Teams Can See Pomsets

Extended Abstract

Gordon Plotkin† Vaughan Pratt‡

August 1990

† Dept. of Computer Science, University of Edinburgh
‡ Dept. of Computer Science, Stanford University

Abstract

Strings may serve as both specifications and observations of behavior. However partial strings or pomsets, superior to strings in certain respects for the representation of concurrent behavior, are provably unobservable and hence apparently suitable only for specifying behavior. The proof however tacitly assumes that observers are isolated individuals. We show that observations by a cooperating team of sequential observers agreeing on *what* happened but not *when* can distinguish all pomsets. The resolving power of a finite team increases strictly with its size k, permitting it to distinguish all pomsets of dimension (not width) k but not all of $k+1$. These results extend to observation of augment closed processes. As expected we depend on the now standard technique of refinement of atomic events to complex events; what is not expected is that their complexity need be only that of nondeterminism, in that we refine one atomic event to a set of alternative atomic events, not to a set of sequences.

1 Introduction

Whereas a sequential observer can perceive only a sequential universe, we shall show that a concurrent observer can see a concurrent one. However we shall model concurrency at the observing end by taking the observer to be a team of communicating sequential processes whose observations are sequences or linearly ordered multisets, and at the observed end in terms of partially ordered multisets or pomsets. In the latter view of concurrency, a pair of events is either *sequential* or *concurrent* according to whether or not the events are comparable. An agent is concurrent or sequential according to whether or not it permits such concurrency. Our real result then is that the latter view draws only those distinctions already visible in the former view.

Our result may appear to contradict recent results about full abstractness of trace semantics for concurrent computation. A semantics of behavior is said to be fully abstract when it makes only those distinctions inferrable from observations of that behavior. Jonsson [Jon] and Russell [Rus] claim to have shown in an absolute sense that trace semantics is fully abstract while pomset semantics is not.

However their proofs each begin by ascribing to their observer exactly the semantics they seek to characterize as fully abstract, namely trace semantics. The essence of their result is then that a single sequential observer can see only sequential behavior. From this they infer that only sequential behavior need be modeled, since any additional distinctions are indiscernible.

A proof of full abstractness of a model by appeal to observational equivalence cannot be construed as evidence of the sufficiency of that model if it ascribes the limitations of the model to the observer. If, Virginia, concurrency does exist, then we must allow for the possibility of concurrent observers, who could well be first class citizens of the universe they observe.

Thus even without our result we have already identified a basic weakness of the extant proofs of full abstractness of trace semantics. In fact as soon as we introduce concurrent observation, not only is the full abstractness result contradicted, but worse, trace semantics becomes unsound by identifying observationally distinguishable situations!

However, were we to prove that pomset semantics is fully abstract by ascribing pomset semantics to observers we would not have significantly improved our understanding of concurrency semantics, since our result in turn would be rendered not only pointless but unsound with a yet more discerning model of computation whose observers were similarly more discerning.

Instead our result is that pomset semantics draws no distinction not inferrable by a set of observers pooling their respective sequential observations. That is, the pomset view is no more discerning than the communicating sequential processes view.

A cornerstone of our argument is the distinction between communication between observers during pooling of sequential observations and the subsequent official communication of their joint observation.

2 Rationale

We now give some rather concrete examples illustrating the main intuitions behind this paper. Though some of the distinctions being drawn here may seem trivially pedantic, they are nevertheless distinctions that any formal model must address. And as the examples get larger the distinctions look less trivial.

A string is a linearly ordered set of symbols, or rather multiset since symbols may be repeated. Many of the intuitions and techniques of formal language theory generalize smoothly when the linear order is generalized to partial; for example the terms alphabet, one-letter alphabet, homomorphism, and substitution [HU] all carry through unambiguously. We call this generalization of string a *partially ordered multiset* [Pr82], abbreviated pomset.

Now picture yourself at a bank. You withdraw ten dollars, the teller asks how you want it, you say "Two fives, please." You have thereby specified a one-letter multiset. You have not specified an amount, in that you won't settle for ten ones. You have not specified a set, for that would imply particular five-dollar bills. You have however specified a set up to isomorphism, meaning that any two sets of five-dollar bills in bijective correspondence will be equally acceptable. But this is what we mean by a multiset. And although you do not appear to have specified an order, this is what we mean by the discrete or empty order, in which no two elements are comparable. Thus you have specified a pomset that happens to be a multiset.

Now suppose to this specification we add "And one at a time, please." We may distinguish the previous specification from this one as respectively $5|5$ and $5;5$ or just 55, their *concurrence* versus their *concatenation.* The former is a multiset, the latter a string, but both are pomsets.

Is there any observable difference?

Had we wanted only six dollars, as a single followed by a five, we would not run into the multiset aspect of this problem. $1|5$ clearly differs from 15 in allowing the possibility of 51.

We take into account three relevant factors, that the bills actually delivered do form a set, i.e. have identities, that they have length, and that their delivery may have more than one observer.

Length of bill is of course a spatial surrogate for duration of events. With length in the picture, overlap gives an obvious criterion for nonsequentiality: if the bills arrive overlapped we have observed $5|5$ without observing 55. If however they are not overlapped we have observed both 55 and $5|5$ (the latter did not *require* overlap). All this assumed just one observer.

Now ignore length and treat bills as atomic. Consider instead the set aspect, calling the two bills x and y. Evidence for the delivered bills forming a set is that if the teller took back a bill and replaced it by another, then even though we knew that this occasioned no change in whether our specification was met we would nevertheless attach significance to this switch.

To a single observer constrained to make only sequential observations of atomic objects, $5|5$ will look like 55 whether or not he takes the identities of the bills into account. However two sequential observers who agree on the identities of the bills, but disagree on their order due to imperfect observation of timing, can infer from their disagreement a violation of 55 but not of $5|5$.

Much the same holds when the two bills are of different denominations. Two observers seeing respectively 15 and 51 may infer $1|5$ and a violation of both 15 and 51, except that now they need not take identity into account. But in this case a single observer who saw 15 and subsequently 51 as observations of separate transactions would not describe the process responsible for these transaction as 15 or 51, but on the other hand he could interpret this not as $1|5$ but rather as the *choice* $15 + 51$ expressing that one or the other of 15 or 51 happened.

If two observers of one transaction however see respectively 15 and 51 then this rules out $15 + 51$ since they do not agree that one or the other happened. However one observer cannot use two observations to draw this inference since these are observations of different situations, as opposed to two observations of the same situation from different vantage points. Thus two observers can distinguish $1|5$ from $15 + 51$ in a way that one observer cannot.

This interpretation of concurrence is then one of reconciling inconsistent observations of relationships arising from spatially different but temporally simultaneous points of view. This constitutes a crude form of averaging of observations, a universally accepted technique in the natural sciences that can regarded either as a legitimate form of observation or as a technique of compensation for unavoidable imperfections in observation.

With regard to identity, a natural objection is that whatever it was that the observers saw that enabled them to reach consensus as to the identities of the bills, such as the serial numbers on the bill, their condition, or their location, that extra information should be reported as part of the final joint observation. But this assumes that the language of communication between observers as they seek consensus is no more than that of their joint report. This assumption is unwarranted. As for any communication situation, the reporting language must be defined in advance. But there is no reason to suppose that the team of observers will limit themselves to that language in exchanging information during the observation, or even to the same language from one observation to the next.

The legal system has taken this distinction to an extreme. A sequestered jury is permitted to use any language, even profane, over a period of possibly weeks to achieve a consensus. However their final joint observation is a single bit! Newspapers may extract a few more bits by interviewing the jury, but even a transcription of the entire dialogue would omit such factors as body language contributing to the decision.

Since the length factor was so much less subtle than the identity factor, why not assume that everything has length and simply not bother with identity? The reason is that length alone is not enough, as the following example shows.

We now ask mysteriously for "Two sixes, please." "What's a six?" "A one followed by a five." We can write this as $15|15$. We then add that one of the singles must precede *both* fives (or equivalently one of the fives must follow both singles); we express this as the more constrained pomset $N(1,1,5,5)$.

What exactly does this last constraint add? Try it yourself with two pairs of whatever comes to hand, call them bills. For each pair place the dollar entirely to the left of the five. This meets the first specification. Now try to adjust the position of one pair relative to the other so as to violate the second specification, $N(1,1,5,5)$, i.e. neither of the dollars strictly to the left of *both* fives (but each dollar has to remain to the left of its matching five). Either you are in trouble or we are.

But here's a way two observers can see $15|15$ without seeing $N(1,1,5,5)$. Place all the bills flat on a table, all with their major axis left-right. Align the two dollars with each other, top edge of one against bottom edge of the other, and likewise the two fives, and place the two dollars with their common right edge half an inch to the left of the common left edge of the fives (each bill being 2.5" high). Seat the two observers with the centers of their heads at table height, four feet back and one foot apart. From this awkward but cozy position the left observer sees a single overlap, namely the near dollar with the far five, while the right observer also sees a single overlap, of the near five with the far dollar. Each observer individually is satisfied that $N(1,1,5,5)$ is met, but when they compare notes each sees that the other has evidence against the sequentiality of the one diagonally opposite pair they thought was sequential. They therefore rule out $N(1,1,5,5)$. Howeever they agree that each dollar precedes its matching five and so report $15|15$.

The disagreement as to which pair of bills is overlapping of course arises from an imperfection in the observation process, whereby the different points of view yield different left-of predicates. This they know; yet each observer otherwise regards himself as a perfect observer, with a completely self-consistent left-of predicate. By observing $15|15$ without $N(1,1,5,5)$ they know now (if they didn't already know) that they can't be using the same left-of predicate. On the other hand they have no a priori basis for preferring one predicate to the other; if they did, then the unpreferred observer may as well not bother to make an observation. These conditions are by no means atypical of the locally consistent but globally inconsistent way in which distributed information is gathered by distributed observers.

Now we can get this effect with just one observer by having him make two sequential observations. But the world does not stand still, which we shall reflect here by assuming that the bills move around. We take motion and observation to be discrete and synchronous, a conservatively traditional view of concurrent computation that avoids any timing subtleties that might be perceived as somehow favoring pomsets. We alternate motion and observation, permitting each observer only one observation between consecutive moves. Observations by two observers in the one clock tick will be of the same configuration albeit from possibly different points of view. Two observations by one observer however will be of different configurations, which therefore cannot be meaningfully pooled the way we pool two observations of the one configuration. In this dynamic setup we will see many configurations, but if ever the above configuration arises the joint observation for that observation period will be 15|15 without $N(1,1,5,5)$. A reliable solo observer however is incapable of such an observation, on that occasion or any other.

One more concern. If the observers' perception of relationships between objects or events is imperfect why are we justified in assuming perfect observation of individual objects or events?

In some situations we may get perfect information about both, in others neither. But we should expect observers working in different frames of reference to find it intrinsically harder to achieve agreement on relationships, especially temporal relationships between ephemeral events, than on which events happened and what entities exist. Hence one may expect our premise, of unreliably observed relations between reliably observed entities, to be met sufficiently often as to render our conclusions of some interest.

Our main result then is that under such circumstances we must anticipate team sightings of nonlinear pomsets. Moreover we must anticipate any pomset of dimension up to the size of the team. While we can infer from a report of a nonlinear pomset a discrepancy between two observers somewhere, we do not attempt to compensate for such discrepancies but instead treat them as part of our knowledge of what was observed and integrate them into our model of the world.

3 Background on Pomsets

Linearly ordered multisets (labelled chains up to isomorphism) are strings. Pomsets as partially ordered multisets therefore constitute a generalization of strings to partial orders. This model as an extension of formal language theory is due to Grabowski [Gra] who called it a partial word, the characterization as a partially ordered multiset being due to Pratt [Pr82]. Pomsets with a conflict relation are called event structures, introduced by Nielsen, Plotkin, and Winskel [NPW]. Prior related notions are Mazurkiewicz's partial monoids [Maz] and Greif's treatment of actors [Gre]. A list of more recent papers on the topic [MS,Gis,Pr86,AH,Wi] would be bound to be incomplete.

We shall identify observation with linearization. That is, at least in the case of atomic events, an observer of a pomset sees its events in some linear order consistent with the order of the pomset.

To a zeroth order approximation, two pomsets should be observationally equivalent when they have the same set of linearizations.

The familiar theorem that (the graph of) a poset is the intersection of the set of (graphs of) its linearizations is due to Szpilrajn [Sz]. In our framework posets are pomsets with no repeated elements, i.e. the function assigning labels to poset elements is injective. Thus in our application Szpilrajn's theorem states that distinct posets are not observationally equivalent.

At the other extreme from posets are pomsets over a one-letter alphabet, say the alphabet $\{a\}$. In our framework these amount to posets up to isomorphism. (So pomsets span a spectrum from posets-up-to-isomorphism to posets.) There are just two two-element pomsets over $\{a\}$, which we write as aa (linearly ordered) and $a|a$ (discretely ordered). These have the same set of linearizations and hence are observationally equivalent. So whereas Szpilrajn's theorem applies to posets this example shows that it does not apply to posets up to isomorphism.

The meaning of $a|a$ is that we have two copies of an activity a that are running in parallel. If a is an instantaneous event, as we have been assuming up to now, and the possibility of exact simultaneity is neglected, then there would seem to be no basis for distinguishing between aa and $a|a$ in either theory or practice.

If however a has duration (or length when a is a dollar bill) we have the possibility of overlap for the case $a|a$, but not for aa. We may represent duration by taking a to be a pomset of size two or more, e.g. the string 01. Then the only linearization of aa is 0101, whereas $a|a$ has for its linearizations both 0101 and 0011. Hence in the presence of events with duration it becomes possible to observe a difference between aa and $a|a$. A similar difference is observable if we take a to be $0|1$. In this case the linearizations of aa are 0101, 0110, 1001, and 1010, while those of $a|a$ are those four together with 0011 and 1100.

Gischer [Gis] shows that any two pomsets that are observationally equivalent for strings of length two are observationally equivalent for strings of any length, whence there is no duration hierarchy for strings. Gischer conjectured [Gis], and Tschantz has shown [Tsch], that duration suffices to distinguish any two series-parallel (N-free) pomsets. (A series-parallel pomset is a pomset constructible using only the operations of concatenation ab and concurrence $a|b$.) Hence series-parallel pomsets are extensional in the presence of duration. (Another striking corollary of this result is that the equational theory of concatenation and interleaving of languages is completely axiomatized by the equations for commutativity of interleaving and associativity of both.)

Gischer gives as an example of pomsets indistinguishable even with duration the two pomsets $N(a, a, b, b)$ and $ab|ab$, where $N(1, 2, 3, 4)$ is the 4-vertex pomset ordered so that $1 < 3$, $2 < 4$, and $1 < 4$, these constraints constituting respectively the two verticals and the diagonal of the letter N, so that $N(a, a, b, b)$ is $ab|ab$ plus the diagonal. As the reader may have already seen by now, if they could be distinguished it would have to be by a string of $ab|ab$ not allowed by $N(a, a, b, b)$, possible only by violating the diagonal $1 < 4$ of the N. Hence 1 and 4 overlap; where they do, 2 cannot have started but 3 must have finished, so the other diagonal $2 < 3$ is satisfied. But that diagonal belongs to an isomorphic copy of $N(a, a, b, b)$, whence that string must be allowed after all.

We may further take a to be not just a single string but a set of strings, that is, a language. This provides a notion of variety for a: we have a variety of choices of behaviors of a. When all strings of a are of unit length we have variety without duration. Variety provides those little unpredictable hints that can allow observers to reach consensus as to the identities of entities without them being a part of the observation language. In some observations the observers may be unlucky and not get enough such hints; it only matters that there exist observations that do provide sufficient hints.

Gischer's argument above remains valid in the presence of variety, giving a pair of pomsets which variety does not help distinguish.

Two minor results concerning refinements of observational equivalence in this setting are as follows.

(i) For a single observer, duration helps but variety does not.

(ii) For multiple observers to make a difference, variety without duration helps but duration without variety does not.

Our main result is:

(iii) With enough variety and observers any two finite pomsets can be distinguished, even without duration.

Results (i) and (ii) assign very different roles to duration and variety. Duration is a loner that can help, though not always, as evidenced by Gischer's example above of $N(a, a, b, b) = ab|ab$. Variety on the other hand is useless by itself but in collaboration with multiple observers is able not only to outperform duration but, as (iii) shows, to make pomsets fully visible, i.e. extensional. The proof of (iii) is via a straightforward reduction to the poset case, allowing us to apply Szpilrajn's theorem.

A refinement of (iii) is that with enough variety, the number of observers needed to distinguish two pomsets is at most the larger of the dimensions of their underlying posets.[1] This shows that the hierarchy of observational equivalences with n observers is strict: $n+1$ observers can resolve more detail than n. Although our proof of this result is not long, neither is it at all obvious!

4 Definitions

The following notions are essentially as in [Gis]. We start out by defining labelled partial orders and their maps.

Definition 1. A *labelled partial order* or *lpo* over a set Σ is a structure $(V, \leq, \sigma, \Sigma)$ where $\leq$ partially orders V and $\sigma : V \to \Sigma$ assigns to each element of V an element of Σ. When necessary we write the components of lpo p as $(V_p, \leq_p, \sigma_p, \Sigma_p)$.

We think of Σ as an alphabet of *actions* and V as instances of that alphabet, or *events* forming a word, with the order of occurrences of letters in the word given by $\leq$. The usual formal language theoretic notion of a word obtains for $\leq$ linear. An atomic lpo is one with $|V| = 1$.

Definition 2. A *map* of lpo's $(f, t) : (V, \leq, \sigma, \Sigma) \to (V', \leq', \sigma', \Sigma')$ consists of a monotone map $f : (V, \leq) \to (V', \leq')$ of posets together with an alphabet map (function) $t : \Sigma \to \Sigma'$ such that for all v in V, $\sigma'(f(v)) = t(\sigma(v))$.

Certain maps of lpo's are of special interest here. An *isomorphism* of lpo's is a map (f, t) for which f is an isomorphism of posets and t is the identity map on Σ (so isomorphic lpo's have a common alphabet). An *augmentation* of lpo's is a map (f, t) for which t is the identity function and f is the identity function on the elements of the poset (but not necessarily an isomorphism of posets, i.e. the order may increase); an augmentation yields an *augment* of its argument. We write $p\alpha q$ to indicate that q is an augment of p; this is the converse of Gischer's *subsumption* relation $q \succ p$ [Gis].

Definition 3. A *pomset* is the isomorphism class of an lpo.

More intuitively a pomset is an lpo in which we pay no attention to the choice of the set V, other than its cardinality, but retain all other details. Thus if we replace $V = \{0, 1, 2\}$ by $V = \{5, 6, 7\}$

[1] The dimension of a poset is the least number of linearizations of that poset whose intersection is that poset. The notion is due to Dushnik and Miller [DM], see Kelly and Trotter [KT] for a survey.

without otherwise disturbing either $\leq$ or σ the pomset does not change. With our definition of observation, isomorphic lpo's will be seen to be observationally equivalent, whence the most we can hope to resolve even with multiple observers is pomsets.

We shall understand a map between two pomsets to be a map between representative lpo's of the respective pomsets.

Definition 4. A *process* P is a set of finite pomsets. A process is *augment closed* when for all $p\alpha q$, $p \in P$ implies $q \in P$. The *augment closure* $\alpha(P)$ of P is the least augment closed process containing P.

We wish to define observation in terms of the notions of *linearization* and *substitution*, which we now define.

Definition 5. A *linearization* of a pomset p is a linear augment of p. We write $\lambda(p)$ for the set of all linearizations of p. This extends to $\lambda(P)$ for P a set of pomsets, namely as $\lambda(P) = \bigcup_{p \in P} \lambda(p)$.

Formal language theory has the notions of homomorphism and substitution [HU]. These both generalize immediately from strings to pomsets. (This notion of homomorphism is quite different from that of map between two pomsets: the former goes between sets of pomsets, the latter between single pomsets.)

Definition 6. A *pomset homomorphism* is a function mapping pomsets on Σ to pomsets on Σ'. It is determined by a function f assigning a pomset on Σ' to each letter of Σ. It maps p to the pomset whose set of events is the disjoint sum of the events of the $f(\sigma(u))$'s over all $u \in V_p$, definable as $\{(u,v) | u \in V_p, v \in V_{f(\sigma(u))}\}$. Each (u,v) is labelled with $\sigma_{f(\sigma(u))}(v)$, i.e. just as v was labelled in $f(\sigma(u))$, and ordered so that $(u,v) \leq (u',v')$ just when $u <_p u'$ (i.e. $u \leq_p u'$ and $u \neq u'$) or ($u = u'$ and $v \leq_{f(u)} v'$), that is, lexicographic ordering.

Intuitively this is what is obtained by substituting a pomset for each label of p and flattening the resulting nested structure in the obvious way. For example the homomorphism taking a to bc takes aa to $bcbc$ and $a|a$ to $bc|bc$, while the homomorphism taking a to $b|c$ takes aa to $(b|c)(b|c)$ and $a|a$ to $b|b|c|c$.

This generalizes to *substitutions* of sets of pomsets exactly analogously to the generalization of homomorphisms of strings to substitutions of sets of strings [HU], in which the result of substituting a set of strings for a letter is the set of all strings obtainable by choosing any string from each substitution instance of such a set. In lieu of a formal definition we offer the example of substituting the set $\{b,c\}$ for a in $a|a$, having two substitution instances of $\{b,c\}$ and so yielding the set of three pomsets $b|b$, $b|c$, $c|c$ ($c|b$ being isomorphic to $b|c$ as an lpo and hence equal as a pomset). Just as for formal languages, a homomorphism can be viewed as the special case of a substitution of singletons.

We may now regard pomsets as expressions, with the labels acting as variables. Evaluation is then just substitution: values for the variables determine the value of the expression. Thus the pomset aba is an expression with variables a and b, and if the value of a is cd and b is $\{e,f\}$ then the value of aba is $\{cdecd, cdfcd\}$. With this interpretation of substitution in mind we write $p(s)$ for the value of p under the substitution s. By $P(s)$ for a set P of pomsets we understand the union over the elements $p \in P$ of $p(s)$.

We might say that two pomsets are equivalent when their values are the same for all substitutions.

But merely taking the value of each variable to be itself already suffices to distinguish distinct pomsets, so this equivalence is trivially the identity relation.

The notion of observation as linearization, reflecting the sequential life of an individual observer, leads to more interesting equivalences. We tentatively define an observation of a pomset to be a linearization of it. Thus the set of all observations of p is $\lambda(p)$, and the set of all observations of a set P of pomsets is $\lambda(P)$. Pomsets p and q are *equivalent* when $\lambda(p(s)) = \lambda(q(s))$ for all substitutions s.

We now extend this notion of observation to multiple observers. The idea is that n observers see n possibly different linearizations of the one observed pomset.

Definition 7. An *n-observation* of a pomset p is an n-tuple of linearizations of p. We write $\lambda_n(p)$ for the set consisting of all n-observations of p, a set of n-tuples of strings. For a process P we take $\lambda_n(P) = \bigcup_{p \in P} \lambda_n(p)$.

Definition 8. Pomsets p and q are *n-equivalent*, written $p \equiv_k q$, when $\lambda_n(p) = \lambda_n(q)$. Likewise for processes, $P \equiv_k Q$ when $\lambda_n(P) = \lambda_n(P)$.

Our tentative definitions of observation and equivalence are now subsumed as 1-observation and 1-equivalence.

Implicit in our definition of n-equivalence is a consensus between the observers as to which pomset of P to linearize, when constructing an n-observation in $\lambda_n(P)$. This reflects our intuition that the observers agreed on what happened but not when.

Finally we need the notion of dimension [KT] in order to show the strictness of the hierarchy of n-equivalence in the presence of variety.

Definition 9. The *dimension* of a poset is the minimum number of its linearizations such that the intersection of those linearizations is that poset. We take the dimension of a pomset p to be the dimension of the underlying poset of a representative lpo of p.

5 Results

In order to capture duration, variety, etc. we need a parametrized notion of n-equivalence, parametrized by the permitted substitutions. If substitutions are restricted so that the assignment to any variable must come from a class C of sets of pomsets, e.g. singletons, sets of one-element pomsets, languages (sets of linear pomsets), we say that two pomsets are n-equivalent for C when they have the same n-observations of their values for all substitutions where the assignments to the variables are drawn from C.

In the following we are interested in substitutions that have variety without duration, and duration without variety. We denote these respective classes of substitutions by **Var** and **Dur** respectively. A substitution from **Var** can replace each label by a set of labels. A substitution from **Dur** can replace each label by a pomset. The class of substitutions permitting neither duration nor variety, corresponding to mere renamings of labels, we call **Atm** for atomic substitutions.

None of our results make essential use of nonlinearity in the substructure of events. For example if **Dur** is taken instead to consist of those substitutions that replace labels by strings rather than pomsets, no modifications are required to either the following propositions or their proofs.

The first few propositions are very simple, but give some insight into the respective roles played by duration and variety.

We first show that for a single observer, duration without variety helps but variety without duration does not.

Proposition 1. 1-equivalence for **Dur** is strictly finer than 1-equivalence for **Atm**.

Proof. It is finer because **Dur** includes **Atm**. The example of aa and $a|a$ shows strictness.

Proposition 2. 1-equivalence for **Var** coincides with 1-equivalence for **Atm**.

Proof. This follows from $\lambda(p(s)) = (\lambda(p))(s)$. That is, we can substitute sets for variables in p and then linearize, or linearize p first (yielding a language) and then substitute, with the same result in either case. Hence $\lambda(p(s)) = (\lambda(p))(s) = (\lambda(q))(s) = \lambda(q(s))$.

We now come to the main result. The next two propositions show that for multiple observers to make a difference, variety without duration helps but duration without variety does not. The former, proposition 3, is the main result in that it shows that any two pomsets can be distinguished by n observers for sufficiently large n. It is noteworthy that duration plays no role in this result! Since our first explorations in this area focused on the role of duration in distinguishing pomsets we did not at first expect such a result. In retrospect it is not so surprising, nor particularly deep, being a straightforward reduction to Szpilrajn's theorem..

Proposition 3. For any pomset p there exists n such that p is not n-equivalent for **Var** to any other pomset.

Proof. We use variety to distinguish the otherwise indistinguishable events of a pomset. Let m be the size of p. We take n to be $m!$. Consider the substitution s mapping each letter a of Σ to the m-element set $\{(a,i)|0 \leq i < m\}$. This is enough variety for $p(s)$ to include at least one poset, call it q. Then $\lambda(q)$ has at most $m!$ members, whence some $m!$-tuple of $\lambda_{m!}(q)$ will contain all of them. This gives us a procedure for recovering p from $\lambda_{m!}(p(s))$. Discard $m!$-tuples of $\lambda_{m!}(q)$ not corresponding to posets (repeated letters). From the remainder select any $m!$-tuple with a maximum number of different components, an $m!$-observation of some poset q. Use Szpilrajn's theorem to infer q from the $m!$-observation. Replace each label (a,i) by a in q, to yield p. This construction shows that the p so recovered will be independent of the choice of poset from $p(s)$.

Proposition 4. 1-equivalence for **Dur** coincides with n-equivalence for **Dur**.

Proof. In this case $p(s)$ is a singleton, substitutions being homomorphisms, for which $\lambda_n(p(s))$ is the set of all n-tuples of linearizations of the pomset $p(s)$. Hence $\lambda_n(p(s))$ can be computed from $\lambda(p(s))$. Thus if $\lambda(p(s)) = \lambda(q(s))$, we must have $\lambda_n(p(s)) = \lambda_n(q(s))$ as well.

The argument for proposition 3 can be extended to show that, for any class including **Var**, n-equivalence for increasing n forms a strict hierarchy. Our particular witnesses to this hierarchy are independent of the class of substitutions. Although the proof of this proposition may not seem very long, its brevity belies its subtlety, as the reader may verify.

Proposition 5. For every $n > 1$ there exist pomsets p and q such that for any class C of substitutions including **Var**, p and q are n-1-equivalent for C but not n-equivalent for C.

Proof. It suffices to consider pomsets over a one-letter alphabet, i.e. posets up to isomorphism. (Note that Szpilrajn's theorem separates even isomorphic posets, and cannot be applied directly here.) Given n we take for our counterexample a certain pair p, q of posets of dimension n. Using essentially the same argument as in Proposition 3 we show that as one-letter pomsets p and q cannot be n-equivalent for **Var**, and hence for any larger class. We then show that they are n-1-equivalent for any class.

We take p to be the *standard* poset S_n [KT], having $2n$ elements $\{a_0, \ldots, a_{n-1}, b_0, \ldots, b_{n-1}\}$, ordered so that $a_i \leq b_j$ just when $i \neq j$. An equivalent description of S_n is as the lattice of atoms and coatoms of an n-atom Boolean algebra. S_n is known to have dimension n [KT]. We take q to be S_n augmented with $a_0 \leq b_0$. (As pomsets, p and q are determined only up to isomorphism, so augmenting p with $a_i \leq b_i$ for any i yields the same pomset q.) Since q has $2n$ elements it is of dimension at most n [KT]. Hence p and q are not n-equivalent for **Var**. The role of **Var** here is as for Proposition 3, namely allowing us to treat pomsets as posets.

For n-1-equivalence, suppose some linearization of an element of $p(s)$ violates $a_i \leq b_i$ for some i, necessary if we are to distinguish p and q. Then there is a point in that string where a_i has not yet finished (a_i could have duration in the general case) yet b_i has started. The constraints of p require that at that point all the other a_j's are done (for b_i to start) and none of the other b_j's have started (since a_i is not yet done). Hence for every $j \neq i$, $a_j \leq b_j$, that is, there can be at most one violation of $a_i \leq b_i$ for any i in any one linearization. But then any n-1-observation of $p(s)$ can collectively violate at most $n-1$ of the constraints of the form $a_i \leq b_i$. This always leaves one such constraint unviolated, which is consistent with observing q. Hence the n-1-observations of $p(s)$ must coincide with those of $q(s)$ for all s.

Proposition 6. Observationally equivalent processes have equal augment closures.

Proof. Any behavior p of a process P must be visible to a team of size $\dim(P)$. If Q is observationally equivalent to P the same team must be able to observe p as an apparent behavior of Q. Hence Q must contain a behavior q of which p is an augment, whence $P \subseteq \alpha(Q)$. By symmetry of equivalence $Q \subseteq \alpha(P)$, whence $\alpha(P) = \alpha(Q)$.

Bibliography

[AH] S. Anderson and P. Hudak, Pomset Interpretations of Parallel Functional Programs, Proc. 3rd International Conf. on Functional Programming Languages and Computer Architecture, Portland, OR, Sept. 1987.

[BG] J. Baeten and R. van Glabbeek, Another look at abstraction in process algebra, ICALP-87, LNCS-267, 84-94, 1987.

[BHR] S. Brookes, C. Hoare and A. Roscoe, A Theory of Communicating Sequential Processes, JACM 31:3, 560-599, 1984.

[DM] B. Dushnik and E.W. Miller, Partially Ordered Sets, *Amer. J. Math.* **63**, 600-610, 1941.

[Gis] J. Gischer, **The equational theory of pomsets**, Theoretical Computer Science, **61**, 199-224, 1988.

[Gra] J. Grabowski, On Partial Languages, *Fundamenta Informaticae* IV.2, 427-498, 1981.

[Gre] I. Greif, Semantics of Communicating Parallel Processes, Ph.D. Thesis, Project MAC report

TR-154, MIT, September, 1975.

[Hoa] C. Hoare, **Communicating Sequential Processes**, Prentice-Hall, 1985.

[HU] J. Hopcroft and J. Ullman, **Introduction to Automata and Computability**, Addison-Wesley, 1979.

[Jon] B. Jonsson, A fully abstract trace model for dataflow networks, Proc. 16th Annual ACM Symposium on Principles of Programming Languages, 155-165, Jan. 1989.

[KT] D. Kelly and W.T. Trotter, Dimension Theory for Ordered Sets, in **Ordered Sets** (ed. I. Rival), 171-211, D. Reidel, 1982.

[Maz] A. Mazurkiewicz, Traces, Histories, Graphs: Instances of a Process Monoid, Proc. Conf. on Mathematical Foundations of Computer Science, Springer-Verlag LNCS 176, 1984.

[Mi80] R. Milner, **A Calculus of Communicating Systems**, LNCS-92, Springer-Verlag, 1980.

[Mi84] R. Milner, Lectures on a Calculus for Communicating Systems, in **Seminar on Concurrency** (eds S. Brookes, A. Roscoe and G. Winskel), LNCS-197, 197-220, Springer-Verlag, 1984.

[MS] U. Montanari and C. Simonelli, On distinguishing between concurrency and nondeterminism, Proc. Ecole de Printemps on Concurrency and Petri Nets, Colleville, 1980.

[NPW] M. Nielsen, G. Plotkin, and G. Winskel, Petri Nets, Event Structures and domains, Part I, TCS 13, 1981.

[Papa] C. Papadimitriou, **The Theory of Database Concurrency Control**, Computer Science Press, 1986.

[Park] D. Park, Concurrency in Automata on Infinite Sequences, Proc. 5th GI Conf., LNCS-104, 167-183, 1981.

[Pr82] V. Pratt, On the Composition of Processes, Proceedings of the Ninth Annual ACM Symposium on Principles of Programming Languages, Jan. 1982.

[Pr86] V. Pratt, Modelling Concurrency with Partial Orders, International Journal of Parallel Programming, **15**:1, 33-71, Feb. 1986.

[Rus] J. Russell, Full Abstraction for Nondeterministic Dataflow Networks, Proc. 30th IEEE Symposium on Foundations of Computer Science, Oct. 1989.

[Sz] E. Szpilrajn, Sur l'extension de l'ordre partiel, *Fund. Math.* **16**, 386-389, 1930.

[Tsch] S. Tschantz, Languages under Concatenation and Shuffling, preliminary report, Dept. of Math., Vanderbilt U., 1988.

[Wi] G. Winskel, Events in Computation, Ph.D. Thesis, CST-10-80, Dept. of Computer Science, University of Edinburgh, 1980. 1986.