A short sequel to EWD842

I think I have now found a formulation of the required generalization of Leibniz's Rule. As it stood it described how a new formula may be generated "in the presence of $P \equiv R$".

The generalization describes how a new formula may be generated "in the presence of $A \vee B$", essentially by interpreting the presence of $A \vee B$ as the presence of at least 1 from the bag $\{A, B\}$.

If F may be generated (presumably in the presence of A but) without using B <u>and</u> F may be generated (presumably in the presence of B but) without using A, then F may be generated in the presence of $A \vee B$.

In particular —but this is only a special case—: if G may be generated using A but not B, whereas we don't succeed generating G using B but not A, we can form a formula F that satisfies the second requirement as well, by choosing for F the formula $B \vee G$.

<u>Note</u>. In EWD842 —and I remember that I hesitated when I did so!— I defined the disjunction $\vee$ (and consequently the conjunction $\wedge$) only on bags containing <u>at least 2</u> operands. The proper extension is of course to define the disjunction (and ~~the~~ consequently the conjunction) of a bag of 1 operand as that operand, (and to define the disjunction of the empty bag in view of Theorem 5 as $\neg$black, and its conjunction as black). The equivalence on a bag of 1 operand should be defined as that operand, the equivalence on an empty bag should, in view of Theorem 0 be defined as black.

Leibniz's Rule in the presence of an equivalence is now recursively defined according to the syntax

```
<equivalence> ::= <disjunction> {≡<disjunction>}*
<disjunction> ::= <term> {∨ <term>}*
<term>        ::= {¬}* <primary>
<primary>     ::= <predicate variable>
                | (<equivalence>)
```

Remark At least the associativity of ≡ could have been expressed syntactically by

```
<equivalence> ::= <disjunction>
                | <equivalence> ≡ <equivalence>   ,
```

etc. (End of Remark.)
(End of Note.)

Note that we did not interpret the presence of $A \vee B$ as a promise that eventually at least 1 formula from the bag $\{A, B\}$ can be generated! In a common application, $A \vee B$ takes the form $A \vee \neg A$ — which we can generate in view of Theorem 2 for any A — , and the rejected interpretation would exclude the existence of undecidable formulae — i.e. formulae A such that neither A nor $\neg A$ can be generated — .

The above use of the presence of $A \vee \neg A$ in the case of undecidable A does not bother me at all. In that case we can add either A or $\neg A$ to our set of Axioms, thus creating two different systems. Our usage of $A \vee \neg A$ generates what those two systems have in common. So I prefer not to be bothered; undoubtedly I am very naive, but I definitely prefer to remain so as long as possible.

Plataanstraat 5
5671 AL NUENEN
The Netherlands

25 October 1982
prof. dr. Edsger W. Dijkstra
Burroughs Research Fellow