On structures

Most of the proofs in this book are much more calculational than we were used to. As we shall explain later, most of the theorems are (or could be) formulated as boolean expressions for which, in principle, true and false are the possible values; the proofs, however, consist of calculations evaluating these boolean expressions to true. We shall return to this later, for the time being focussing our attention on some of the notational consequences of this approach.

The potential advantages of such a calculational style, we found, are a rather homogeneous proof format and the possibility to obtain brevity without committing the "sin of omission", i.e. making such big leaps in the argument that the reader is left puzzled, wondering how to justify them. In fact, almost all our steps are small and from a repertoire so small that the reader can familiarize himself with it as we go along. These advantages, however, we could only harvest by a careful choice of our notational conventions, thus tailoring our formulae to our manipulative needs, and to our needs of distinction.

Every classical physicist is, for instance, thoroughly familiar (in his own way) with the notion of a vector in three-dimensional Euclidean space, independently of the question of whether it is a displacement, a speed, a force, an acceleration or a component of the electro-magnetic field. Also, he is equally familiar with the sum $v+w$ of two vectors v and w. But that sum raises a question, in particular if one adopts the view —as some physicists do— that their variables stand for the physical quantities themselves and not for their measures in some units. The question is, how many different vector additions the physicist has: is the sum of two speeds the same sort of sum as the sum of two forces? Well, the answer seems negative in the sense that no physicist that is well in his mind will ever add a speed to a force.

Given the fact that in some way we can distinguish those different sorts of additions, one could feel tempted or intellectually obliged to introduce as many different addition symbols as one can distinguish additions, say $+_v$ for the addition of speeds, $+_a$ for the addition of accelerations, etc. In a purist way this would be very correct, but we all know that the physical community has decided against it: it has decided that a single symbol for addition will do.

When challenged to defend that decision, the physicists will give the following argument. Firstly, the purist convention would complicate manipulalations: the differentiation rule for sums

$$\frac{d}{dt}(x+y) = \frac{dx}{dt} + \frac{dy}{dt}$$

would emerge in many forms such as

$$\frac{d}{dt}(x +_v y) = \frac{dx}{dt} +_a \frac{dy}{dt} \quad ,$$

thereby practically destroying that differentiation distributes over addition. Secondly, they would point out that in every physical context the subscripts of the +'s are really redundant, as you can reconstruct the sort of vectors added. And, finally, they would point out that the use of a single addition symbol never seduces them to add a speed to a force, as a simple dimension check protects them against such errors. The defence is purely pragmatic.

The physicist goes further. He knows full well that, depending on the choice of coordinate system, the same vector corresponds to different triples of coordinates. But he also knows that in a context with only one coordinate system, there is a one-to-one correspondence between vectors and coordinate triples, so that he need not distinguish between the two. And this identification gives a new reading to v+w : v and w being triples of coordinates, v+w is a coordinate triple obtained by forming the 3 sums of the corresponding coordinates of v and w "by forming the element-wise sum".

In the mean time we are very close to the notational practice of the matrix calculus, in which the same addition symbol "+" is used to add elements, row vectors of the same length, column vectors of the same length, and whole matrices of the same shape. Is this overloading of the addition symbol "an abuse of notation"? Again, the pragmatist will argue that it is only an abuse in the eyes of the purists (who habitually see abuses everywhere), and he will confidently announce that the use of a single addition symbol will never seduce him to add a column vector to a row vector. (Is he sure he will never do that if both vectors have length 1?)

We would like to point out that in a major part of matrix calculus the apparent overloading of the addition symbol can be eliminated by the simple device of identifying the element with the 1 x 1 matrix and the vectors with 1 x n and n x 1 matrices, thus reducing all "different" additions to the addition of matrices of the same shape. In, for instance, the addition rules

$$\left\|\begin{matrix} A_0 \\ A_1 \end{matrix}\right\| + \left\|\begin{matrix} B_0 \\ B_1 \end{matrix}\right\| = \left\|\begin{matrix} A_0 + B_0 \\ A_1 + B_1 \end{matrix}\right\| \qquad \text{and}$$

$$\|A_0 \quad A_1\| + \|B_0 \quad B_1\| = \|A_0 + B_0 \quad A_1 + B_1\|$$

the A's and B's are not restricted to elements, they may be matrices of any properly matching shapes.

<u>Remark</u> In passing we note that the same applies to the A's and the B's in the rule for matrix multiplication

$$\left\| \begin{matrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{matrix} \right\| \times \left\| \begin{matrix} B_{00} & B_{01} \\ B_{10} & B_{11} \end{matrix} \right\| =$$

$$\left\| \begin{matrix} A_{00}\times B_{00} + A_{01}\times B_{10} & A_{00}\times B_{01} + A_{01}\times B_{11} \\ A_{10}\times B_{00} + A_{11}\times B_{10} & A_{10}\times B_{01} + A_{11}\times B_{11} \end{matrix} \right\|$$

and in the rule for transposition

$$\left\| \begin{matrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{matrix} \right\|^T = \left\| \begin{matrix} A_{00}^T & A_{10}^T \\ A_{01}^T & A_{11}^T \end{matrix} \right\| \quad .$$

(End of Remark.)

Ignoring the distinction between an element and a 1×1 matrix – e.g. by not mentioning elements at all – allows one to cover only part of the matrix calculus: for instance, for the definition of the rank of a matrix, the concept of "element" is indispensable. On the other hand, when such concepts are not needed, carrying around the distinction between an element and a 1×1 matrix would be a needless complication. Such complications will be avoided in the theory that follows later.

From now onwards we assume that, for matrices A and B of the same shape, the reader knows what is meant by expressions such as $-A$, $A+B$ and $A-B$: a matrix of the same shape, obtained by the element-wise application of the unary or binary operator.

The matrices A and B we talked about so far are a specific example of something more general. (In this generalization, there is no room any more for an analogue of the very matrix-specific operation of the matrix multiplication that we mentioned in a remark; we confined that to a remark so as to assist the reader in forgetting that we ever mentioned it.)

We can view a real matrix as a real function defined on a finite —in most visualizations rectangular! — grid of points, with as many points as the matrix has elements, and as many "dimensions" as the matrix has. We consider that domain on which the real function is defined, i.e. that grid of points, to be a "space" and call the matrix "a real structure on that space". Two different structures are said to be "of the same shape" if they are structures on the same space.

The generalization concerned is that that space need no longer be a finite rectangular grid. In large parts of our considerations we do not even care how many coordinates of what types might be needed to identify all the distinct points of that space. The only thing to be understood at this stage is that all the points of the space have, though not explicitly labelled, a distinct identity so as to define what is meant by "element-wise" application of operators. With A and B real structures of the same shape, A+B is a real structure of the same shape and there is no confusion which pairs of real numbers have to be added together and where in the space those sums belong: under all circumstances A+B-B yields A again. For reasons that will become clear later, we impose one restriction on the spaces on which we care to introduce structures: the spaces are non-empty, i.e. contain at least one point.

Note As potential labelling of the points, so as to be able to distinguish distinct points of the space, has no rôle in a one-point space, the one and only point of a one-point space is considered truly anonymous: the one-point space can do completely without coordinates. This view regards a constant (such as 3 or 4) as a function of zero variables. (End of Note.)

Just before introducing the matrix as an example of a structure, we said one could view a real matrix as a real "function defined on a finite grid". Well, the latter was an abuse of language — that we allowed ourselves for a short while because we could not explain everything simultaneously — because structures are not functions: structures are more like expressions in the coordinates of the space in question.

Let us spend some time on explaining the difference. Let f be a natural function of a natural argument. Then, for instance, $f.87$ stands for a natural number — we use the period in formulae as visible notation for functional application —. Let x and y be natural variables. What are $f.x$ and $f.y$? They are not functions, they are expressions in the natural variables x and y respectively: though expressed in terms of the same function f they are different expressions. Let us introduce for them the shorthands X and Y respectively. Then, X and Y are natural structures, but of different shape, because they are structures on different spaces, viz. spanned by the natural coordinates x and y respectively.

Are we going to allow the new structures $X+X$ and $X \cdot X$ — using the raised dot to denote multiplication — ? Well, we have already decided that we shall do so: our rule of element-wise application is in accordance with considering them shorthands for $f.x + f.x$ and $f.x \cdot f.x$ — where functional application is given a higher syntactic binding power than the arithmetic operators —. Are we now also going to allow a structure like $2 \cdot X$ and would it then be the same one as $X+X$? The answer is two times "yes": we accept $2 \cdot X$ as shorthand for $2 \cdot f.x$, and since, for all x, $2 \cdot f.x = f.x + f.x$, we equate the two structures $2 \cdot X$ and $X+X$.

Why so much fuss about such a trivial decision? Well, look at the now accepted structure $2 \cdot X$

from the point of view of element-wise application: with X an infinite sequence of natural numbers, it dangerously looks as if we have decided to ignore the difference between a single 2 and an infinite series of 2's !

This looks a bit alarming, but we should derive some comfort from the following observation. The rule for differentiation of a product

$$\frac{d}{dx}(u \cdot v) = \left(\frac{du}{dx}\right) \cdot v + u \cdot \left(\frac{dv}{dx}\right)$$

has now for centuries been applied quite successfully to differentiate both of the products $x \cdot x$ and $2 \cdot x$: in the latter case the constant 2 is in fact treated as an expression in x , whose derivative is everywhere — i.e. for all x — equal to zero. It is apparently not completely unsafe to consider a constant as an expression in a variable — be it then a "silent" variable, that does not occur in / has no influence on the value of the variable.

And now we know what to do! An expression in x is also an expression in x and y , be it that y is then one of its silent variables; an expression in y is similarly an expression in x and y (this time with silent x). So now we also know what meaning to attach to $X + Y$ and $X \cdot Y$ — were f the identity function, they would stand for the two-dimensional addition and multiplication table, whereas $X \cdot X$ would stand for the one-dimensional table of squares — .

In short, for integer or real (or complex) structures X and Y, the expression X + Y is a structure on the space spanned by the union of the coordinates of the spaces on which X and Y are structures. The resulting space is sometimes referred to as "the smallest Cartesian multiple" in analogy to the Cartesian product (which would be the resulting space, if the operand spaces are spanned by disjoint coordinate sets).

The anonymity of the spaces supporting our structures could conceivably lead to confusion, had we arguments in which structures on many different spaces occurred. Fortunately, our arguments deal with structures on very few different spaces. We usually encounter

(i) the one-point space, as each constant may be —and usually is— treated as a structure on it;

(ii) one arbitrary space, which is a hardly noticeable parameter of the whole argument; in most applications of our theory, it is the state space of the program under consideration.

Let us refer to the space mentioned under (ii) by "the state space". In certain arguments we shall encounter structures defined on

(iii) the Cartesion product of the state space and (the space of) some other coordinate.

The simplest way in which we encounter structures on a space of type (iii) is by pair-forming -or, more general, n-tuple forming- . With $X0$ and $X1$ integer structures on the state space, the pair $(X0, X1)$ is, in turn, a single integer structure, be it "on a doubled state space", i.e. the Cartesian product of the state space and a two-valued coordinate -ranging probably over $\{0,1\}$- . Sometimes it is preferable to consider $(X0, X1)$ as a pair of structures, other arguments go more smoothly if we consider $(X0, X1)$ as a single structure on a space "twice as big".

Similarly we may consider the infinite sequence $X.i$ $(0 \leq i)$ of -say, real- structures on the state space to be a single -real- structure on the Cartesian product of the state space and that of a natural variable. Each time we shall choose the most convenient way.

We can think of other ways of building up -or partitioning- spaces, but this is not the place to elaborate them; we shall do so if and where the need arises.

Remark The reader may wonder why we have spent so many pages on such a cautious introduction of structures defined on a space. Once one has grasped the notion, it seems pretty simple and straightforward. In fact, it is.

Our caution has been inspired by our experience. With only a quick explanation, people get confused by it, not because it is difficult, but because it is different from what they are used to. People have a feeling of familiarity with what they have learned to regard as "mathematical objects" — points, lines, planes, triangles, sets, subsets, elements, functions, matrices, vectors, quaternions, integers, reals — . Accordingly, they are quite used to naming them — the point P, the line l, the function f — and feel quite comfortable with formulae with such names as they can interpret these formulae in terms of the model provided by these familiar "mathematical objects".

By an accident of history, many are unused to the introduction of names for something like otherwise unspecified expressions in otherwise unspecified variables. And even after they have accepted that model, it may still take quite some time before they are comfortable with it: an expression in Heaven knows how many variables, isn't that potentially a very complicated thing?

Yes, it is, but not on the level of abstraction in which we deal with our structures. The rules according to which we manipulate formulae with variables and constants standing for structures are so independent of the space on which they are defined that, as time goes by, the model fades in very much the same way as we have learned to add two big decimal numbers without worrying whether the sum digit we are determining stands for the tens, the thousands, or the millions. Already thousands, let alone millions, baffle the imagination, so it is much simpler not to try to visualize them when adding two big numbers and to manipulate instead two short uninterpreted sequences of decimal digits. But remember how long it took us in our youth to learn how to do the latter!

With our structures the reader will observe the same phenomenon: as he becomes more and more familiar with the rules of manipulation, the model will retreat into the recesses of his mind (which is a good thing because —as we shall see later— the model is overspecific in almost all of our theory). But all this takes time and a lot of getting used to. Hence the (large) number of pages spent on our cautious introduction of structures defined on some space. (End of Remark.)

* *
*

We have one further notational hurdle to take, and that one has to do with the notion of equality. Difficulty with the notion of equality might surprise the unsuspecting reader who -not without justification- feels that equality is one of the most fundamental, one of the most "natural" relations. But that is precisely the trouble! The notion of equality came so natural that for many centuries it was overlooked in the sense that there was no symbol to express it. Equality was expressed verbally or implicitly —by writing two expressions on the same line and leaving the conclusion to the intelligent reader— and we had to wait until Robert Recorde gave us (in "Whetstone of witte") in 1557 the -consciously designed!- symbol = to denote equality. In the words of E.T. Bell, "It remained for Recorde to do the right thing.".

Yes, Recorde did the right thing, but the speed with which mankind can absorb progress is very limited.

Example In the late 18th century a Dutchman by the name of Bartjes wrote an at the time well-received textbook on elementary arithmetic. (It must have been widely accepted, for the name of Bartjes still survives in a Dutch expression - "volgens Bartjes" - indicating the obviousness of a conclusion.) In his book, Bartjes used a decimal notation of sorts: in writing 102 he omitted the zero but provided the 1 with a superscript C to indicate its weight! And this almost six centuries after the "Liber abaci" (1202) of Leonardo of Pisa, also known as Fibonacci, in the words of E.T. Bell "at last converted Europe to the Hindu arithmetic". Evidently, the conversion was far from complete. Is it a wonder that many teachers consider mankind as almost education-proof? (End of Example.)

In the case of the equality sign it took almost another three centuries before the equality sign gained the full status of an infix operator that assigned a value to expressions of the form $a = b$. The landmark was the publication of George Boole's "Laws of thought" in 1854 (fully titled "An Investigation of The Laws of Thought on which are founded The Mathematical Theories of Logic and Probabilities"). George Boole did so by the introduction of what is now known as the

"Boolean domain", a two-valued domain comprising the values denoted by "true" and "false" respectively. It goes without saying that the boolean values, being only 130 years old, are by and large still treated as second-class citizens. (Without saying, indeed: what else could we expect from a mathematical community that in the 20th century has not made up its mind yet whether to count zero among the natural numbers!)

Let X and Y be two real structures of the same shape, say matrices. Had people been thoroughly familiar with the boolean domain when matrices were introduced, they might have attached to $X=Y$ a meaning in analogy to the meaning they had decided to attach to $X+Y$. Since $X+Y$ stands for a real matrix of the same shape, formed by element-wise addition, it stands to reason to let $X=Y$ stand for a boolean matrix of the same shape, formed by element-wise comparison.

But this is not what happened. People were at the time unfamiliar with the boolean domain, a boolean matrix was beyond their vision, and hence $X=Y$ was interpreted as the statement that X and Y were "completely" - i.e. element by element - equal. In retrospect we can consider this conventional interpretation an anomaly, but that is not the point. The point is: can we live with that anomaly in the same way as mathematicians have lived with it since matrices were invented?

It would be convenient, were the answer affirmative, as it could be in a context in which there is no need for dealing with boolean structures. In our case, however, structures being boolean will be rather the rule than the exception and we have to do something about it.

One could try to find a way out by introducing two different equality signs, one to express the complete equality of the two structures and one to express the boolean structure formed by element-wise comparison. This, however, would lead to an explosion of symbols as the same problem presents itself in expressions like $X < Y$, $X \leq Y$ etc.

After ample consideration we have decided that a relational operator connecting two structures of the same shape is to be applied element-wise and thus yields a boolean structure of the same shape. In addition we provide a notation expressing that all elements of a boolean structure are true: for any boolean structure Z, $[Z]$ stands for the boolean value expressing that all elements of Z are true. Hence, for X and Y of the same shape

$[X = Y]$ expresses that structures X and Y are completely equal;

$\neg [X = Y]$ expresses that structures X and Y are not completely equal, i.e. that there exists a point in the space on which they are defined at which they differ;

$[\neg(X = Y)]$ expresses that structures X and Y are completely different, i.e. differ in each point of the space on which they are defined.

The ample considerations mentioned involved more than a fear of an explosion of the number of symbols: if one needs a large class of symbols in two versions, one can introduce a convention for generating the symbols in the other version. The point is that there is no justification for attaching to an operator the distinction that is now expressed by the presence or absence of a surrounding pair of square brackets. To begin with, the operator to attach the distinction to might not be there, as exemplified by the expression $[Z]$. Secondly, it may be unclear which operator to attach it to. Consider

$$[(X \vee Y \vee Z) = (X \vee Y)] \quad ;$$

in this case the candidate $=$ for marking is quite clear. But later we shall see that the above may be replaced by the shorter

$$[X \vee Y \vee (\neg Z)] \quad ;$$

which of the $\vee$ should now be marked? The first one? The last one? Both?

Instead of marking each time an operator —whose presence or identity we have just seen to be subject to doubt— it is more convenient and in a way more honest to introduce one extra unary operator, whose operand can then be manipulated freely. We have allowed ourselves to denote this unary operator by a distinctive bracket pair surrounding its argument because almost always that argument is a sizeable expression.

In short, we use square brackets to denote a unary operator with all the properties of universal quantification over the space on which the enclosed boolean structure is defined. The introduction of this special notation is the price we pay for the anonimity of the coordinates of that space; as we proceed, it will transpire that this price is well-paid

It is in this connection that we recall our liberty to treat values usually not considered structures as structures on the one-point space. This convention attaches meaning to $[x=y]$ also in the case that x and y stand for such simple values such as integers, reals, or booleans.

Finally, in the case of boolean operands, we admit besides $=$ an alternative symbol for the equality relation, viz. $\equiv$. We do so for more than historical reasons. The one reason is purely opportunistic as it gives us a way out of the traditional syntactic dilemma what binding power to assign to the infix equality operator. By giving $=$ a higher binding power than the boolean operators and $\equiv$ one of the lowest binding powers we enable ourselves to write

$$x=y \equiv x \leq y \wedge x \geq y$$

without parenthesis. The other reason that justifies a special symbol is that in the case of boolean operands, equality —in this connection also called equivalence— enjoys a special property: in that case it is associative.

Note A special character is more attractive than the alternative of giving equality between boolean operands a lower binding power than equality between operands of other types, as the latter convention would make parsing context-dependent. (End of Note.)

Comment After completion of the above I asked myself whether I should state explicitly that infix operators defined not to distribute over pair forming don't distribute over pair forming. The example that comes to mind is lexical ordering. (End of Comment.)

Austin, 25 November 1985

prof.dr. Edsger W. Dijkstra
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78712-1188
USA