In: In H. Kirchner, editor, "Trees in Algebra and Programming — CAAP '96," LNCS 1059, pp. 272–286. Springer-Verlag, 1996.

Denotational, Causal, and Operational Determinism in Event Structures

Arend Rensink
Institut für Informatik, University of Hildesheim*

Abstract. Determinism of labelled transition systems and trees is a concept of theoretical and practical importance. We study its generalisation to event structures. It turns out that the result depends on what characterising property of tree determinism one sets out to generalise. We present three distinct notions of event structure determinism, and show that none of them shares all the pleasant properties of the one concept for trees.

Keywords: behavioural models, event structures, concurrent languages, determinism, partial order models, trees.

1 Introduction

Consider the class of edge-labelled *trees*, i.e., labelled transition systems in which the transition relation induces a tree ordering over the states. A *path* in a tree is an alternating sequence of states and labels starting in the initial (smallest) state; a *word* is the corresponding sequence of labels only. A tree is called *deterministic* if from every state there is at most one transition with any given label. The following properties are easily seen to hold:

– A tree is deterministic if and only if each of its words corresponds to a unique path;
– Every tree can be collapsed to a deterministic tree with the same set of words, which is unique up to isomorphism.

In fact, either of these properties can be used to formulate an alternative, equivalent definition of the property of determinism in trees. Under a suitable notion of tree morphism, these properties are combined in the following category theoretic result (which is in fact relatively robust with respect to the choice of morphism):

– Deterministic trees form a reflective subcategory of trees, where the underlying functor is language-preserving.

Whereas trees have been used very successfully to model the (in general) *nondeterministic* behaviour of systems, to capture at the same time the nondeterministic *and concurrent* aspects of system behaviour, a widely accepted model

* Postfach 101363, D–31113 Hildesheim; email: rensink@informatik.uni-hildesheim.de.
Research partially supported by the HCM Network "EXPRESS" (Expressiveness of Languages for Concurrency). For a report version with full proofs see [8].

is that of *event structures*, introduced originally to model Petri net unfoldings (cf. Nielsen, Plotkin and Winskel [4]). That is, trees model the concurrent execution of actions by representing all their linear orderings, and thus do not truly capture the inherent concurrency. The "words" of event structures, on the other hand, are not sequences but *partially ordered multisets* (pomsets) of labels (called *concurrent words* in the sequel); consequently, sequential and concurrent executions are distinguished.

It might be expected that the notion of determinism can be extended easily from trees to event structures; in particular, that its various characterisations discussed above generalise smoothly. As it turns out, however, this is not the case. Rather, one may distinguish *three* kinds of determinism, resulting from the three alternative definitions referred to above; the category theoretical result does not hold fully with respect to any of the resultant properties, although it can be recovered partially for subclasses of event structures.

- For every event structure, there is a *denotationally* deterministic event structure with the same concurrent words, which is unique up to isomorphism. The concurrent words of denotationally deterministic event structures can be arbitrary.
- An event structure is called *causally* deterministic if every concurrent word uniquely corresponds to a run. The concurrent words of causally deterministic event structures are such that distinct events must either have distinct sets of causal predecessors or distinct labels.
- An event structure is called *operationally* deterministic if from every state, at most one event may occur with any given label. The concurrent words of operationally deterministic event structures are actually *auto-sequential*, meaning that equilabelled events are totally ordered; moreover, no distinct concurrent words have a common *linearisation*.

Operational determinism has been studied before in several contexts: in the series of papers [11, 12, 5, 10], Sassone with Nielsen and Winskel put operationally deterministic event structures in a categorical framework with other behavioural models, whereas Vaandrager showed in [13] that such event structures have precisely the expressive power of step sequences. We studied causally deterministic event structures in [7], presenting a complete equational theory for them. To our knowledge, denotational determinism has not been investigated before.

2 Definitions

This section defines a number of more or less standard concepts that are used in the remainder of the paper. Throughout the paper, we assume a universe $\mathbf{E}$ of events, ranged over by d, e, and a universe $\mathbf{A}$ of actions, ranged over by a, b, c.

Labelled transition systems, trees, paths and words. A *labelled transition system* is a tuple $T = \langle S, \rightarrow, \iota\rangle$ where S is a set of *states*, $\rightarrow \subseteq S \times \mathbf{A} \times S$ is a *transition relation* and $\iota \in S$ is the *initial state*. We write $s \xrightarrow{a} s'$ for $(s, a, s') \in \rightarrow$. A *path* in T is a sequence $s_0 a_0 s_1 \cdots a_{n-1} s_n$ for some $n \in \mathbb{N}$, where $s_0 = \iota$ and $s_i \xrightarrow{a_i} s_{i+1}$ for all $0 \leq i < n$; the sequence $a_0 \cdots a_{n-1}$ is then called a *word* of T. T is a *tree* if every $s \in S$ is the final state of precisely one path. T is called *deterministic* if $s \xrightarrow{a} s_1$ and $s \xrightarrow{a} s_2$ implies $s_1 = s_2$.

There is a standard notion of morphism that turns the class of trees into a category $\mathbf{T}$, with as a subcategory the deterministic trees, $\mathbf{T}_{\mathrm{d}}$. On the other hand, one can define a category $\mathbf{L}$ of *languages* (i.e., prefix closed sets of sequences over $\mathbf{A}$). The following properties can be seen to hold with respect to these categories (cf. Nielsen, Sassone and Winskel [5]):

Proposition 1. $\mathbf{L}$ *is equivalent to* $\mathbf{T}_{\mathrm{d}}$.

Proposition 2. *There is a language-preserving reflection from* $\mathbf{T}$ *to* $\mathbf{T}_{\mathrm{d}}$.

It is the existence of a like situation for event structures that we investigate in this paper. Note that the condition of language preservation in the latter proposition was not taken as essential in [5], and indeed does not generally hold in the framework presented there. It is open for discussion to what degree language preservation is, or should be, an inherent property of determinisation. We return to this issue in the conclusion of the paper.

Event structures and morphisms. We define prime event structures with general conflict; see Winskel [14]). An *event structure* is a tuple $\mathcal{E} = \langle E, <, \mathrm{Coh}, \ell \rangle$ where $E \subseteq \mathbf{E}$ is a set of events, $< \subseteq E \times E$ an irreflexive and transitive *causal ordering* such that $\{d \in E \mid d < e\}$ is finite for all $e \in E$, $\mathrm{Coh} \subseteq \mathbf{2}^E$ is a set of finite sets of events representing a multi-ary *coherence predicate*, such that $F \subseteq G \in \mathrm{Coh}$ implies $F \in \mathrm{Coh}$ and $d < e \in F \in \mathrm{Coh}$ implies $F \cup \{d\} \in \mathrm{Coh}$, and $\ell: E \to \mathbf{A}$ is a *labelling function*. We denote $d \# e$ for $\{d, e\} \notin \mathrm{Coh}$ and $\#^=$ for the reflexive closure of $\#$, and $\leq$ for the reflexive closure of $<$. Finally, d and e are called *concurrent* if they are neither causally ordered nor conflicting. We use the following notation for the predecessors, resp. the proper predecessors of a set $F \subseteq E$:

$$\begin{aligned} \lceil F \rceil_{\mathcal{E}} &:= \{d \in E \mid \exists e \in F.\, d \leq e\} \\ \llbracket F \rrbracket_{\mathcal{E}} &:= \lceil F \rceil_{\mathcal{E}} - F \ . \end{aligned}$$

We use $\lceil e \rceil_{\mathcal{E}}$ and $\llbracket e \rrbracket_{\mathcal{E}}$ to abbreviate $\lceil \{e\} \rceil_{\mathcal{E}}$ and $\llbracket \{e\} \rrbracket_{\mathcal{E}}$, respectively. We use $E_{\mathcal{E}}$, $<_{\mathcal{E}}$, $\mathrm{Coh}_{\mathcal{E}}$ and $\ell_{\mathcal{E}}$ to denote the components of an event structure $\mathcal{E}$, but omit indices when they are clear from the context. An function ϕ is an *isomorphism* from $\mathcal{E}$ to $\mathcal{F}$, denoted $\phi: \mathcal{E} \cong \mathcal{F}$, if ϕ is a bijection from $E_{\mathcal{E}}$ to $E_{\mathcal{F}}$ such that $d <_{\mathcal{E}} e \Leftrightarrow \phi(d) <_{\mathcal{F}} \phi(e)$, $F \in \mathrm{Coh}_{\mathcal{E}} \Leftrightarrow \phi(F) \in \mathrm{Coh}_{\mathcal{F}}$ and $\ell_{\mathcal{E}}(e) = \ell_{\mathcal{F}}(\phi(e))$ for all $d, e \in E_{\mathcal{E}}$ and $F \subseteq E_{\mathcal{E}}$. $\mathcal{E}$ and $\mathcal{F}$ are then called *isomorphic*, denoted $\mathcal{E} \cong \mathcal{F}$. The *restriction* of an event structure $\mathcal{E}$ to a set of events $F \subseteq E$ is defined by

$$\mathcal{E} \restriction F := \langle F, < \cap (F \times F), \mathrm{Coh} \cap \mathbf{2}^F, \ell \restriction F \rangle \ .$$

See Fig. 1 for some examples of event structures, where the arrows represent causality and the dotted lines conflict. The notation ${}_1a$ etc. denotes the event 1

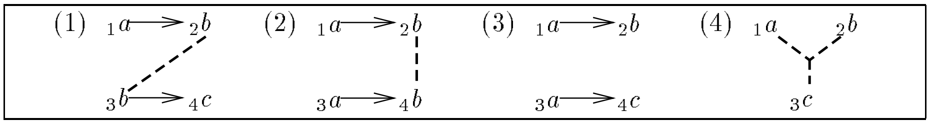

Fig. 1.: Some event structures. In (4), the set $\{1,2,3\}$ is conflicting, but all its proper subsets are coherent.

labelled with the action a (where we assume $\mathbb{N} \subseteq \mathbf{E}$). We omit events when the structure is to be interpreted modulo isomorphism.

In order to state our results in a category theoretic setting, we define a notion of *event structure morphism*. In this, we deviate from the standard notion of Winskel [14] and Nielsen, Sassone and Winskel [5], because we want to highlight the issue of determinism in isolation, rather than regarding it in combination with concurrency. To be precise, our morphisms are more restricted than the standard ones, in that they are allowed to manipulate conflict but not causality. At the end of the paper (Sect. 6) we will discuss how the situation changes when the standard notion of morphism is used instead.

An event structure morphism $\mathcal{E}$ to $\mathcal{F}$ is a pair (λ, η) (notation: $(\lambda, \eta): \mathcal{E} \to \mathcal{F}$) where λ is a partial function from $\mathbf{A}$ to $\mathbf{A}$ and η a partial function from $E_{\mathcal{E}}$ to $E_{\mathcal{F}}$, such that for all $e \in E_{\mathcal{E}}$, $\eta(e)$ is defined iff $\lambda(\ell_{\mathcal{E}}(e))$ is defined, in which case $\ell_{\mathcal{F}}(\eta(e)) = \lambda(\ell_{\mathcal{E}}(e))$; moreover, η preserves and reflects sets of predecessors (i.e., $\forall e \in \operatorname{dom} \eta . \eta(\llbracket e \rrbracket_{\mathcal{E}}) = \llbracket \eta(e) \rrbracket_{\mathcal{F}}$),[2] is non-injective only on conflicting events (i.e., $\forall d, e \in \operatorname{dom} \eta .\ \eta(d) = \eta(e) \implies d \#^{=}_{\mathcal{E}} e$), and preserves coherency (i.e., $\forall F \subseteq E_{\mathcal{E}}.\ F \in \mathrm{Coh}_{\mathcal{E}} \implies f(F) \in \mathrm{Coh}_{\mathcal{F}}$). Event structures and their morphisms, with identity morphisms $(id_{\mathbf{A}}, id_{\mathcal{E}})$ for all $\mathcal{E}$ and pairwise composition of morphisms, trivially give rise to a category **ES**. Note that the resulting notion of *iso*morphism coincides with the one presented explicitly above; that is, $(id, \eta): \mathcal{E} \to \mathcal{F}$ is an isomorphism iff $\eta: \mathcal{E} \cong \mathcal{F}$.

Partially ordered sets and multisets. A *labelled partially ordered set* (lposet) is a finite event structure without conflict; i.e., a triple $p = \langle E, <, \ell \rangle$ (where the conflicting sets are omitted altogether). The notion of isomorphism is inherited from event structures. p is a *prefix* of an lposet q, denoted $p \preceq q$, if $E_p \subseteq E_q$ is left-closed according to $<_q$ ($d \in E_p \implies d <_q e \in E_p$) and $p = q \upharpoonright E_p$. An lposet p is called *topped* if it has a greatest (top) element $\top_p \in E_p$ (i.e., $\forall d \in E_p . d \leq \top_p$).

A *partially ordered multiset* (pomset) is an isomorphism class of lposets $[p]_{\cong} = \{q \mid q \cong p\}$; we usually denote $[p]_{\cong}$ by $[p]$. The concept of lposet prefix is lifted to pomsets: $[p] \preceq [q]$ if $p \cong p' \preceq q$ for some lposet p'.

A *concurrent language* $\mathcal{L}$ is a prefix closed sets of pomsets (i.e., $[p] \preceq [q] \in \mathcal{L} \implies [p] \in \mathcal{L}$). Concurrent languages give rise to a category **CL** where morphisms are partial functions λ from $\mathbf{A}$ to $\mathbf{A}$, which are extended to functions $\hat{\lambda}$ from pomsets to pomsets by defining $\hat{\lambda}([p]) = \langle E, <_p \upharpoonright E, \lambda \circ (\ell_p \upharpoonright E) \rangle$ with $E = \ell_p^{-1}(\operatorname{dom} \lambda)$. (Note that this is well-defined modulo the choice of representative p.) Then λ is a morphism from $\mathcal{L}$ to $\mathcal{M}$ (notation: $\lambda: \mathcal{L} \to \mathcal{M}$) iff $\hat{\lambda}(\mathcal{L}) \subseteq \mathcal{M}$.

[2] In contrast, standard morphisms satisfy $\eta(\llbracket e \rrbracket) \supseteq \llbracket \eta(e) \rrbracket$.

Configurations, event transitions and concurrent languages. A *configuration* of an event structure $\mathcal{E}$ is a coherent (and therefore finite) set $F \in \mathrm{Coh}$ which is $<$-left-closed (i.e., $d < e \in F \implies d \in F$). The configurations of $\mathcal{E}$ are collected in $\mathcal{C}(\mathcal{E})$. $\mathcal{E}$ thus naturally gives rise to the tree $es.t(\mathcal{E}) = \langle \mathcal{C}(\mathcal{E}), \rightarrow, \varnothing \rangle$ where for all $F, G \in \mathcal{C}(\mathcal{E})$, $F \xrightarrow{a} G$ iff $G = F \cup \{e\}$ for some $e \notin F$ such that $\ell_{\mathcal{E}}(e) = a$. A *concurrent word* of $\mathcal{E}$ is a pomset $[p]$ such that E_p is a configuration of $\mathcal{E}$ and $p = \mathcal{E} \upharpoonright E_p$. The concurrent words of $\mathcal{E}$ are collected in $es.cl(\mathcal{E})$. It is clear that $es.cl(\mathcal{E})$ is prefix-closed, hence a concurrent language. For instance, the concurrent language corresponding to structure (1) of Fig. 1 is $\left\{ \epsilon, \boxed{a}, \boxed{a \to b}, \boxed{b}, \boxed{\begin{smallmatrix} a \\ b \end{smallmatrix}}, \boxed{b \to c}, \boxed{\begin{smallmatrix} a \\ b \to c \end{smallmatrix}} \right\}$. We have the following connection from event structures to concurrent languages:

Proposition 3. *The mapping* $es.cl\colon \mathbf{ES} \to \mathbf{CL}$ *gives rise to a functor, with arrow part* $(\lambda, \eta) \mapsto \lambda$.

3 Denotational determinism

We come to the first of our notions of event structure determinism. It is based on the idea that denotationally, a deterministic model is completely determined (up to isomorphism) by its concurrent language. We will show that for any event structure there is a denotationally deterministic event structure, unique up to isomorphism, with the same concurrent language. However, due to the possible presence of equilabelled events which are *causally indistinguishable*, in the sense of having the same set of proper predecessors, the construction of the denotationally deterministic event structure is not always straightforward. Causal indistinguishability is defined as follows:

$$d \sim e \;:\Leftrightarrow\; \ell(d) = \ell(e) \wedge \llbracket d \rrbracket = \llbracket e \rrbracket$$

We give the formal definition of denotational determinism here, and an extensive discussion afterwards.

Definition 4 (denotational determinism). An event structure $\mathcal{E}$ is called *denotationally deterministic* if the following conditions hold:

- for all $e \in E$, if $F \subseteq_{\mathrm{fin}} [e]_\sim$ then $F \in \mathrm{Coh}$.
- for all pairwise concurrent $F \in \mathrm{Coh}_{\mathcal{E}}$ and $d \sim e \in F$, there is an auto-isomorphism $\phi\colon \mathcal{E} \cong \mathcal{E}$ such that $\phi(e) = d$ and ϕ is the identity on $F - \{e\}$.

(The first condition cannot be simplified to $[e]_\sim \in \mathrm{Coh}$, because $[e]_\sim$ may be an infinite set.) The class of denotationally deterministic event structures will be denoted $\mathbf{ES}_{\mathrm{dd}}$. Consider the event structures in Fig. 2. They have the same concurrent language, namely $\left\{ \boxed{\begin{smallmatrix} a \to b \\ a \end{smallmatrix}}, \boxed{\begin{smallmatrix} a \\ a \to c \end{smallmatrix}} \right\}$ and prefixes; however, their choice structure is different. Neither (1) nor (2) is in any way deterministic, since in

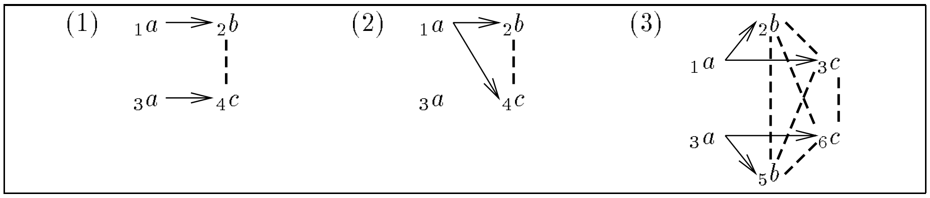

Fig. 2.: Event structures with the same concurrent languages

either case, when an a occurs, the choice of event (either 1 or 3) affects the possible continuations. Structure (3) does not share this characteristic, and indeed it is denotationally deterministic. In fact, structure (3) *determinises* the other two (where determinisation is the operation of constructing a deterministic event structure with the same concurrent language).

By the same token, even an event structure that contains no conflict may be nondeterministic, and to determinise it, conflict may have to be introduced; see structure (2) in Fig. 3. In contrast, if equilabelled events have different causal predecessors, such as 2 and 7 resp. 4 and 7 of structure (4) in Fig. 3, or have isomorphic continuations, such as 2 and 4 in the same structure, then this does not violate denotational determinism.

Basically, an event structure is denotationally deterministic if all causally indistinguishable events are *non-conflicting*, and moreover *isomorphic* in the sense that there is an auto-isomorphism of the entire event structure that maps them to each other. The operation of determinising a given event structure therefore consists of manipulating its causally indistinguishable events: if they are conflicting then they are merged, otherwise a copy of the "causal context" of each is added with respect to the other, so that they end up being isomorphic. This is illustrated by structures (1)–(3) of Fig. 3.

A necessary condition for denotational determinism is that every isomorphism between two configurations of an event structure (which therefore give

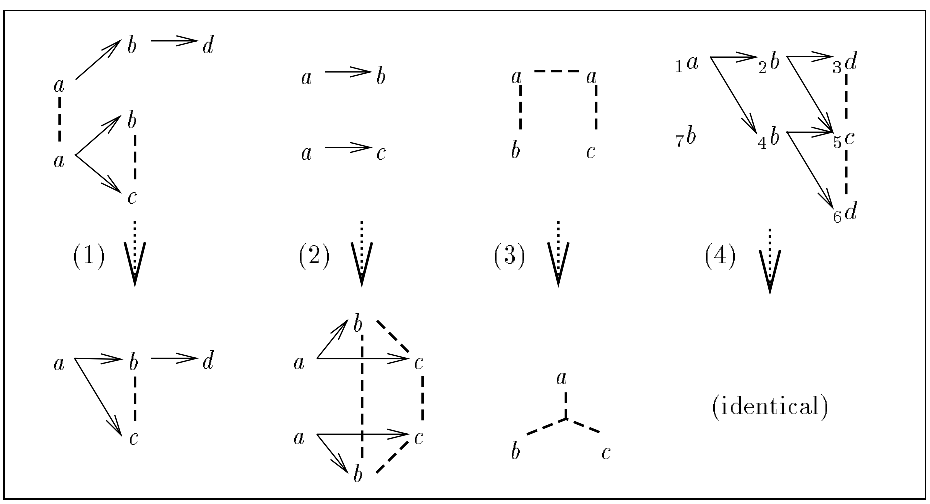

Fig. 3.: Event structures and their denotational determinisations

rise to identical concurrent words) can be extended to an isomorphism of the entire structure.

Proposition 5. *If $\mathcal{E}$ is denotationally deterministic then for all $F, G \in \mathcal{C}(\mathcal{E})$ such that $\phi: \mathcal{E} \upharpoonright F \cong \mathcal{E} \upharpoonright G$ there is a $\psi: \mathcal{E} \cong \mathcal{E}$ such that $\psi \upharpoonright F = \phi$.*

One of the crucial consequences of denotational determinism is that there exists a denotationally deterministic event structure for every concurrent language. This is proved in the following theorem.

Theorem 6. *$\mathcal{L} \in \mathbf{CL}$ iff $\mathcal{L} = es.cl(\mathcal{E})$ for some $\mathcal{E} \in \mathbf{ES}_{\mathrm{dd}}$.*

Proof. The "if" part is trivial. For the "only if", we give the construction of $\mathcal{E}$ through a series of approximants $\mathcal{E}_i = \langle E_i, <_i, \mathrm{Coh}_i, \ell_i \rangle$ for $i \in \mathbb{N}$, by induction on the *depth* of events (where the depth of $e \in E_{\mathcal{E}}$ equals the length of the longest chain $e_0 <_{\mathcal{E}} e_1 <_{\mathcal{E}} \cdots <_{\mathcal{E}} e$; hence initial events have depth 1). $\mathcal{E}_0$ is the empty structure; the construction of $\mathcal{E}_{i+1}$ from $\mathcal{E}_i$ and $\mathcal{L}$ is as follows.

Events. For all *topped* $[p] \in \mathcal{L}$ where $\top_p$ has depth $i+1$, let n be the least upper bound of the number of distinct, p-isomorphic prefixes of any element of $\mathcal{L}$, i.e., $n = \bigsqcup_{[q] \in \mathcal{L}} |\{e \in E_q \mid q \upharpoonright \lceil e \rceil \cong p\}|$. Note that $n \in \{1, \ldots, \infty\}$. Now for all $G \in \mathcal{C}(\mathcal{E}_i)$ such that $\mathcal{E}_i \upharpoonright G \cong p \upharpoonright \llbracket e \rrbracket$ and all $m < n$ let $(G, \ell_p(\top_p), m)$ be a new event of $\mathcal{E}_{i+1}$.

Orderings. For all new events (G, a, m), let $e <_{i+1} (G, a, m)$ iff $e \in G$.

Labels. For all new $(G, a, m) \in E_{i+1}$, let $\ell_{i+1}(G, a, m) = a$.

Coherence. For all $F \subseteq E_{i+1}$, let $p = \langle \lceil F \rceil, <_{i+1} \upharpoonright (\lceil F \rceil \times \lceil F \rceil), \ell_{i+1} \upharpoonright \lceil F \rceil \rangle$ be the smallest initial segment of $\mathcal{E}_{i+1}$ containing F; let $F \in \mathrm{Coh}_{i+1}$ iff $[p] \in \mathcal{L}$.

It follows that $\mathcal{E}_i = \mathcal{E}_{i+1} \upharpoonright E_i$ for all $i \in \mathbb{N}$; we define $\mathcal{E} = \bigcup_{i \in \mathbb{N}} \mathcal{E}_i$ by componentwise union of the approximants. The proof of $es.cl(\mathcal{E}) = \mathcal{L}$ is omitted. $\square$

The resulting mapping from concurrent languages to (deterministic) event structures will be denoted $cl.es: \mathbf{CL} \to \mathbf{ES}_{\mathrm{dd}}$. It follows that, in a sense, $\mathbf{ES}_{\mathrm{dd}}$ is large enough (namely to capture all concurrent languages). However, this is perforce also true of any *larger* class of event structure; for instance of the entire $\mathbf{ES}$. The following theorem, however, expresses the *dual* fact that $\mathbf{ES}_{\mathrm{dd}}$ is also, in a sense, *small* enough: its elements are completely determined (up to isomorphism) by their concurrent language. The proof is involved and omitted here.

Theorem 7. *For all $\mathcal{E}, \mathcal{F} \in \mathbf{ES}_{\mathrm{dd}}$, $es.cl(\mathcal{E}) = es.cl(\mathcal{F})$ iff $\mathcal{E} \cong \mathcal{F}$.*

It follows that every event structure can be *determinised* uniquely, in the sense that there exists an event structure, unique up to isomorphism, with the same concurrent language. The determinisation mapping will be denoted $es.des = cl.es \circ es.cl$.

Corollary 8. *For every $\mathcal{E} \in \mathbf{ES}$, $es.des(\mathcal{E}) \in \mathbf{ES}_{\mathrm{dd}}$ is unique up to isomorphism such that $es.cl(\mathcal{E}) = es.cl(es.des(\mathcal{E}))$.*

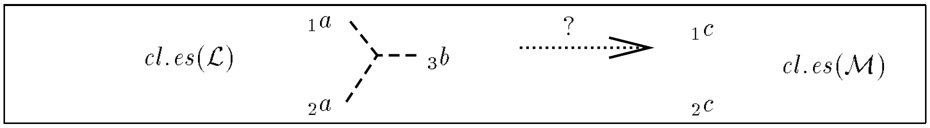

Fig. 4.: There is no morphism $cl.es(\mathcal{L}) \to cl.es(\mathcal{M})$

Denotational determinism categorically. So far for the positive results about denotational determinism. We now show that the role of the objects of $\mathbf{ES}_{\mathrm{dd}}$ as representatives of the concurrent languages of arbitrary event structures is rather superficial, in the sense that it cannot be generalised to a category theoretical setting. In particular, the one-to-one correspondence between concurrent languages and deterministic event structures modulo isomorphism does *not* give rise to an equivalence of categories. By the same token, the subcategory $\mathbf{ES}_{\mathrm{dd}}$ does *not* occupy any special position within **ES**.

In fact, the first surprise is that the mapping $cl.es: \mathbf{CL} \to \mathbf{ES}_{\mathrm{dd}}$ cannot even be extended naturally to a functor: there are morphisms $\lambda: \mathcal{L} \to \mathcal{M}$ for which no $(\lambda, \eta): cl.es(\mathcal{L}) \to cl.es(\mathcal{M})$ exists. This is due to the fact that the relabelling part of a morphism may map different topped pomsets onto the same one, which on the level of event structures gives rise to confusion about which indistinguishable events of the source are to be mapped onto which events of the target. Consider for instance $\lambda = (a \mapsto c, b \mapsto c)$, which is a morphism from $\mathcal{L} = \left\{\epsilon, \boxed{a}, \boxed{b}, \boxed{\begin{smallmatrix}a\\a\end{smallmatrix}}, \boxed{\begin{smallmatrix}a\\b\end{smallmatrix}}\right\}$ to $\mathcal{M} = \left\{\epsilon, \boxed{c}, \boxed{\begin{smallmatrix}c\\c\end{smallmatrix}}\right\}$. The corresponding deterministic event structures are given in Fig. 4, where $cl.es(\mathcal{L})$ is such that every pair of events is coherent, but the three together are conflicting. The coherence of every pair of events implies that no pair of events may be mapped to the same event of $cl.es(\mathcal{M})$, and hence no morphism exists.

If we disallow relabelling in morphisms (that is, $\lambda = id_{\mathbf{A}}$ always), a functor can be defined on the basis of $cl.es$, but even so $cl.es$ and $es.cl$ do not form an equivalence of categories. One way of explaining this is that deterministic event structures contain nonessential information (due to the copying of events in the context of causally indistinguishable events) that is accessible by morphisms; there are consequently too many morphisms, which on the level of concurrent languages collapse or disappear. For much the same reason, there does not exist a *reflection* from **ES** to $\mathbf{ES}_{\mathrm{dd}}$ —the existence of which is taken in [11, 5] as the *sine qua non* of a proper notion of determinisation. A more extensive discussion of this issue can be found in the full report version [8].

4 Causal determinism

We move to the second notion of determinism over event structures, called *causal*. It is stricter than denotational determinism, i.e., rules out certain denotationally deterministic models; the property that every event structure can be determinised is therefore automatically lost. However, causal determinism is much better behaved categorically, albeit only with respect to a subcategory of **ES**. Causal determinism was studied under the name of *determinism* in [6, 7]. (Note, however, that with exception of the characterisation theorem Th. 11, the results of this section are new.)

Definition 9 (causal determinism). An event structure $\mathcal{E}$ is called *causally deterministic* if for all $d, e \in E_{\mathcal{E}}$, $d \sim e$ implies $d = e$.

The class of causally deterministic event structures will be denoted $\mathbf{ES}_{\mathrm{cd}}$. For instance, of the event structures in Fig. 1, (1) and (4) are causally deterministic. The following is immediate.

Proposition 10. $\mathbf{ES}_{\mathrm{cd}} \subset \mathbf{ES}_{\mathrm{dd}}$.

Note that the inclusion is proper; structure (2) of Fig. 1 is an element of $\mathbf{ES}_{\mathrm{dd}} - \mathbf{ES}_{\mathrm{cd}}$. Below, we reconsider the results we established for denotational determinism in the current, more restrictive setting.

Causal determinism has the characterising property that there are no non-trivially isomorphic configurations in the model; in other words, the mapping from configurations to concurrent language is injective. This can be seen to rule out precisely the existence of distinct causally indistinguishable events; since these were the prime source of complications in the previous section, this is one indication why the categorical situation improves.

Theorem 11. *$\mathcal{E} \in \mathbf{ES}_{\mathrm{cd}}$ iff for all $F, G \in \mathcal{C}(\mathcal{E})$, $\mathcal{E} \upharpoonright F \cong \mathcal{E} \upharpoonright G$ implies $F = G$.*

The concurrent words of causally deterministic event structures are themselves causally deterministic; i.e., although they may contain concurrent events with the same label, those may not have precisely the same predecessors. Let $\mathbf{CL}_{\mathrm{cd}}$ denote the subclass of causally deterministic concurrent languages. We recall some facts about causally deterministic pomsets from [6] in order to facilitate proofs later on. For arbitrary finite sets P of causally deterministic pomsets, there is a least upper bound $\bigvee P$ with respect to pomset prefix. In fact, $\bigvee P$ is easily constructed, given an appropriate choice of representatives:

$$\bigvee P = \left[\textstyle\bigcup_{[p]\in P} E_p, \bigcup_{[p]\in \mathcal{L}} <_p, \bigcup_{[p]\in P} \ell_p\right]$$

where the representatives $[p], [q] \in P$ are chosen such that if $[\![d]\!] = [\![e]\!]$ and $\ell_p(d) = \ell_q(e)$ for some $d \in E_p, e \in E_q$ then $d = e$. (Such representatives exist by virtue of causal determinism.) Moreover, $[p] = \bigvee\{[q] \preceq [p] \mid [q] \text{ is topped}\}$ for all causally deterministic $[p]$. We now get the following counterpart to Th. 6.

Theorem 12. *$\mathcal{L} \in \mathbf{CL}_{\mathrm{cd}}$ iff $\mathcal{L} = es.cl(\mathcal{E})$ for some $\mathcal{E} \in \mathbf{ES}_{\mathrm{cd}}$.*

Proof. The "if" part is immediate. The "only if" in fact follows from Th. 6 and Th. 11; however, we give the construction of $\mathcal{E}$ explicitly for the present, much simpler case. We define

$$\begin{aligned}
E_{\mathcal{E}} &= \{[p] \in \mathcal{L} \mid [p] \text{ is topped}\} \\
<_{\mathcal{E}} &= \preceq \cap\, (E_{\mathcal{E}} \times E_{\mathcal{E}}) \\
\mathrm{Coh}_{\mathcal{E}} &= \{P \subseteq E_{\mathcal{E}} \mid \textstyle\bigvee P \in \mathcal{L}\} \\
\ell_{\mathcal{E}} &= \{([p], a) \in E_{\mathcal{E}} \times \mathbf{A} \mid a = \ell_p(\top_p)\}
\end{aligned}$$

$es.cl(\mathcal{E}) = \mathcal{L}$ due to the properties of causally deterministic pomsets: $\mathcal{E} \upharpoonright F \in \bigvee F$ for all $F \in \mathcal{C}(\mathcal{E})$ and $[p] = \bigvee \{[q] \in E_{\mathcal{E}} \mid [q] \preceq [p]\}$ for all $[p] \in \mathcal{L}$. □

Naturally, $\mathbf{ES}_{\mathrm{cd}}$ being properly smaller than $\mathbf{ES}_{\mathrm{dd}}$, not every event structure can be causally determinised while retaining its concurrent language. The class of event structures for which this is still possible will be called *causally distinct*.

Definition 13 (causal distinctness). An event structure $\mathcal{E}$ is called *causally distinct* if for all $d, e \in E_{\mathcal{E}}$, $d \sim e$ implies $d \;\#^{=}\; e$.

The class of causally distinct event structures will be denoted $\mathbf{ES}_{\mathrm{cdst}}$. Causal distinctness is easily seen to be equivalent to having only causally deterministic concurrent words. The following is the counterpart to Corollary 8.

Theorem 14. *An event structure $\mathcal{E}$ is causally distinct iff there exists a $\mathcal{F} \in \mathbf{ES}_{\mathrm{cd}}$, unique up to isomorphism, such that $es.cl(\mathcal{E}) = es.cl(\mathcal{F})$.*

Causal determinism categorically. With respect to the categorical situation, the properties that failed to hold in the general case turn out to be valid when regarded only for causally deterministic and causally distinct event structures.

Proposition 15. *$cl.es\colon \mathbf{CL}_{\mathrm{cd}} \to \mathbf{ES}_{\mathrm{cd}}$ gives rise to a functor, with arrow part given by $\lambda \mapsto (\lambda, \eta)$ where*

$$\eta\colon [p] \mapsto \begin{cases} \hat{\lambda}([p]) & \text{if } \lambda \text{ is defined on } \ell_p(\top_p) \\ \text{undefined} & \text{otherwise.} \end{cases}$$

The following is an event structure counterpart to the property for trees expressed in Prop. 1.

Theorem 16. *$(es.cl, cl.es)$ is an equivalence between $\mathbf{ES}_{\mathrm{cd}}$ and $\mathbf{CL}_{\mathrm{cd}}$.*

It suffices to show that given a partial mapping from actions to actions, there is at most one morphism between any pair of causally deterministic event structures. The following is an event structure counterpart to Prop. 2.

Theorem 17. *$es.des$ is the underlying functor of a concurrent-language-preserving reflection from $\mathbf{ES}_{\mathrm{cdst}}$ to $\mathbf{ES}_{\mathrm{cd}}$.*

5 Operational determinism

The last of the notions of determinism studied in this paper is the one obtained by observing the *transition structure* of the event structures in question, without taking causality into account. This makes for a stronger property than the previous two. Among other things, event structures can only be operationally deterministic if they contain *no auto-concurrency*, i.e., equilabelled events cannot be concurrent. Operationally deterministic event structures were studied under the name *deterministic event structures* in the aforementioned papers [11, 5]. A number of the results of this section are reconstructed from those papers.

Definition 18 (operational determinism). An event structure $\mathcal{E}$ is called *operationally deterministic* if the underlying transition system $es.t(\mathcal{E})$ is deterministic.

The class of operationally deterministic event structures is denoted $\mathbf{ES}_{\mathrm{od}}$. The following is immediate.

Proposition 19. $\mathbf{ES}_{\mathrm{od}} \subset \mathbf{ES}_{\mathrm{cd}}$.

Note that the inclusion is proper; structure (1) of Fig. 1 is an element of $\mathbf{ES}_{\mathrm{cd}} - \mathbf{ES}_{\mathrm{od}}$. Below, we reconsider the results we established for denotational determinism in the current, more restrictive setting.

Just how strong the property of operational determinism is has been made clear by Vaandrager in [13], where he shows that operationally deterministic event structures are isomorphic if and only if their set of *step sequences* (i.e., words over sets of actions rather than single actions) are equal. [13] also gives an easy characterisation of operational determinism in terms of the relations between events. Say that in some event structure $\mathcal{E}$, $d, e \in E$ are in *direct conflict*, denoted $d \#^! e$, if $d \# e$ and for all $d' \leq d$ and $e' \leq e$, $d' \# e'$ implies $d = e$ (in other words, no proper predecessors of d $[e]$ are in conflict with e $[d]$).

Theorem 20 (see [13, Propositon 3.8]). *$\mathcal{E}$ is operationally deterministic if for all $d, e \in E$ such that $\ell(d) = \ell(e)$, if $d \not\leq e \not\leq d$ then $d \# e$ and $\neg(d \#^! e)$.*

In [5] it is shown that the concurrent languages of operationally deterministic event structures are characterised by two properties: (i) no concurrent word may be *auto-concurrent*, and (ii) no pair of distinct concurrent words may share a *linearisation*.

- A pomset $[p]$ is said to be auto-concurrent if there are $d, e \in E_p$ such that $\ell(d) = \ell(p)$ and $d \not\leq e \not\leq d$.
- A linearisation of a pomset $[p]$ is a word $\ell_p(e_1) \cdots \ell_p(e_n)$, where all e_i are distinct such that $E_p = \{e_1, \ldots, e_n\}$ and $e_i <_p e_j$ implies $i < j$.

After [12], we call a concurrent language $\mathcal{L}$ a *semilanguage* if no $[p] \in \mathcal{L}$ is auto-concurrent and a *deterministic semilanguage* if in addition, for all $[p], [q] \in \mathcal{L}$, the existence of a sequence $a_1 \cdots a_n$ that linearises both $[p]$ and $[q]$ implies $p \cong q$. The subclass of deterministic semilanguages will be denoted $\mathbf{CL}_{\mathrm{ds}}$. The following is the counterpart of Th. 12; see also [5, Theorems 4.8 and 4.9].

Theorem 21. *$\mathcal{L} \in \mathbf{CL}_{\mathrm{ds}}$ iff $\mathcal{L} = es.cl(\mathcal{E})$ for some $\mathcal{E} \in \mathbf{ES}_{\mathrm{od}}$.*

Finally, we also characterise the class of event structures that can be determinised operationally (under preservation of the concurrent language). Call an event structure $\mathcal{E}$ *operationally distinct* if for all $d, e \in E_{\mathcal{E}}$, if $\ell(d) = \ell(e)$ then $d \not\leq e \not\leq d \Rightarrow d \# e$ and $d \#^! e \Rightarrow d \sim e$. The class of operationally distinct event structures will be denoted $\mathbf{ES}_{\mathrm{odst}}$. The following is the counterpart to Th. 14.

Theorem 22. *$\mathcal{E} \in \mathbf{ES}_{\text{odst}}$ iff $es.cl(\mathcal{E}) = es.cl(\mathcal{F})$ for some $\mathcal{F} \in \mathbf{ES}_{\text{od}}$.*

There are no new category theoretic results about operationally deterministic event structures that we had not already established for the larger class of causally deterministic ones. The adjunctions we had proved there (Theorems 16 and 17) simply specialise to the subcategories considered here. (However, see also Sect. 6 for a discussion of the effect that our choice of morphisms has had on these results.) For the sake of completeness we list the results below. They are special cases of [5, Theorem 4.10] and [5, Theorems 7.3 and 7.16], respectively, except for the phrase "concurrent-language-preserving" in Corollary 24.

Corollary 23. *$\mathbf{ES}_{\text{od}}$ and $\mathbf{CL}_{\text{ds}}$ are categorically equivalent.*

Corollary 24. *There is a concurrent-language-preserving reflection from $\mathbf{ES}_{\text{od}}$ to $\mathbf{ES}_{\text{odst}}$.*

6 Conclusion

Summary and discussion of the results. We have developed three notions of determinism for event structures, corresponding to three different characterisations of determinism of transition trees. For each of these we have investigated whether the category theoretical properties of deterministic trees, expressed in Propositions 1 and 2, can be extended to event structures. The results are summarised below.

Denotational determinism corresponds to the view that every event structure should give rise to a deterministic one, unique up to isomorphism, with the same concurrent language. In other words, the correspondence between concurrent languages and denotationally deterministic event structures is one-to-one. Unfortunately, the *determinisation* of a given event structure is nontrivial, involving the *duplication* of events. Mainly because of this duplication, denotationally deterministic event structures do not seem to exhibit many interesting categorical properties.

Causal determinism corresponds to the view that there should be a one-to-one correspondence between the runs of a (causally) deterministic behaviour and its (concurrent) words. Causal determinism is strictly stronger than denotational determinism; consequently, the ability to determinise any event structure is necessarily lost. Causally deterministic event structures share the categorical properties of deterministic trees that they are equivalent (as a category) to the corresponding concurrent languages and that they form a reflective subcategory of the causally distinct event structures. Another result, omitted from this paper for lack of space but included in the full report version [8], is that when event structures are converted into *causal trees* (see [1]) then the resulting causal tree is deterministic iff the event structure is causally deterministic.

Operational determinism corresponds to the view that from every state of the behaviour there should be at most one transition with any given label. For event structures, this was already known (see Vaandrager [13]) to correspond to the absence of auto-concurrency and direct conflict. Operational determinism is strictly stronger than causal determinism. The categorical properties of operationally deterministic event structures are those of causally deterministic ones, restricted to the appropriate subcategories. Hence in this respect operational determinism does not yield additional insight.

The categorical results mentioned above hold with respect to our chosen notion of morphism, which, as mentioned before, is more restrictive than the usual one. We briefly discuss how this has affected the outcome of our investigation.

The standard notion of event structure morphism (see [5, 14]) allows to *forget causality*, i.e., only requires $\eta(\llbracket e \rrbracket_{\mathcal{E}}) \supseteq \llbracket \eta(e) \rrbracket_{\mathcal{F}}$ rather than equality, as we have done. Then configurations $F, G \in \mathcal{C}(\mathcal{E})$ yielding identical concurrent words (i.e., such that $\mathcal{E} \upharpoonright F \cong \mathcal{E} \upharpoonright G$) can be mapped to non-isomorphic configurations of $\mathcal{F}$ (i.e., such that $\mathcal{F} \upharpoonright \eta(F) \not\cong \mathcal{F} \upharpoonright \eta(G)$), which situation cannot in general be reflected in their deterministic counterparts, since there F and G have just been collapsed in the process of determinisation. Summarised, this more general notion of morphism has the following effect on the results of this paper.

- The reflection of $\mathbf{ES}_{\mathrm{cdst}}$ in $\mathbf{ES}_{\mathrm{cd}}$ (Th. 17) is lost. Its restriction to the operational case (Corollary 24), however, still holds, as a corollary of a result proved in [5] which we recall below. Explained in terms of the discussion above, $[\mathcal{F} \upharpoonright \eta(F)]$ and $[\mathcal{F} \upharpoonright \eta(G)]$ have $[\mathcal{E} \upharpoonright \{e \in F \mid \eta(e) \text{ is defined}\}]$ as a common augmentation; hence if $\mathcal{F}$ is operationally deterministic then Th. 21 implies that $\mathcal{F} \upharpoonright \eta(F) \cong \mathcal{F} \upharpoonright \eta(G)$ after all.
- The equivalence of $\mathbf{ES}_{\mathrm{cd}}$ to the causal concurrent languages $\mathbf{CL}_{\mathrm{cd}}$ (Th. 16) can be generalised; we have worked this out in a separate paper [9]. Consequently, this is also true of its restriction to the operational case (Corollary 23) —which special case was in fact already proved in [12, 5].
- On the other hand, precisely because morphisms may forget causality, a reflection does not necessarily retain the concurrent language. In fact, one of the main results of [5] is that a reflection from the *entire* **ES** to $\mathbf{ES}_{\mathrm{od}}$ exists under these circumstances, which does perforce *not* preserve the concurrent language (instead, for every concurrent word of an event structure, a less ordered one is in the language of its determinisation).

It should be noted that, as a matter of course, the more restricted notion of morphism chosen in this paper affects some other categorical constructions as well. In particular, the *product* in our categories is no longer guaranteed to exist and hence can no longer be used to model synchronisation (although the coproduct still models choice). Indeed, in contrast to operationally deterministic event structures, causally and denotationally deterministic ones are not closed with respect to synchronisation.

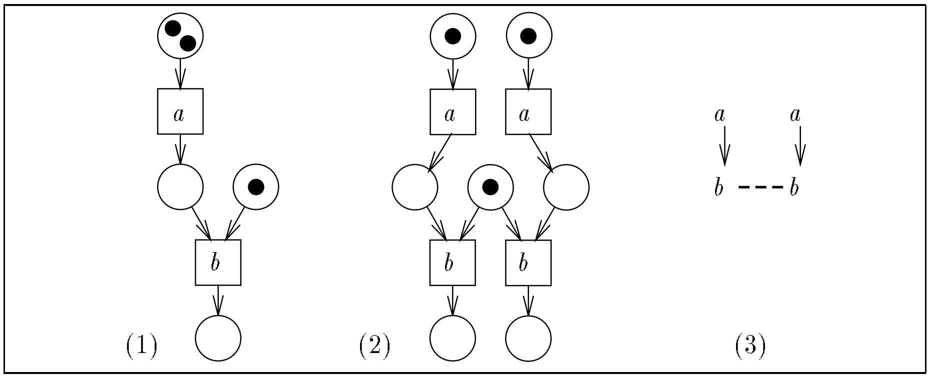

Fig. 5.: A non-safe Petri net (1), its naive unfolding (2), and the derived denotationally deterministic event structure (3)

Related work: Petri net unfoldings. Apart from the work of Sassone, Nielsen and Winskel on the one hand and Vaandrager on the other, discussed extensively above, a field of research from which there exists a somewhat tenuous connection to this paper is that of *Petri net unfoldings* according to the so-called *individual token philosophy*, as investigated by Engelfriet in [2] and by Sassone with Meseguer and Montanari in [3, 10]. The subclass of non-safe P/T-nets for which unfoldings can be defined smoothly (namely those where the initial markings and the post-places of any transition are sets rather than proper multisets) can be seen to give rise to *causally deterministic event structures* if one takes the event structure corresponding to the occurrence net derived in [2, 3] and labels its events with the transitions of the original non-safe net. This notion of unfolding is known to be quite hard to extend to all Petri nets, however. Now, it is interesting to note that under the same notion of labelling, a naive unfolding of general Petri nets would yield *denotationally deterministic events structures*; see Fig. 5 for an example. Our strong feeling is that the problems encountered in unfolding general Petri nets are precisely the same as the ones involved in the categorical treatment of denotationally determinism. In particular, the absence of a notion of event structure determinism that gives rise to a category equivalent to (general) concurrent languages could very well be directly related to the difficulty in unfolding general Petri nets. If this feeling is justified, then the investigation of denotational determinism in *causal trees* proposed below might also shed light on Petri net unfoldings.

Future work. The results of this paper point out directions of further research. For one thing, the notion of *denotational determinism* might be captured more effectively by *causal trees* (see [1] than by event structures. Especially, the duplication of events during denotational determinisation may be avoided at least partly in the causal tree representation.

Part of this investigation is to generalise the results of [12], namely the categorical equivalence between deterministic semilanguages, a generalisation of Mazurkiewicz trace languages, and operationally deterministic event structures. A similar equivalence might exist between causally deterministic concurrent languages, a subclass of *causal trace languages*, and causally deterministic event structures.

As a possible further consequence of this line of research, we intend to investigate if the framework of models proposed by Sassone, Nielsen and Winskel might not be improved if one replaces event structures with causal trees. In particular, it might be possible to get rid of the need to forget auto-concurrent events when moving from nondeterministic to concurrency model.

Acknowledgement. Thanks to Roberto Gorrieri and Frits Vaandrager for clarifying some of the issues of operationally deterministic event structures.

References

1. P. Darondeau and P. Degano. Causal trees = interleaving + causality. In Guessarian, ed., *Semantics of Systems of Concurrent Processes*, vol. 469 of *LNCS*, pp. 239–255. Springer, 1990.
2. J. Engelfriet. Branching processes of Petri nets. *Acta Inf.*, 28:575–591, 1991.
3. J. Meseguer, U. Montanari, and V. Sassone. On the semantics of Petri nets. In W. R. Cleaveland, ed., *Concur '92*, vol. 630 of *LNCS*, pp. 286–301. Springer, 1992.
4. M. Nielsen, G. D. Plotkin, and G. Winskel. Petri nets, event structures and domains, part 1. *Theoretical Comput. Sci.*, 13:85–108, 1981.
5. M. Nielsen, V. Sassone, and G. Winskel. Relationships between models for concurrency. In J. W. de Bakker, W.-P. de Roever, and G. Rozenberg, eds., *A Decade of Concurrency*, vol. 803 of *LNCS*, pp. 425–476. Springer, 1994.
6. A. Rensink. Deterministic pomsets. Hildesheimer Informatik-Berichte 94/30, Universität Hildesheim, Nov. 1994.
7. A. Rensink. A complete theory of deterministic event structures. In I. Lee and S. A. Smolka, eds., *Concur '95: Concurrency Theory*, vol. 962 of *LNCS*, pp. 160–174. Springer, 1995.
8. A. Rensink. Denotational, causal, and operational determinism in event structures. Hildesheimer Informatik-Berichte 95/29, Universität Hildesheim, Sept. 1995.
9. A. Rensink. Further deterministic behavioural models for concurrency. Hildesheimer Informatik-Berichte 95/30, Universität Hildesheim, Oct. 1995.
10. V. Sassone. *On the Semantics of Petri Nets: Processes, Unfoldings and Infinite Computations*. PhD thesis, Universitá de Pisa, Mar. 1994. Available as TD–6/94.
11. V. Sassone, M. Nielsen, and G. Winskel. A classification of models for concurrency. In E. Best, ed., *Concur '93*, vol. 715 of *LNCS*, pp. 82–96. Springer, 1993.
12. V. Sassone, M. Nielsen, and G. Winskel. Deterministic behavioural models for concurrency. In A. M. Borzyszkowski and S. Sokołowski, eds., *Mathematical Foundations of Computer Science 1993*, vol. 711 of *LNCS*, pp. 682–692. Springer, 1993.
13. F. W. Vaandrager. Determinism $\longrightarrow$ (event structure isomorphism = step sequence equivalence). *Theoretical Comput. Sci.*, 79:275–294, 1991.
14. G. Winskel. Event structures. In W. Brauer, W. Reisig, and G. Rozenberg, eds., *Petri Nets: Applications and Relationships to Other Models of Concurrency*, vol. 255 of *LNCS*, pp. 325–392. Springer, 1987.

This article was processed using the LaTeX macro package with LLNCS style