Two-categorical models of parallelism

C.A.R. Hoare

February 1995

Summary

This paper follows Joseph Goguen's suggestion that two-categories provide a good framework within which to construct models of parallelism. The horizontal and vertical compositions of the two-category are applied to the individual observations of the behaviour of processes, which are composed in series (for sequential execution) or in parallel (for concurrent execution). As in Burghard von Karger's [?] Sequential Calculus, a process is modelled by the set of observations that it could possibly give rise to; and the two compositions are defined pointwise. Their algebraic properties turn out to be shared by a fair range of models of parallelism.

1 Parallel composition of observations.

It is well known that categorical composition provides a good model of sequential composition of (observations of) sequential programs. It is equally applicable to parallel composition of concurrent processes: here are four very simple examples.

1. The behaviour of some process may be recorded as a sequence t of interactions with its environment. The behaviour of the environment is similarly recorded as a sequence s of interactions with the process. A record of the parallel interaction of a process composed with its environment must be simultaneously a record of both their behaviours, i.e., t must equal s. This is the tightest possible coupling between two concurrent processes!

   The lock-step model of parallelism is represented by the *discrete* category, where all the arrows are identities, and they compose only with themselves. This composition is denoted δ, after the Kronecker delta. Its properties are expressed categorically:

   $$x\uparrow^1 = x\downarrow = x$$
   $$x\delta x = x.$$

   Like many parallel combinators, δ is commutative as well as associative. When lifted to sets, this operation turns to simple intersection, which is exactly the way that parallelism is introduced in the trace model of CSP:

   $$\{x\delta y \mid x \in X \wedge y \in Y\} = X \cap Y.$$

[1] For $x : A \rightarrow B$, $x\uparrow$ refers to the unit of A (the source of x), and $x\downarrow$ to the unit of B (the target of x).

2. Another familiar kind of parallelism is the coupling of two processes evolving simultaneously but disjointly, in complete isolation from each other. Their joint behaviour is well described as a pair (x, y), where x describes the behaviour of one of them, and y the behaviour of the other.

   This pairing operator is everywhere defined, so we need a special object, the terminal object 1, which is the source and target of every arrow

   $$x\downarrow = x\uparrow = 1.$$

   Of course, pairing is not associative; but there is a canonical isomorphism between different bracketings of pairs:

   $$((x, y), z) \sim (x, (y, z)).$$

   That is good enough for the purposes of categorical modelling.

3. Most interesting models of concurrent execution lie between the extremes of lockstep concurrency and complete disjointness. There are certain events (like synchronised communications) in which the two processes engage simultaneously together, and certain other events or occurrences in which they engage separately and independently. For example, let us confine attention to just a single event of interest, and let p be a set of times at which one process engages in an occurrence of that event, and let q be the set of times at which another process engages in the same event. Then $(p \cup q)$ is the set of event times observable of the process resulting from their parallel composition.

   The times in the intersection $(p \cap q)$ represent events in which both processes have engaged simultaneously; whereas times in their exclusive union $(p \nabla q)$ represent events in which only one of them has engaged; their joint events have been treated as internal events, and so concealed. This models the CCS definition of parallelism, which is only slightly more complicated.

   In general, a powerset M is a commutative monoid $(M, \cup, \{\ \})$, with composition $\cup$ and unit $\{\ \}$. It is turned into a category by the definitions $p\uparrow = p\downarrow = \{\ \}$.

4. Consider a process that engages in only two kinds of event, say inputs and outputs, where we ignore the value of the message communicated. Its behaviour is represented by the *pair* of sets (p, q), where p gives the times of the input events and q gives the times of the output events. Two such processes can be chained $(\gg)$ by connecting the outputs of the first to the inputs of the second, so that each message is input and output at the same time; but each occurrence of such an internal communication is concealed. This behaviour is defined in the same way as for the relational model of sequential composition:

   $$\begin{aligned}(p, q)\downarrow &= (q, q) = \uparrow(q, p)\\ (p, q) \gg (q, r) &= (p, r).\end{aligned}$$

   This gives the same effect as chaining in CSP. In a more realistic model the sets p and q will contain pairs (t, m), where t is the time at which message value m was communicated. In further generality, a greater number of channels can be modelled by a tuple of such sets.

2 Two-categories: sequential and parallel composition.

If we wish to model both sequential and parallel combinators, we will need two separate compositions: the sequential composition, denoted by semicolon, is often called horizontal (with horizontal source and target arrows); and the parallel composition, denoted by $\|$, is called vertical (with vertical source and target arrows). The category is called a two-category if the two compositions distribute through each other in the sense of the exchange law:

$$(x;y)\|(a;b) \;=\; (x\|a);(y\|b) \qquad \text{whenever } \textit{both} \text{ sides are defined.}$$

The proviso is equivalent to

$$\begin{array}{rl} & \overrightarrow{x} \;=\; \overleftarrow{y} \text{ and } \overrightarrow{a} \;=\; \overleftarrow{b} \\ \text{and} \quad & x\downarrow \;=\; a\uparrow \text{ and } y\downarrow \;=\; b\uparrow . \end{array}$$

The other defining condition is that every horizontal unit must also be a vertical unit, i.e. $\overrightarrow{x}\uparrow \;=\; \overrightarrow{x}$, $\overleftarrow{x}\downarrow \;=\; \overleftarrow{x}$, etc.

1. The simplest example is provided by the Kronecker δ; its introduction as a vertical composition into *any* category makes it into a two-category

   $$\begin{array}{l} \overrightarrow{x}\uparrow \;=\; \overrightarrow{x} \;=\; (x\downarrow)^{\rightarrow} \\ (x;y)\delta(a;b) \;=\; (x\delta a);(y\delta b) \qquad \text{whenever } x=a \text{ and } y=b. \end{array}$$

   The horizontal units have the form $\overrightarrow{x}$ or $\overleftarrow{x}$; all of them are vertical units, because every arrow is a vertical unit.

2. The standard definition of composition in the product category is nothing but a statement of the exchange law:

   $$(x,a);(y,b) \;=\; ((x;y),(a;b)).$$

   Pairing has to be classified as the horizontal composition, because the only horizontal unit is the terminal unit, which is certainly a unit for the other composition

   $$(x\uparrow)^{\rightarrow} \;=\; \overrightarrow{1} \;=\; 1 \;=\; x\uparrow .$$

3. Let T be the partial order category of time intervals. Its arrows are pairs (x,y) such that $x \leq y$, and

   $$(x,y);(y,z) \;=\; (x,z).$$

   Let M be the monoid of time-sets under union. Consider the category $T\times M$. The standard composition of the product category gives:

   $$\begin{array}{l} (s,p)^{\rightarrow} \;=\; (\overrightarrow{s},\{\,\}) \\ (s,p)^{\leftarrow} \;=\; (\overleftarrow{s},\{\,\}) \\ (s,p);(t,q) \;=\; (s;t,p\cup q). \end{array}$$

In (s,p), $\overleftarrow{s}$ and $\overrightarrow{s}$ represent the start time and finish time of the process, and p represents the intermediate times at which it interacts with its environment. So it is reasonable to stipulate that $p \subseteq s$; and then the definition given above is an exact description of an observation of the sequential composition of two processes, contributing the behaviours (s,p) and (t,q) respectively to their joint observer.

When two processes are run in parallel, they begin and end at the same time; and each event that occurs is contributed by one or both the partners. That leads to a definition of the vertical composition:

$$(s,p)\downarrow = (s,\{\}) = (s,p)\uparrow$$
$$(s,p)\|(s,q) = (s,p \cup q).$$

This example is just the standard definition for the two-categorical product $T \times M$, where T and M are two-categories, and T is regarded as the two-category $(T,;,\delta)$ and M as the two-category $(M,\cup,\cup)$. Obviously, any commutative composition like union forms a two-category with itself.

4. Consider the categorical chaining operation ($\gg$) defined in 1. 4 on the product of the monoid M with itself. But M is itself a category with composition $\cup$, which lifts in the standard way to the product $M \times M$. This is now a two-category with $\cup$ as horizontal composition and $\gg$ as vertical

$$(m,n)^{\rightarrow} = (\{\},\{\}) = (m,n)^{\leftarrow}$$
$$(m,n)\uparrow = (m,m) \quad (m,n)\downarrow = (n,n)$$
$$(p,q) \cup (r,s) = ((p \cup r),(q \cup s))$$
$$(p,q) \gg (q,r) = (p,r).$$

The exchange law is proved:

$$\begin{aligned} & [(p,q) \cup (r,s)] \gg [(p',q') \cup (r',s')] \\ = \ & [(p \cup r),(q \cup s)] \gg [(p' \cup r') \cup (q' \cup s'] \\ = \ & (p \cup r),(q' \cup s)] \qquad \text{if } q = p' \wedge s = r' \\ = \ & (p,q) \gg (p',q')] \cup [(r,s) \gg (r',s')]. \end{aligned}$$

$(\{\},\{\})$ is the only horizontal unit, whereas vertical units have the form (p,p) for any set p . It is to satisfy this unit law that M must be a monoid (with only one unit). The exchange law would be satisfied if M were any category.

This example is interesting in that both the operators involved can be interpreted as different forms of parallelism, one implemented by interleaving and the other by synchronisation. A sequential composition can be introduced by the construction of the previous section.

3 Lifting to sets: additional realism and complexity.

The simple examples given so far fail to recognise some of the complicated ways in which real parallel systems can go wrong, for example by deadlock, or divergence. Furthermore, we have not modelled the important external choice operator (+ in CCS or $[]$ in CSP).

Realistic modelling of these errors will require more complicated observations, for example inclusion of a deadlocked observation (say O, with $\overleftarrow{O} = \overrightarrow{O} = O$). This could be the target of all observations that end in deadlock, whereas $\surd$ could be the target of those that terminate properly. Let $\surd$ be the source of *all* observations except O; this ensures that all properly terminated processes can be extended by sequential composition, but deadlocked ones cannot, because $(O; x)$ is undefined for all x other than O.

In general, one also needs to consider the interplay between individual observations and the set of observations that model all the possible behaviours of a process, perhaps a non-deterministic one. Consider, for example, a process that may immediately deadlock or may immediately terminate. It has two observations $\{x, y\}$ where

$$\overleftarrow{x} = \overleftarrow{y} = \surd = \overrightarrow{y} \text{ and } \overrightarrow{x} = O.$$

Consider its sequential composition with $\{y\}$, the deterministically terminating process SKIP

$$\{x, y\}; \{y\} = \{y\}!$$

(in CSP, this "law" would be expressed $(\texttt{STOP} \sqcap \texttt{SKIP}); \texttt{SKIP} = \texttt{SKIP}$).

The theory says that the deadlock (x) disappears: in practice it does not. It is the theory that must be adjusted by ensuring that *every* set representing a process will respect prior deadlock. An easy way of doing this is to stipulate that every set contains the object O itself. Now we have

$$\{O, x, y\}; \{O, y\} = \{O, x, y\}$$

and the possibility of deadlock is preserved.

It would be interesting gradually to extend the complexity of the simplest models, say in the direction of timed CSP, or Abramski's interaction categories. Would anyone like to help in doing this?

Meanwhile, thanks to Gavin Lowe for help and encouragement.