete: given any task to be performed (simulation, postdiction,...), it discovers an assemblage, i.e. a set of relations reducing the task to a straightforward propagation. We might thus expect to apply qualitative reasoning to large-scale systems. **All this is developed in detail in the above mentioned paper [Dormoy & Raiman, 1988], and we strongly recommend that the reader consult it before reading the following.**
Solving a set of confluences turns out to be an NP-complete problem [Dormoy, 1987]. Hence, the number of potential applications of the resolution rule is likely to increase exponentially as it is being fired. In practical terms, combinatorial explosion happens even when dealing with very simple models.
In the second section, we present a heuristic, which we call the "joining rule", for controlling the resolution rule, i.e. for choosing between its potential applications. It is based on the simple commonsense idea of consolidation [Bylander, 1987]. At the same time, it is borne out by mathematical results. In theory, a qualitative model may be out of the scope of this heuristic. We justify why we have not yet hit upon a physical system with this kind of pathology.
Though the joining heuristic is self-sufficient, some rules can be added to the basic machinery in order to speed up the assembling step. We present them in the third section and we show how the whole system works through the use of a simple example.
In conclusion, we think that the assembling technique, controlled by the joining heuristic, can assemble large artefacts. We are currently working on a model for a large-scale plant.

2 The joining rule

2.1 Consolidation

Consider a component-based model of a device, and let C_1, C_2 and C_3 be three mutually interacting components. If we denote C_{12} the compound component $C_{12}=\{C_1,C_2\}$, the interactions between C_1 and C_2 define how C_{12} works. Indeed, they are of no interest to C_3: from C_3's point of view, the set made up of C_1 and C_2 is equivalent to C_{12}. C_3 cannot distinguish C_1 and C_2 from each other. Hence, it should be possible to draw a model of C_{12} from models of C_1 and C_2 regarded by C_3 as equivalent. Joining local models together in order to provide more global ones is what has been called consolidation [Bylander, 1987]. The problem lies in giving concrete expression to this idea. In particular, certain rules must be stated as regards to the selection of the pair of components to be consolidated at each inference step: the pair certainly cannot be randomly selected.

2.2 The resolution rule under the microscope

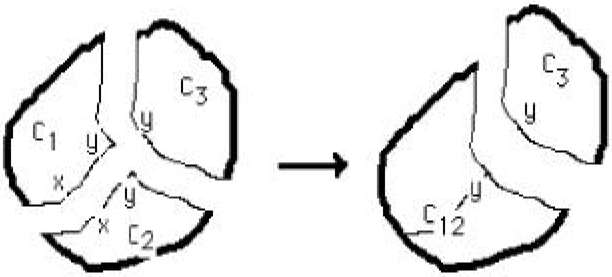


Figure 1: Joining two components

In a confluence-based model, C_1 and C_2 interact through their common variables. Hence, building a model for C_{12} means providing confluences by eliminating them. Consider a variable involved in both C_1 and C_2 models. If it is involved in some other component model, then it must appear in a model of C_{12} (like variable y in Fig. 1). But if it is not, then it must be completely eliminated (like variable x in Fig. 1).

The resolution rule (Fig. 2) seems to tackle this problem, but we must examine what it accomplishes closely.

```
Let x, y, z, a, b be qualitative
quantities such that
     x + y - a
and -x + z - b
If x is different from ?, then
     y + z - a + b
```

Figure 2: The qualitative resolution rule

Consider a simple case (but this case happens more often than not), when both C_1 and C_2 models are made up of a single confluence, say respectively E_1 and E_2. Let x be a variable involved in both, and assume that the resolution rule applies to E_1 and E_2 and so eliminates x. Then we get a new confluence, say E_{12}, which is global to C_{12}. Any other variable involved in E_1 or E_2, or both, will belong to E_{12} as well. But it is not true in general that E_{12} is equivalent to E_1 and E_2. For instance, if the model of another component C_3 mentions x explicitely, then E_{12} is certainly not a proper model of C_{12}. But the equivalence should hold whenever x is exclusively involved in E_1 and E_2.

2.3 Joining two components

Previous remarks provide a heuristic rule for choosing between the potential applications of the resolution rule at each inference step:

Joining rule: Let E be a set of confluences corresponding to a component-based description of a device.

> If the resolution rule applies to confluences E_1 and $E_2$by eliminating variable x, and if x is exclusively involved in E_1 and E_2, then choose this application. An equivalent model (as far as variables different from x are concerned) is obtained by substituting confluence E_{12} produced in this way for confluences E_1 and E_2.

If E_1 and E_2 are the respective models of components C_1 and C_2, then E_{12} is a proper model for C_{12}. C_1 and C_2 are joined.

This rule can be applied recursively. Indeed, a variable y different from x and involved solely in E_1, E_2 and a third confluence belongs to exactly two confluences after the joining rule has been fired. Therefore, the joining rule might choose to eliminate it at a next step. This means that a compound component can be joined in turn to another atomic or equally compound component.

2.4 A mathematical justification

The choice heuristic contained in the joining rule conditions has been justified above by some commonsense arguments. It needs no mathematical proof. But the conclusion, which claims that substituting E_{12} for E_1 and E_2 provides an equivalent model for the variables different from x, does need one. We have proved that this is true for square systems, i.e. when the number of confluences is equal to the number of internal variables. Indeed, it can be proved in this case that, starting from task-oriented confluences, all the pieces of task-oriented assemblages (involving variables different from x) that can be drawn from the initial model can be drawn after the joining rule has been fired as well. We do not give the proof here, because it is too long and requires mathematical notions which are beyond the scope of this paper. It can be found in [Dormoy, 1987].

We have proved further:

> Let E be a non decomposable set of confluences, and x a variable involved exactly in two confluences, say E_1 and E_2. If the resolution rule does not apply to E_1 and E_2 by eliminating x, then no piece of assemblage involving a variable different from x can be drawn from E.

A set of confluences E is said to be decomposable if it contains a subset E' involving variables that are not mentioned in E - E'. In practical terms, if E happens to be decomposable, then one considers E' first. This is what Iwasaki and Simon [1986] called causal ordering. The problem comes down to the study of non decomposable sets of confluences. In concrete terms, a "loop of components" is not decomposable. Efficient algorithms have been described for decomposing a set of equations (see for example [Travé & Kaskurewicz, 1986]).

This second property is important: it states what happens when two components are about to be joined, but ultimately cannot be so. The conclusion seems natural: finding a piece of assemblage for a variable different from x requires eliminating x at some step. This property can be viewed as the "negative part" of the joining rule (it states when joining is not possible).

However, it must be pointed out that this second property never applies when the qualitative model is *stationary*. A *stationary* qualitative model based on confluences can be formally defined as having a *full qualitative rank* (the *qualitative rank* of a system is defined as the maximum number of its column vectors which are *qualitatively independant*). This means that the single solution when all the reference variables are 0 is 0. In physical terms, this means that all the internal variables remain steady when the reference variables do. This is why we call it a *stationary* model. It can be proved that an assemblage can be drawn from a non decomposable model iff it is stationary. The model example presented in this paper is stationary.

2.5 When can the joining rule fail?

The system presented here has been tried in various examples, stemming from different physical areas: electronic circuits, thermodynamic systems (e.g., the pressurizer of a PWR nuclear power plant),.... It never failed in yielding an assemblage in a straightforward way. So, it is justified to ask whether this method is complete, i.e. always leads to an assemblage. If this is the case, then any model which can be assembled must involve at least one variable belonging to exactly two confluences.

Indeed, the joining rule may fail. Some models can be assembled, but have no variable belonging to less than 3 confluences. We shall not discuss the underlying mathematics, but previous work related to this question has to be mentioned.

Similar issues were studied more than twenty five years ago by mathematical economists. They led to many mistakes. Lancaster [1962] claimed that the matrix of any square system having a determinate value turns out to be deducible from the form:

```
+ - 0 . . . 0
+ + - 0 . . 0
.   . . .   .
.     . . . .
+     + + - 0
+ . . . + + -
+ . . . . + +
```

Now, a system having a determinate value can be assembled. This would imply that the joining rule is complete in the square case.

Two years later, Gorman [1964] showed that this is wrong by producing the following counter-examples:

```
 ┌─────┐
 │ N1  │ 0
 └─────┼─────┐
       │ N2  │
   0   │     │
       └─────┘
  + + + + +
```

`N1` and `N2` are square matrices. They have a single line in common. They are themselves supposed to be Lancaster's or Gorman's matrices. Gorman claimed in a footnote that he had proved that all the determinate matrices are deducible from this generic form. Unfortunately, this is wrong, too, as shown by the counter-example:

```
0 + + +
+ 0 - +
+ + 0 -
+ - + 0
```

It can be shown that Lancaster's and Gorman's forms, plus this last form, are the only generic forms of 4x4 matrices. There are 6 basic forms of 5x5 matrices, but we do not know how many there are for nxn matrices with n>5. A generalized control for qualitative resolution is strongly related to these topics.

Let's go back to the real world. The fact that the joining rule works without trouble within a physical model can be justified by a commonsense argument: there must be a variable linking two components, but not involved in the interaction with any other component.

3 Implementation

Here follows a demonstration of how the joining rule is implemented. Though it is self-sufficient, some rules can be added in order to speed up the assembling step. They all turn a set of confluences into an equivalent one. Their advantage lies in the fact that they reduce the number of confluences or of variables. New confluences are produced by some of them: they can also be produced by the resolution rule. However, their complexity is polynomial. Hence, it is worth firing them first.
The set of confluences considered at the current step will be denoted E in the overall section.

3.1 Basic machinery

Let E_0 be the qualitative model to be assembled. Perform *choice, step 0*.

Choice, step i: Select within the current set of confluences $\mathbf{E_i}$ a variable x such that:

© x is involved in exactly two confluences of $\mathbf{E_i}$,

© x has not been yet selected at step i,

© there is a variable different from x involved in $\mathbf{E_i}$ which has not been yet assembled.

Joining rule (JR), step i: Let x be the selected variable, and E_1 and E_2 the confluences involving x. Then, eliminate x by mean of the resolution rule. This produces the confluence E_{12}. Set E_{i+1}<--E_i-{E_1,E_2}U{E_{12}}. Perform choice, step i+1.

Backtracking, step i: Make a new choice, step i. If no such choice is possible, and if i is different from 0, then go back to step i-1.

In addition, as soon as a confluence involving a single variable is produced, the corresponding piece of assemblage is kept and the backtracking step is performed. The "negative part" of the joining rule may also be added.

3.2 Simplification rules

3.2.1 Equality rule

Let ax+by˜0 (e) be a confluence in E, such that a and b are both different from 0. Then x=-aby, and the expression -aby can be substituted for x in all the confluences of E different from (e). Then discarding (e) provides an equivalent set of confluences.

This rule is of great practical importance: it discards a variable and at least one equation. At the same time, there are often in physical systems confluences having the pattern of (e). Some examples are: the valve of the pressure regulator, the form of Ohm's law involving voltage drop, or a confluence of a component involving three variables and corresponding to a "connected-to-ground" component.

Example: (from CE-feedback, see Fig. 4 below)

From $[dV_{FP}]-[di_{B2}]$˜0 one draws $[dV_{FP}]=[di_{B2}]$. $[di_{B2}]$ can be replaced by $[dV_{FP}]$.

3.2.2 Ritschard's rule

In the field of economics, Ritschard proposed [1983] a more constrained form of the resolution rule, but leading to a more informative conclusion (the divergences from the resolution rule are underlined):

Let $x+E_1$˜a (C_1) and $-x+E_2$˜b (C_2) be two confluences, where x is a variable and E_1 and E_2 have no variable with opposite coefficients in common. <u>Assume that all the variables involved in E_2 are also involved in E_1</u> (though the reverse may not be the case). Then E_3˜a+b (C_3) is a valid confluence, where E_3 is the same expression as E_1+E_2, but with no repeated variable. <u>Moreover, if a+b=b, then substituting confluence (C_3) for confluence (C_1) provides an equivalent set of confluences</u>.

This rule eliminates the occurrence of a variable in an equation. Its complexity is polynomial, but it costs much more than the other rules presented here (including the joining rule). Nevertheless, it is worth examining it at the beginning of the assembling step, for it may cause the application of the equality rule (see above) or the single-occurrence-elimination rule (see below).

Example: (from the pressure regulator, see [Dormoy & Raiman, 1988])

This rule applies to the pressure regulator after the equality rule substituted $-[dP_4]$ for $[dA]$.

Let (C_1) and (C_2) be the two confluences:

$$[dP_2]-[dP_3]-[dP_4]-[dQ]\sim 0 \qquad (6)$$

$$[dP_3]-[dP_4]-[dQ]\sim 0 \qquad (3)$$

Then $[dP_3]$ can be eliminated in confluence (6), and confluence (6) can be replaced by confluence (7):

$$[dP_2]-[dP_4]-[dQ]\sim 0 \qquad (7)$$

3.2.3 Single-occurrence-elimination rule

If a variable x occurs in a single confluence (e) involving at least two variables, then discard x and (e) until assembling is completed.

Indeed, (e) is not a constraint upon the variables involved in (e) and different from x: whatever value they are assigned, an assignment to x that satisfies confluence (e) can always be found.

Example: After previous application of Ritschard's rule, $[dP_3]$ occurs only in confluence (3). Hence $[dP_3]$ and (3) can be discarded.

3.2.4 Assemblage propagation rules

These rules generalize the basic propagation rules in order to deal with task-oriented confluences.

Let (e) be a confluence involving a single internal variable x. Then deduce the corresponding piece of assemblage.

This rule simply achieves the goal of the assembling step.

Let $x\sim f(w_1,...,w_p)$ be a piece of assemblage, and (e) a confluence involving x. Then replace x by $f(w_1,...,w_p)$, provided that this adds no new ? coefficient to any reference variable or that (e) has been discarded by the single-occurrence-elimination rule. Any global relation deduced after this replacement will be a piece of assemblage under the usual conditions.

Adding a new ? coefficient to some reference variable could make assemblage deduction impossible. For instance, $[dP_2]$ should not be replaced by $[dP_1]+[dP_5]$ as soon as the resolution rule produces the piece of assemblage:

$$[dP_2]\sim[dP_1]+[dP_5] \qquad (A1)$$

This would lead to a new form of confluence (1):

$$-[dQ]\sim ?[dP_1]+[dP_5]$$

Afterwards, no piece of assemblage could be deduced for $[dQ]$.

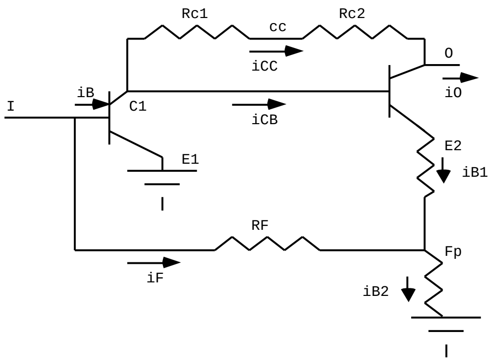
Example: (from the pressure regulator)

This rule draws a piece of assemblage for $[dP_3]$ from confluence (3) and pieces of assemblage (A2) and (A3):

$$[dP_3]\sim[dP_1]+?[dP_5] \qquad (A5)$$

The last part of the rule makes sure that this is a proper piece of assemblage.

3.3 A full example



Transistor Q_1:

$[dv_I]-[di_B]\sim 0$ $\quad$ $[dv_I]-[di_{C1}]\sim 0$

$[dv_I]-[di_{E1}]\sim 0$

Transistor Q_2:

$[dv_{C1E2}]-[di_{CB}]\sim 0$ $\quad$ $[dv_{C1E2}]-[di_{B1}]\sim 0$

Ohm's law:

$[dv_I]-[dv_{FP}]-[di_F]\sim 0$ Ohm(I,FP)

$[dv_{E2}]-[dv_{FP}]-[dv_{C1E2}]\sim 0$ Ohm(E2,FP)

$[dv_{FP}]-[di_{B2}]\sim 0$ Ohm(FP,Ground)

$[dv_{C1}]-[di_{CC}]\sim 0$ Ohm(C1,CC)

KCL:

$[di_I]-[di_B]-[di_F]\sim 0$ KCL(I)

$[di_{C1}]-[di_{CC1}]-[di_{CB}]\sim 0$ KCL(C1)

$[di_{B2}]-[di_F]-[di_{B1}]\sim 0$ KCL(FP)

Definition of drop of potential:

$[dv_{C1E2}]-[dv_{C1}]+[dv_{E2}]\sim 0$ PD(C1,E2)

Figure 4: CE-Feedback and its loop model

We present here how the "loop" in CE-feedback (Fig. 4) [De Kleer, 1984] can be assembled using the joining rule and the simplification rules afore-mentioned. Some inference steps are illustrated by diagrams. They are intended to show the similarity between the way the system joins the components and the way an engineer would.

The equality rule applies first. It gives:

$[dv_I]=[di_B]=-[di_{C1}]=[di_{E1}]$

$[dv_{C1E2}]=[di_{CB}]=-[di_{C2}]=[di_{B1}]$

$[dv_{FP}]=[di_{B2}]$

After replacements have been performed, we get:

$[dv_I]-[dv_{FP}]-[di_F]\sim 0$ Ohm(I,FP)

$[dv_{E2}]-[dv_{FP}]-[dv_{C1E2}]\sim 0$ Ohm(E2,FP)

$-[dv_I]-[di_F]\sim -[di_I]$ KCL(I)

$-[dv_I]-[dv_{C1}]-[dv_{C1E2}]\sim 0$ KCL(C1)

$[dv_{FP}]-[di_F]-[dv_{C1E2}]\sim 0$ KCL(FP)

$[dv_{C1E2}]-[dv_{C1}]+[dv_{E2}]\sim 0$ PD(C1,E2)

The joining rule now applies. The steps are:

Choice, step 0

$[dv_{C1}]$ selected, KCL(C1)-PD(C1,E2)

JR, step 0

$-[dv_I]-[dv_{E2}]-[dv_{C1E2}]\tilde{}0$ (14)

Choice, step 1

$[dv_{E2}]$ selected, Ohm(E2,FP)+(14)

JR, step 1

$-[dv_I]-[dv_{FP}]-[dv_{C1E2}]\tilde{}0$ (15)

Let's sum up the situation. The current model at step 2 is:

$[dv_I]-[dv_{FP}]-[di_F]\tilde{}0$	Ohm(I,FP)
$-[dv_I]-[di_F]\tilde{}-[di_I]$	KCL(I)
$[dv_{FP}]-[di_F]-[dv_{C1E2}]\tilde{}0$	KCL(FP)
$-[dv_I]-[dv_{FP}]-[dv_{C1E2}]\tilde{}0$	(15)

The joining rule goes on firing:

Choice, step 2

$[dv_{C1E2}]$ selected, KCL(FP)-(15)

JR, step 2

$[dv_I]+[dv_{FP}]-[di_F]\tilde{}0$ (16)

Choice, step 3

$[dv_{FP}]$ selected, Ohm(I,FP)-(15)

JR, step 3

$[dv_I]-[di_F]\tilde{}0$ (16)

At this step, the equality rule applies, and deduces that $[dv_I]$ and $[di_F]$ are equal: $[dv_I]=[di_F]$. Propagating this equality in KCL(I) leads to the first pieces of assemblage: $[dv_I]=[di_F]=[di_I]$.

Backtracking to step 2, the second assemblage propagation rule applies. The set of confluences at step 2 reduces to:

$[dv_{FP}]-[dv_{C1E2}]\tilde{}[di_I]$	KCL(FP)
$-[dv_{FP}]-[dv_{C1E2}]\tilde{}[di_I]$	(15)

The joining rule applies again:

Choice, step 2

$[dv_{FP}]$ selected, KCL(FP)+(15)

JR, step 2

$[dv_{C1E2}]\tilde{}[di_F]$

and gets a new piece of assemblage: $[dv_{C1E2}]=[di_F]$.

One can check that no other informative piece of assemblage can be obtained.

4 Conclusion

If not controlled, qualitative resolution leads to combinatorial explosion. But the fact that qualitative models stem from real-world devices prevents qualitative resolution from meeting the fate of resolution in logic. The heuristic control presented here is strongly related to the structural properties of a sane device. We have tried our system in examples corresponding to different physical areas. However, these were all small devices. Nevertheless, we believe that the assembling technique, controlled by the joining heuristic, could assemble some larger artefacts. We are currently working on a model of a large-scale plant.

References

[Bylander, 1987] Tom Bylander. Using consolidation for reasoning about devices. Technical Report of the Ohio State University, March 1987.

[De Kleer, 1984] Johan de Kleer. How circuits work. Artificial Intelligence Vol.24 n°1-3, December 1984.
[Dormoy, 1987] Jean-Luc Dormoy. Résolution qualitative: complétude, interprétation physique et contrôle. Mise en oeuvre dans un langage à base de règles: BOOJUM. Paris 6 University Doctoral Thesis, December 1987.
[Dormoy & Raiman, 1988] Jean-Luc Dormoy and Olivier Raiman. Assembling a device. AAAI88.
[Gorman, 1964] W. M. Gorman. More scope for qualitative econonomics. Review of economic studies XXXI pp 65-68, 1964.
[Iwasaki & Simon, 1986] Y. Iwasaki & H. A. Simon. Causality in device behavior. Artificial Intelligence Vol. 29 n°1, July 1986.
[Lancaster, 1962] K.J. Lancaster. The scope of Qualitative economics. Review of economic studies XXIX n°2, 1962.
[Ritschard, 1983] Gilbert Ritschard. Computable qualitative comparative statics techniques. Econometrica Vol. 51 n°4, July 1983.
[Travé & Kaskurewicz, 1986] Louise Travé & Eugenius Kaskurewicz. Qualitative solutions of linear homogeneous systems. Internal report of the LAAS-CNRS at Toulouse (France), October 1986.