Relevance Logic and Concurrent Composition

Mads Dam
Department of Computer Science, University of Edinburgh

ABSTRACT

We show that the operation of relativising properties with respect to parallel environments often employed in obtaining compositionality in theories for concurrency corresponds to a notion of (contraction—free) relevant deduction. We propose to consider program logics in which this notion of deduction is internalized by means of the corresponding implication. The idea is carried through for safety properties of a simple system of SCCS-type synchronuous processes with an internal choice operator. We present two completeness results; first for a modal extension of positive propositional linear logic w.r.t. the equational class of algebras containing the safety testing quotient of our process system as its free member, and secondly for the free algebra itself.

A central problem in successfully applying modal and temporal logics to the development and verification of concurrent programs is somehow to obtain compositionality. We must be able to compose and decompose properties in accordance with model structure [5,22]. In the context of parallel composition this is often done by some form of "environment relativization" [cf. 1,5,12,14,19,20] by appealing to potential computations rather than actual ones, and by indexing w.r.t. properties of parallel environments.

Logically, context relativization amounts to the dependency of properties on assumptions about the environment, a dependency admitting basic rules of deduction such as reflexivity and cut. In fact the proper notion of deduction is a *relevant* one [3,7]; once an assumption is introduced, it must be used in deriving the conclusion. We propose to consider program logics in which this notion of deduction is *internalized*, by adding to the logic an operation of (relevant) implication. We illustrate this idea by offering a clean logical account of a simple system of SCCS-type synchronous processes [16], following the suggestion of [11] to let logical properties govern the choice of process combinators. When taken with the safety testing notion of equivalence of [6], the system coincides with the free algebra in an equational class of algebras complete for a modal extension of positive propositional linear logic [8] (our choice of modalities bear no relation to those in [8]). To obtain completeness for the free algebra, however, we have to add a number of extra axioms as well as introduce a kind of intuitionistic negation. In addition we show the safety testing preorder to be characterized by the logic, in the sense that this and the ordering induced by the logic, when interpreting formulas directly over processes, coincide.

ENVIRONMENT RELATIVIZATION AS LINEAR CONSEQUENCE

Consider the problem of "structured model checking" where we have a satisfaction relation, $p \models \Phi$, between processes p and formulas Φ, and suppose there is an operation $\cdot$ of parallel composition of processes. It is hard to concieve of a operation $\odot$ which is tractable and licences interesting deductions of the form

if $p \models \Phi$ and $q \models \Psi$ then $p \cdot q \models \Phi \odot \Psi$.

Consequently properties of concurrent programs are usually proved either by analyzing out $\cdot$ in terms of more primitive process combinators, resulting in "state explosion" problems, or by "chasing derivations" using the satisfaction conditions for formulas. In either case compositionality is lost.

A proposal towards solving this problem is to introduce, as in [20,22], a doubly indexed turnstile, $\models'$, defined by

Φ,Ψ $\models' \Gamma$ iff whenever $p \models \Phi$, $q \models \Psi$ then
$p \cdot q \models \Gamma$.

Suppose we generalize this to allow arbitrary finite, nonempty sequences on the left hand side of $\models'$ (using $\bar{p}$, $\bar{\Phi}$, etc. to denote sequences, extending $\models$ pointwise):

$\bar{\Phi} \models' \Gamma$ iff for all $\bar{p}$, if $\bar{p} \models \bar{\Phi}$ then $\prod(\bar{p}) \models \Gamma$,

where $\prod(p_1, \ldots, p_n)$ denotes, say, $(p_1 \cdot (\ldots (p_{n-1} \cdot (p_n)) \ldots))$, for $n \geq 1$.

Then $\models'$ will satisfy

i. Reflexivity: $\Phi \models' \Phi$,
ii. Permutation: $\bar{\Phi} \models' \Gamma$ only if $\bar{\Psi} \models' \Gamma$, $\bar{\Psi}$ a permutation of $\bar{\Phi}$,
iii. Cut: $\bar{\Phi} \models' \Psi$ and $\bar{\Theta}, \Psi, \bar{\Delta} \models' \Gamma$ only if $\bar{\Phi}, \bar{\Theta}, \bar{\Delta} \models' \Gamma$,

as in fact $\models'$ will for any commutative, associative operation $\cdot$.

So it makes sense to think of $\models'$ as a consequence relation. In fact $\models'$ will be a linear consequence relation [8,2] in the sense that it will fail

iv. Contraction: $\bar{\Phi}, \Psi, \Psi \models' \Gamma$ only if $\bar{\Phi}, \Psi \models' \Gamma$, and
v. Weakening: $\bar{\Phi} \models' \Gamma$ only if $\bar{\Phi}, \Psi \models' \Gamma$.

Let us be slightly more bold and assume an identity, 1, for $\cdot$ (for instance 1 will be NIL in CCS [15], and in TCSP [4], when $\cdot$ is $\|(|||)$, 1 will be RUN(STOP)). Then it is natural to stipulate that $\prod(\varepsilon) = 1$, for ε the empty sequence. This means that $\varepsilon \models' \Gamma$ iff $1 \models \Gamma$.

Now the proposal is to consider logics in which this $\models'$ is internalized, in the sense that there is an operation, $\rightarrow$, of linear implication, s.t. $\bar{\Phi}, \Psi \models' \Gamma$ iff $\bar{\Phi} \models' \Psi \rightarrow \Gamma$. One can show that $p \models \Psi \rightarrow \Gamma$ iff for all q, if $q \models \Psi$ then $p \cdot q \models \Gamma$. So the property $\Psi \rightarrow \Gamma$ tells us something about the behaviour of elements as $\cdot$-contexts — one could read $\Psi \rightarrow \Gamma$ e.g. as "in every Ψ-context, Γ".

The logic we have thus defined is in fact the implicative fragment of linear logic [8], and the model and notion of satisfaction is but a slight variation of the semilattice models of relevance logics of [21]—the semilattice operation being idempotent, which $\cdot$ is not in general.

In the remainder of the paper we pursue the idea in detail, by building a logic for a concrete system of simple processes.

SYNCHRONOUS PROCESSES WITH INTERNAL CHOICE

Consider the following simple language of processes $p \in \mathcal{P}$, where $\alpha \in$ Act, a set of actions:

$$p ::= 0 \mid 1 \mid \alpha(p) \mid p \oplus q \mid p \cdot q,$$

where $\oplus$ is internal choice; and $\cdot$ synchronous parallel. We assume Act to be structured—forming an abelian group with unit e, as in [16], such that the simultaneous occurrence of actions can be accounted for.

In order to capture internal nondeterminism in a synchronous setting, we split up the notion of reduction into those of stabilizing and performing actions, as in [18]. Stable terms, $\sigma \in \Sigma \subseteq \mathcal{P}$, are generated by the subprocess language

$$\sigma ::= 0 \mid 1 \mid \alpha(p) \mid \sigma \cdot \sigma,$$

and the stabilization relation, $\rightarrow$, is the least s.t.

i. $0 \rightarrow 0$,
ii. $1 \rightarrow 1$,
iii. $\alpha(p) \rightarrow \alpha(p)$,
iv. $p \rightarrow \sigma$ only if $p \oplus q \rightarrow \sigma$ and $q \oplus p \rightarrow \sigma$, and
v. $p \rightarrow \sigma$, $q \rightarrow \tau$ only if $p \cdot q \rightarrow \sigma \cdot \tau$.

For a $p \in \mathcal{P}$ let $st(p) = \{\sigma \mid p \rightarrow \sigma\}$. The transition relation between stable terms and processes is standard:

i. $1 \xrightarrow{e} 1$,
ii. $\alpha(p) \xrightarrow{\alpha} p$, and
iii. $\sigma \xrightarrow{\alpha} p$, $\tau \xrightarrow{\beta} q$ only if $\sigma \cdot \tau \xrightarrow{\alpha\beta} p \cdot q$.

Define the family of predicates **can** α, for $\alpha \in$ Act, by σ **can** α iff there is a p s.t. $\sigma \xrightarrow{\alpha} p$. Clearly, if σ **can** α, then α is unique, and so is the p s.t. $\sigma \xrightarrow{\alpha} p$. Let σ/α denote this p, whenever it is defined. Say p **can** α iff for some $\sigma \in st(p)$ σ **can** α, p **live** iff for all $\sigma \in st(p)$ there is an α s.t. σ **can** α, and $p/\alpha = \{p' \mid \exists \sigma \in st(p).\sigma \xrightarrow{\alpha} p'\}$.

Processes are identified according to the set of potential outcomes, when running them with a test [6]; an outcome being either failure or success, depending on whether or not the test is brought to termination. This notion of equivalence (in fact its safety part, see below) is closely related to the failure set model introduced in [4]. There are finer (i.e. more discriminating) notions of equivalence, notably the observational equivalence of [15,10]. These, however, will not in general admit the distribution laws desirable in the present context.

Tests $t \in T$ are finitely branching trees: $0 \in T$, and for I finite and nonempty and $t_i \in T$ for all $i \in I$ then $\sum_{i \in I} \alpha_i t_i \in T$. Sets of outcomes are distinguished in two different ways, according to whether or not they are 1. sometimes successful and 2. never unsuccessful. This idea is formalized by the "may" and "must" notions of test acceptance, defined by

i. p **may** 0 for all $p \in \mathcal{P}$, and
p **may** $\sum_{i \in I} \alpha_i t_i$ iff $p \rightarrow \sigma$ for some $\sigma \in \Sigma$ s.t.
σ **can** α_i and σ/α_i **may** t_i for some $i \in I$.
ii. p **must** 0 for all $p \in \mathcal{P}$, and for $I \neq \emptyset$,
p **must** $\sum_{i \in I} \alpha_i t_i$ iff for all $\sigma \in \Sigma$, if $p \rightarrow \sigma$ then
σ **can** α_i for some i, and for all $i \in I$, if
σ **can** α_i then σ/α_i **must** t_i.

We then define the testing approximation relations, $\preceq_i$, for $i \in \{1, 2, 3\}$, by

i. $p \preceq_1 q$ iff for all $t \in T$, p **may** t only if q **may** t,
ii. $p \preceq_2 q$ iff for all $t \in T$, p **must** t only if q **must** t,
iii. $\preceq_3 = \preceq_1 \cap \preceq_2$.

Let $\simeq_i = \preceq_i \cap \succeq_i$, for $i \in \{1, 2, 3\}$. We refer to $\preceq_1$ as the *liveness* and $\preceq_2$ as the *safety* preorder. These preorders can be characterized recursively much like the weak equivalence of [13]. Let P,Q,... (A) range over finite, nonempty subsets of $\mathcal{P}$ (Act). Extend $\preceq_1$,$\preceq_2$ to sets P,Q by defining P **may** t (P **must** t) iff for some $p \in P$ (for all $p \in P$) p **may** t (p **must** t). Say P **must** A iff P **must** $\sum_{\alpha \in A} \alpha 0$, and extend the notions **can**, **live** and $\cdot/\alpha$ to sets P, Q in the obvious way. Now define, like for observational equivalence [15], the decreasing chains $\{\sqsubseteq_{i,n}\}_{n \geq 0, i \in \{1,2\}}$ of relations by

i. $P \sqsubseteq_{1,0} Q, P \sqsubseteq_{2,0} Q$ for all P, Q.
ii. $P \sqsubseteq_{1,n+1} Q$ iff for all α, if P **can** α then
Q **can** α and $P/\alpha \sqsubseteq_{1,n} Q/\alpha$.
iii. $P \sqsubseteq_{2,n+1} Q$ iff for all A, α,
P **must** A only if Q **must** A, and
P **live**, Q **can** α only if P **can** α and
$P/\alpha \sqsubseteq_{2,n} Q/\alpha$.

Let $\sqsubseteq_1 = \bigcap_{n \geq 0} \sqsubseteq_{1,n}$ and $\sqsubseteq_2 = \bigcap_{n \geq 0} \sqsubseteq_{2,n}$. We extend operations to sets P, Q by pointwise extensions, e.g. $P \cdot Q = \{p \cdot q \mid p \in P, q \in Q\}$.

Theorem 1.
i. *For all* $i \in \{1, 2\}$, $\preceq_i = \sqsubseteq_i$.
ii. *For all* $i \in \{1, 2, 3\}$, $\simeq_i$ *is substitutive.*

Proof: i. For $\subseteq$ show for $i = 1, 2$ respectively that $\preceq_i \subseteq \sqsubseteq_{i,n}$ for all $n \geq 0$. For the converse assume e.g. for $i = 2$ that P **must** t but not Q **must** t, and proceed to show that then $P \not\sqsubseteq_{2,n} Q$ for some n.
ii. It suffices to show $\sqsubseteq_{i,n}$ substitutive for all $n \geq 0$ and $i \in \{1, 2\}$. We take only the case for $i = 2$ and $\cdot$. So assume $P_1 \sqsubseteq_{2,n+1} P_2$. If $P_1 \cdot Q$ **must** A then P_1 **must** $\{\alpha_1 \mid P_1$ **can** $\alpha_1, \exists \alpha_Q.Q$ **can** $\alpha_Q, \alpha_1\alpha_Q \in A\} = A_1$. Then P_2 **must** A_1, whence $P_2 \cdot Q$ **must** A. Suppose that $P_1 \cdot Q$ **live** and $P_2 \cdot Q$ **can** α. Then P_1, Q **live** and P_2 **can** α_1, Q **can** α_2 for some α_1, α_2 s.t. $\alpha_1\alpha_2 = \alpha$. Then P_1 **can** α_1 whence $P_1 \cdot Q$ **can** α. Also whenever P_2 **can** α_1, Q **can** α_2 and $\alpha_1\alpha_2 = \alpha$ then $P_1/\alpha_1 \cdot Q/\alpha_2 \sqsubseteq_{2,n} P_2/\alpha_1 \cdot Q/\alpha_2$ by the induction hypothesis. But then, as for all n,P,P',Q if $P \sqsubseteq_{2,n} Q$ and $P \subseteq P'$ then $P' \sqsubseteq_{2,n} Q$, we obtain $(P_1 \cdot Q)/\alpha \sqsubseteq_{2,n} (P_2 \cdot Q)/\alpha$, whence $P_1 \cdot Q \sqsubseteq_{2,n+1} P_2 \cdot Q$ and we are done.

SYNCHRONOUS ALGEBRAS

The three preorders, $\preceq_i$, $i \in \{1, 2, 3\}$, have simple (in-)equational axiomatizations, in terms of "synchronous algebras".

These are structures $\mathcal{A} = \langle A, \leq, 0, \oplus, \cdot, 1, \text{Act}\rangle$, where
i. $\langle A, \leq, 0\rangle$ is a poset with 0 least and $\leq$ substitutive w.r.t. $\oplus$ and $\cdot$,
ii. $\langle A, \oplus\rangle$ is a semilattice,
iii. $\langle A, \cdot, 1\rangle$ is a commutative monoid with $\cdot$ preserving $\oplus$ and 0, and
iv. Act is an abelian group as above of operators on $\mathcal{A}$ s.t. each $\alpha \in$ Act preserves $\leq$ and $\oplus$, and s.t. the following equations hold:
a. $\alpha(x) \oplus 0 = \alpha(x \oplus 0) \oplus 0$
b. $\alpha(x) \cdot \beta(y) = (\alpha\beta)(x \cdot y)$
c. $e(1) = 1$

If $\oplus$ is the inf(sup) w.r.t. $\leq$ we say $\mathcal{A}$ is a *safety* (*liveness*) algebra and denote the $\oplus$ by $\sqcap(\sqcup)$. Clearly the safety (liveness) algebras form an equational class—define $x \leq y$ ($y \leq x$) iff $x \oplus y = x$. Note that if $\mathcal{A}$ is a safety or liveness algebra, a. is redundant. Let $\mathcal{C}_3$ denote the class of all synchronous algebras, and $\mathcal{C}_2(\mathcal{C}_1)$ the class of safety(liveness) algebras.

All three classes admit free algebras—let $\mathcal{F}_1(\mathcal{F}_2)$ denote the free liveness(safety) algebra, and $\mathcal{F}_3$ the free synchronous algebra. These algebras have simple representations in terms of sets of irredundant paths.

A *path*, s, is a pair $\langle \bar{\alpha}, i\rangle$, with $\bar{\alpha} \in$Act* and $i \in \{0, 1\}$; and $s = \langle \bar{\alpha}, i\rangle$ is *irredundant*, if e is a suffix of $\bar{\alpha}$ only when $i = 0$. We assume all paths below irredundant, unless otherwise specified. We order paths by $\langle \bar{\alpha}, i\rangle \leq \langle \bar{\beta}, j\rangle$ iff either $i = 1 = j$ and $\bar{\alpha} = \bar{\beta}$, or $i = 0$ and $\bar{\alpha}$ is a prefix of $\bar{\beta}$. Sets, S_1, S_2, of paths are ordered by $S_1 \leq_1 S_2$ iff $\forall s_1 \in S_1 \exists s_2 \in S_2$ s.t. $s_1 \leq s_2$, $S_1 \leq_2 S_2$ iff $\forall s_2 \in S_2 \exists s_1 \in S_1$ s.t $s_1 \leq s_2$, and $\leq_3 = \leq_1 \cap \leq_2$. A set, S, of paths is *lower*(1), if $s \in S$ and $s' \leq s$ implies $s' \in S$, it is *upper*(2), if $s \in S$ and $s \leq s'$ implies $s' \in S$, and it is *convex*(3), if $s_1, s_2 \in S$ and $s_1 \leq s \leq s_2$ implies $s \in S$. Let cl_i, $i \in \{1, 2, 3\}$ denote the corresponding closure operators. A closed set, S, of paths is *finitely generated* (f.g.), if S is the closure of a finite set—clearly a 1- or 3-closed f.g. set is finite. Note that generating sets are closed under intersections—hence every f.g. set contains a unique least generating set.

Prefixing of paths is defined by $\alpha(\langle \bar{\alpha}, i\rangle) = \langle (\alpha\bar{\alpha}), i\rangle$, if $\alpha \neq e$ or $\langle \bar{\alpha}, i\rangle \neq \langle \varepsilon, 1\rangle$, and $e(\langle \varepsilon, 1\rangle) = \langle \varepsilon, 1\rangle$. Multiplication of paths is defined similarly. Then the operators, for $i \in \{1, 2, 3\}$, are defined by $0_i = cl_i\{\langle \varepsilon, 0\rangle\}$, $1_i = cl_i\{\langle \varepsilon, 1\rangle\}$, $\alpha_i(S) = cl_i\{\alpha(s) \mid s \in S\}$, $S_1 \oplus_i S_2 = cl_i(S_1 \cup S_2)$, and $S_1 \cdot_i S_2 = cl_i\{s_1 \cdot s_2 \mid s_1 \in S_1, s_2 \in S_2\}$.

Let then $D_i = \{S \mid S$ is an i-closed, f.g. and nonempty set of irredundant paths$\}$, and $\mathcal{D}_i$ denote the D_i ordered by $\leq_i$ with operations as defined above, for $i \in \{1, 2, 3\}$. We obtain

Theorem 2. *For all* $i \in \{1, 2, 3\}$, $\mathcal{F}_i \cong \mathcal{D}_i$.
Proof: See appendix.

Then processes modulo the three preorders are characterized by

Theorem 3. *For all* $i \in \{1, 2, 3\}$,
$\langle \mathcal{P}/\simeq_i, \preceq_i / \simeq_i\rangle \cong \mathcal{F}_i$.
Proof: See appendix.

Thm. 2 and 3. automatically gives us a fully abstract semantics, $[\![\cdot]\!]_i$, for $i \in \{1, 2, 3\}$, from $\mathcal{P}$ onto $\mathcal{D}_i$.

From this point onwards we shall deal solely with the safety case — hence $\mathcal{D}$, $[\![\cdot]\!]$, $\preceq$, etc. shall denote the corresponding safety entities.

A RELEVANT LOGIC OF PROCESSES

We treat safety properties as filters rather than ideals, in contrast to [11]: for safety property Φ, if $p \models \Phi$ and $p \preceq q$ then $q \models \Phi$; and if $p \models \Phi$ and $q \models \Phi$ then $p \sqcap q \models \Phi$ (where $\sqcap$ is $\oplus$ in the safety case).

As formulas $\Phi \in$Form we take

$$\Phi ::= X \mid \Phi \rightarrow \Phi \mid \Phi \wedge \Phi \mid \Phi \vee \Phi \mid (\alpha)\Phi \mid \overline{(\alpha)}\Phi \\ \mid t \mid \bot,$$

where $X \in$Var, some set of propositional variables, $\rightarrow$ is relevant implication, $\wedge$/$\vee$ are extensional ("additive", in the terminology of [8]) and/or, (α) a future, $\overline{(\alpha)}$ a past, or reverse modality, t is intensional ("multiplicative") truth and $\bot$ is extensional falsehood. The operations $\top$, $\neg$, $\leftrightarrow$ and (α) are defined by $\top \overset{\text{def}}{=} \bot \rightarrow \bot$, $\neg\Phi \overset{\text{def}}{=} \Phi \rightarrow \bot$, $\Phi \leftrightarrow \Psi \overset{\text{def}}{=} (\Phi \rightarrow \Psi) \wedge (\Psi \rightarrow \Phi)$, and $(\alpha) \overset{\text{def}}{=} (\alpha)\top$.

Initially we interpret formulas over safety algebras. A model is a pair $\mathcal{M} = \langle \mathcal{A}, V \rangle$, where $\mathcal{A}$ is a safety algebra and V a valuation of propositional variables s.t. for each $X \in$Var, $V(X)$ is a filter in $\mathcal{A}$.

Then satisfaction $\models_{\mathcal{M}}$ is defined inductively by (we suppress subscripting of $\mathcal{M}$):

$x \models X$ iff $x \in V(X)$,
$x \models \Phi \rightarrow \Psi$ iff for all $y \in A$, if $y \models \Phi$ then $x \cdot y \models \Psi$,
$x \models \Phi \wedge \Psi$ iff $x \models \Phi$ and $x \models \Psi$,
$x \models \Phi \vee \Psi$ iff $x \models \Phi$ or $x \models \Psi$ or there are $x_1, x_2 \in A$, s.t. $x_1 \sqcap x_2 \leq x$, $x_1 \models \Phi$ and $x_2 \models \Psi$,
$x \models (\alpha)\Phi$ iff there is an $x' \in A$ s.t. $\alpha(x') \leq x$, $x' \models \Phi$,
$x \models \overline{(\alpha)}\Phi$ iff $\alpha(x) \models \Phi$,
$x \models t$ iff $1 \leq x$, and
$x \not\models \bot$.

So the $\rightarrow$ is the operation of relativizing w.r.t. $\cdot$-contexts introduced above. For the $\vee$ note that the standard satisfaction condition ($x \models \Phi \vee \Psi$ iff $x \models \Phi$ or $x \models \Psi$) will not work with our interpretation of properties as filters — here, an x will have the property $\Phi \vee \Psi$ iff it dominates an "internal branching" (i.e. inf) of two elements either of which satisfies either Φ or Ψ.

Validity is defined by: Φ valid in $\mathcal{M}$, $\models_{\mathcal{M}} \Phi$ iff $1_{\mathcal{M}} \models_{\mathcal{M}} \Phi$ (cf. sec. 2). We obtain

Theorem 4. *For all* $\Phi \in$Form, $[\![\Phi]\!] = \{x \mid x \models \Phi\}$ *is a filter.*

Proof: Induction on the structure of Φ. We take only the case for $\Phi = \Phi_1 \rightarrow \Phi_2$ — the other cases are similar. Let $x \models \Phi_1 \rightarrow \Phi_2$ and $x \leq y$. Let $z \models \Phi_1$. Then $x \cdot z \models \Phi_2$ and $x \cdot z \leq y \cdot z$ by the monotonicity of $\cdot$, whence by the induction hypothesis $y \cdot z \models \Phi_2$. But then $y \models \Phi_1 \rightarrow \Phi_2$. Next, if $x, y \models \Phi_1 \rightarrow \Phi_2$ and $z \models \Phi_1$ then $x \cdot z, y \cdot z \models \Phi_2$. Now $x \cdot z \sqcap y \cdot z = (x \sqcap y) \cdot z$, and by the induction hypothesis $x \cdot z \sqcap y \cdot z \models \Phi_2$, and we have shown $x \sqcap y \models \Phi_1 \rightarrow \Phi_2$. Thus $[\![\Phi_1 \rightarrow \Phi_2]\!]$ is a filter.

For the free safety algebra, the defining clauses for $\models$ may be strengthened by replacing $\leq$ by $=$ throughout. This makes our clause for $\vee$ vaguely similar to the corresponding one in [11], as well as the one for $+$ in [9]. Notice that in $\mathcal{D}$ each element is the inf of the set of primes (= one-element sets) above it (an S is prime if whenever $S_1 \sqcap S_2 \leq S$ then $S_1 \leq S$ or $S_2 \leq S$). In fact we obtain

Theorem 5: *For all* $S \in \mathcal{D}$, $\Phi, \Psi \in \mathrm{Form}$,

i. $S \models \Phi$ *iff for all prime* $S' \in \mathcal{D}$, $S \leq S'$ *only if* $S' \models \Phi$,
ii. $S \models \Phi \vee \Psi$ *iff for all prime* $S' \in \mathcal{D}$, $S \leq S'$ *only if* $S' \models \Phi$ *or* $S' \models \Psi$.

Proof: Straightforward.

The property 5.ii is reminiscent of the barring in Beth models for intuitionistic logic. We can use this property to give an account of satisfaction directly on the processes themselves. Notice that an $S \in \mathcal{D}$ is prime iff it is the image of some trace, where a trace is a process not containing occurrences of $\oplus$. Define satisfaction, $\dot{\models}$, of variable-free formulas by

$p \dot{\models} \Phi \vee \Psi$ iff for all traces σ of p, $\sigma \dot{\models} \Phi$ or $\sigma \dot{\models} \Psi$,
$p \dot{\models} (\alpha)\Phi$ iff for all traces σ of p, σ **can** α, $\sigma/\alpha \dot{\models} \Phi$,
$p \dot{\models} t$ iff for all traces σ of p, σ **can** e, $\sigma/e \dot{\models} t$,

and the other clauses being identical to those of $\models$. Then one can show that our logic characterizes the safety ordering in the sense that

Theorem 6. *For all* $p \in \mathcal{P}$,

i. *For all variable-free* $\Phi \in$Form, $p \dot{\models} \Phi$ *iff* $[\![p]\!] \models \Phi$,
ii. $p \preceq q$ *iff for all variable-free* $\Phi \in$Form, $p \dot{\models} \Phi$ *implies* $q \dot{\models} \Phi$.

Proof (sketch)**:** i. Induction in the structure of Φ. Use thm. 3, 4 and 5. ii. The "only if" direction follows from i. and thm. 4. For the converse direction, note that an $S \in \mathcal{D}$ has a simple representation, $R(S)$, in the logic, depending on the least generating subset of S:

$$\begin{aligned} & R(cl(\{\langle \varepsilon, 0 \rangle\})) = \top, \\ & R(cl(\{\langle \varepsilon, 1 \rangle\})) = t, \\ & R(cl(\{\langle (\alpha\bar{\alpha}), i \rangle\})) = (\alpha)R(cl(\{\langle \bar{\alpha}, i \rangle\})), \\ & R(cl(\{s_1, \cdots, s_n\})) \\ & \qquad = R(cl(\{s_1\})) \vee \cdots \vee R(cl(\{s_n\})), \text{ for } n > 1. \end{aligned}$$

Then one shows by induction on the complexity of the least generating set of S that for all $S' \in D$, $S' \models R(S)$ iff $S \leq S'$. Now the result follows, for if $q \dot{\models} R([\![p]\!])$, then $[\![q]\!] \models R([\![p]\!])$ by i., whence $[\![p]\!] \leq [\![q]\!]$, so $p \preceq q$ by thm. 3.

A COMPLETENESS RESULT FOR SAFETY ALGEBRAS

In this section we exhibit a Hilbert-type axiomatization of the $\perp$-free fragment of our logic w.r.t. validity in all models.

Axioms: 1. $\vdash \Phi \to \Phi$

2. $\vdash (\Phi \to \Psi) \to ((\Gamma \to \Phi) \to (\Gamma \to \Psi))$

3. $\vdash (\Phi \to (\Psi \to \Gamma)) \to (\Psi \to (\Phi \to \Gamma))$

4. $\vdash (\Phi \to \Psi) \wedge (\Phi \to \Gamma) \to (\Phi \to \Psi \wedge \Gamma)$

5. $\vdash \Phi \wedge \Psi \to \Phi$

6. $\vdash \Phi \wedge \Psi \to \Psi$

7. $\vdash (\Phi \to \Gamma) \wedge (\Psi \to \Gamma) \to (\Phi \vee \Psi \to \Gamma)$

8. $\vdash \Phi \to \Phi \vee \Psi$

9. $\vdash \Psi \to \Phi \vee \Psi$

10. $\vdash \Phi \leftrightarrow (t \to \Phi)$

11. $\vdash (\alpha)(\Phi \vee \Psi) \to (\alpha)\Phi \vee (\alpha)\Psi$

12. $\vdash \overline{(\alpha)}\Phi \wedge \overline{(\alpha)}\Psi \to \overline{(\alpha)}(\Phi \wedge \Psi)$

13. $\vdash \Phi \to \overline{(\alpha)}((\alpha)\Phi)$

14. $\vdash (\alpha)(\overline{(\alpha)}\Phi) \to \Phi$

15. $\vdash \overline{(\alpha)}((\beta)\Phi \to \Psi) \leftrightarrow (\Phi \to \overline{(\alpha\beta)}\Psi)$

Rules:		
	m.p.	$\vdash \Phi \to \Psi, \vdash \Phi$ only if $\vdash \Psi$
	adj.	$\vdash \Phi, \vdash \Psi$ only if $\vdash \Phi \wedge \Psi$
	(e)-nec.	$\vdash \Phi$ only if $\vdash (e)\Phi$
	$\overline{(e)}$-nec.	$\vdash \Phi$ only if $\vdash \overline{(e)}\Phi$
	(α)-mon.	$\vdash \Phi \to \Psi$ only if $\vdash (\alpha)\Phi \to (\alpha)\Psi$
	$\overline{(\alpha)}$-mon.	$\vdash \Phi \to \Psi$ only if $\vdash \overline{(\alpha)}\Phi \to \overline{(\alpha)}\Psi$.

Notice that our axiomatization of the modal-free fragment (axioms 1–10, rules m.p., adj.) is just a standard axiomatization of the ($\to$, $\wedge$, $\vee$, t)-fragment of linear logic. Incidentally the semilattice ordered monoids underlying our notion of safety algebra are closely related to the models for BCK- and related logics of [17]. Note also that we shall not obtain completeness for the free algebra; e.g. in $\mathcal{F}$, $(\alpha)\Phi \wedge (\beta)\Psi$ is unsatisfiable whenever $\alpha \neq \beta$, whereas this is not true in general. We obtain

Theorem 7. $\vdash \Phi$ *iff for all safety models* $\mathcal{M}$, $\models_{\mathcal{M}} \Phi$.
Proof (sketch)**:** Soundness is proved in the standard way (note that $\models \Phi \to \Psi$ iff for all x, $x \models \Phi$ implies $x \models \Psi$). Completeness is obtained by a simple Henkin-type construction. A *theory* ∇ is any (!) set of formulas s.t. $\Phi \in \nabla$ and $\vdash \Phi \to \Psi$ only if $\Psi \in \nabla$, and Φ, $\Psi \in \nabla$ only if $\Phi \wedge \Psi \in \nabla$. For any set, S, of formulas, there is a least theory $th(S)$ containing S. We build a safety algebra from theories, by taking $\sqcap$ to be $\cap$, 0 to $\emptyset$, and

$$\nabla_1 \cdot \nabla_2 = \{\Psi \mid \exists\Phi \in \nabla_2 . \Phi \to \Psi \in \nabla_1\}$$

$$= \{\Psi \mid \exists\Phi \in \nabla_1 . \Phi \to \Psi \in \nabla_2\},$$

$$1 = \{\Phi \mid\vdash \Phi\},$$

$$\alpha(\nabla) = \{\Phi \mid \overline{(\alpha)}\Phi \in \nabla\} = th\{(\alpha)\Phi \mid \Phi \in \nabla\},$$

and define the valuation, V, by $\nabla \in V(X)$ iff $X \in \nabla$. The check that the operations are well-defined and that we indeed do obtain a safety algebra in this way is a straightforward application of the axioms and rules. As an example we show $\cdot$ associative. So assume that $\Phi \in \nabla_1 \cdot (\nabla_2 \cdot \nabla_3)$. Then there is a $\Psi \in \nabla_1$ s.t. $\Psi \to \Phi \in \nabla_2 \cdot \nabla_3$, and then there is a $\Gamma \in \nabla_2$ s.t. $\Gamma \to (\Psi \to \Phi) \in \nabla_3$. Using 1–3 and m.p. it is easy to show, that

$$\vdash \Psi \to (\Gamma \to ((\Gamma \to (\Psi \to \Phi)) \to \Phi))$$

and then $(\Gamma \to (\Psi \to \Phi)) \to \Phi \in \nabla_1 \cdot \nabla_2$, whence $\Phi \in (\nabla_1 \cdot \nabla_2) \cdot \nabla_3$ as desired. It is straightforward also to check that $\nabla \models \Phi$ iff $\Phi \in \nabla$, and we are done, for if $\not\vdash \Phi$ then $\Phi \notin 1$, and then $1 \not\models \Phi$ in the canonical model.

COMPLETENESS FOR THE FREE SAFETY ALGEBRA

Finally we exhibit an axiomatization of the logic with falsehood, but without propositional variables, complete for $\mathcal{F}$.

First we have the standard axioms for $\top$ and $\perp$:

16. $\vdash \Phi \to \top$

17. $\vdash \perp \to \Phi$

We strengthen 13. and 14. to

13’. $\vdash \Phi \leftrightarrow \overline{(\alpha)}((\alpha)\Phi)$

14’. $\vdash \Phi \wedge (\alpha) \leftrightarrow (\alpha)(\overline{(\alpha)}\Phi)$

Then the additions are

18. $\vdash t \leftrightarrow (e)t$

19. $\vdash \Phi \wedge (\Psi \vee \Gamma) \to (\Phi \wedge \Psi) \vee \Gamma$

20. $\vdash \neg\Phi \to (\Psi \to \neg\Phi)$

21. $\vdash \neg\overline{(\beta)}((\alpha)\Phi)$, for $\alpha \neq \beta$

22. $\vdash \neg((\alpha)\Phi) \leftrightarrow \neg\Phi$

23. $\vdash ((\alpha)\Phi \to \bigvee_{\beta \in B}(\beta)\Psi_\beta) \to$
$\bigvee_{\beta \in B}((\alpha)\Phi \to (\beta)\Psi_\beta)$

24. $\vdash \neg(\top \to \bigvee_{\alpha \in A}(\alpha)\Phi_\alpha)$

25. $\vdash \top \to \overline{(\alpha)}\top$

26. $\vdash \neg\neg\Phi \wedge ((\alpha)\Phi \to (\beta)\Psi) \to$
$(\beta(\alpha^{-1}))(\Phi \to \Psi)$

Note:

i. Axiom 18 makes rules (e)-nec., $\overline{(e)}$-nec. redundant.

ii. Axiom 19 (distributivity) marks a departure from linear logic.

iii. Axiom 20 reveals the strong nature of the negation ($x \models \neg\Phi$ iff for all y, $y \not\models \Phi$).

To summarize, our axiom system consists of axioms 1–12,13’,14’,15–26 plus rules m.p., adj., (α).-mon. and (α)-mon. Provability, $\vdash$, from now on denotes provability in this system.

Completeness is proved via a normal form theorem for formulas. Note first, that we have

Theorem 8. *Let* $\Phi \equiv \Psi$ *iff* $\vdash \Phi \leftrightarrow \Psi$. *Then* $\equiv$ *is a congruence.*

Proof: The equivalence property of $\equiv$ follows from axioms 1, 2, 5, 6 plus m.p. and adj. We show that $\equiv$ is respected by the operations. E.g. for $\rightarrow$, if $\Phi \equiv \Psi$ then $\Phi \rightarrow \Gamma \equiv \Psi \rightarrow \Gamma$, $\Gamma \rightarrow \Phi \equiv \Gamma \rightarrow \Psi$ by axioms 2, 3, 5, 6 plus m.p. and adj. The other cases are similar.

Theorem 9. *For all variable-free formulas,* $\Phi \in$Form, $\vdash \Phi$ *iff* $\models_{\mathcal{F}} \Phi$.

Proof (sketch)**:** Soundness is proved as usual, using the representation in thm. 2. For completeness, define the set NF' inductively by

i. $t, \top \in NF'$,

ii. $\bigvee_{\alpha \in A} (\alpha)\Phi_\alpha \in NF'$, if

   a. A is a finite, nonempty subset of Act,

   b. $e \in A$, $\Phi_e = t$ only if card$(A) > 1$, and

   c. for all $\alpha \in A$, $\Phi_\alpha \in NF'$,

and then we let $NF = NF' \cup \{\bot\}$. In order to apply axiom 26 note that:

Lemma 1. *If* $\Phi \in NF'$ *then* $\vdash \neg\neg\Phi$.

Proof: By induction on the structure of normal forms. First, we have $\vdash \Phi \rightarrow \neg\neg\Phi$. Hence, if $\Phi = \top$ or $\Phi = t$, by $\vdash \top$, $\vdash t$ we have $\vdash \neg\neg\top$, $\vdash \neg\neg t$. By axiom 22, $\vdash \neg((\alpha)\Phi) \rightarrow \neg\Phi$, hence if $\vdash \neg\neg\Phi$ also $\vdash \neg\neg((\alpha)\Phi)$. Additionally, if $\vdash \neg\neg\Phi$, $\vdash \neg\neg\Psi$ then $\vdash \neg\neg(\Phi \vee \Psi)$. This handles the inductive case and we are done.

Now one can show by a long structural induction and an inner induction on the modal depth of formulas (note only we assign t the modal depth 0), that each formula Φ can be rewritten into a $\Psi \in NF$, s.t. $\Phi \equiv \Psi$. We outline the proof for $\Phi = \Phi_1 \rightarrow \Phi_2$. By the induction hypothesis, Φ_1, Φ_2 can be rewritten into normal form — suppose that $\Phi_1 \equiv \bigvee_{\alpha \in A} (\alpha)\Phi_\alpha \in NF$ and $\Phi_2 \equiv \bigvee_{\beta \in B} (\beta)\Phi'_\beta \in NF$. Using 2,3,7,8,9 and 23 we obtain

$$\vdash \Phi \leftrightarrow \bigwedge_\alpha \bigvee_\beta ((\alpha)\Phi_\alpha \rightarrow (\beta)\Phi'_\beta).$$

For each $\alpha \in A$, $\Phi_\alpha \in NF'$, hence $\vdash \neg\neg\Phi_\alpha$ by lemma 1. Using 26 we can obtain

$$\vdash \Phi \leftrightarrow \bigwedge_\alpha \bigvee_\beta (\beta(\alpha^{-1}))(\Phi_\alpha \rightarrow \Phi'_\beta).$$

By the inner induction hypothesis we can obtain for each $\alpha \in A$, $\beta \in B$ a $\Psi_{\alpha,\beta} \equiv \Phi_\alpha \rightarrow \Phi'_\beta$ s.t. $\Psi_{\alpha,\beta} \in NF$. By thm. 8 we obtain $\vdash \Phi \leftrightarrow \bigwedge_\alpha \bigvee_\beta (\beta\alpha^{-1})\Psi_{\alpha,\beta}$. We can assume, that for all $\alpha \in A$ there is a $\beta \in B$ s.t. $\Psi_{\alpha,\beta} \neq\bot$, for otherwise $\vdash \Phi \leftrightarrow\bot$ and we are done. By 19 we obtain

$$\vdash \Phi \leftrightarrow \bigvee_{f:A\rightarrow B} \bigwedge_\alpha (f(\alpha)\alpha^{-1})\Psi_{\alpha,f(\alpha)}.$$

Using 14’,16,21,22 and (α)-mon one can show for all Γ_1, Γ_2, $\vdash \neg((\alpha)\Gamma_1 \wedge (\beta)\Gamma_2)$, whenever $\alpha \neq \beta$. Hence if for all $f : A \rightarrow B$ there are $\alpha_1, \alpha_2 \in A$ s.t. $f(\alpha_1)\alpha_1^{-1} \neq f(\alpha_2)\alpha_2^{-1}$, then $\vdash \Phi \leftrightarrow\bot$. Otherwise a little bit of manipulation establishes a finite, nonempty set $C \subseteq$ Act and for each $\gamma \in C$ a $\Gamma_\gamma \in NF'$ s.t.

$$\vdash \Phi \leftrightarrow \bigvee_{\gamma \in C} (\gamma)\Gamma_\gamma.$$

If for all $\gamma \in C$, $\Gamma_\gamma =\bot$ then $\vdash \Phi \leftrightarrow\bot$ by 22. Otherwise, using 22 again, we can assume for each $\gamma \in C$, that $\Gamma_\gamma \neq\bot$. If $C = \{e\}$ and $\Gamma_e = t$ then by 18, $\vdash \Phi \leftrightarrow t$. Otherwise $\bigvee_{\gamma \in C} (\gamma)\Gamma_\gamma \in NF'$ and we are done.

Now we are almost home, for it is easy to show that for $\Phi \in NF$, $\models \Phi$ iff Φ “contains an e-trace” (that is, $\Phi = t$, $\Phi = \top$ or $\Phi = \bigvee_{\alpha \in A} (\alpha)\Phi_\alpha$, $e \in A$ and Φ_e contains an e-trace). But for such Φ it is very easy to construct a proof — thus ending the proof of thm. 9.

Notice, that the rewriting procedure used in the proof of thm. 9 gives us a procedure for deciding in one go

i. Validity: Φ valid iff $\models \Phi$ iff $1 \models \Phi$,

ii. “Universal” validity: Φ “universally” valid iff for all $x \in \mathcal{F}$, $x \models \Phi$, iff $\Phi \equiv \top$, and

iii. “Satisfiability”: Φ “satisfiable” iff there is an $x \in \mathcal{F}$ s.t. $x \models \Phi$, iff $\Phi \not\equiv\bot$.

CONCLUDING REMARKS

Of course lots of questions remain to be answered. First on the axiomatization of section 6: Can one add falsehood, $\bot$, to the axiomatization? Properties such as $x \cdot 0 \leq 0$ seem hard to obtain, due to peculiarities of our negation. Also it does not seem possible to obtain a complete axiomatization with future modalities alone. However, one can choose to add the “fusion” (intensional conjunction) instead of the past modality.

Secondly, how do we deal with the liveness case? One idea could be to let the disjunction induce underspecification, or “Hoare-type” nondeterminism [4] into the process system, enabling a treatment similar to the safety case—maintaining our identification of properties as filters. But more generally it remains to be seen how these ideas can cope with various extensions of the process language, primarily external choice.

Acknowledgements. The present work was supported by the Danish Technical Research Council, grant no. 16-3809.E. Thanks to my Ph.D. supervisor, Colin Stirling for innumerable valuable discussions on the subject.

APPENDIX

Proof of theorem 2: First check, that for each $i \in \{1, 2, 3\}$, $\mathcal{D}_i \in \mathcal{C}_i$. Next each mapping, $f : \mathcal{D}_i \to \mathcal{A}$, determines a unique mapping, $f^\dagger$, from finite, nonempty sets of (irredundant) paths into $\mathcal{A}$, defined by

$$\begin{aligned} f^\dagger(\{s_1, \ldots, s_n\}) & \\ &= f(cl_i(\{s_1, \cdots, s_n\})) \\ &= f(cl_i(\{s_1\}) \oplus_{\mathcal{A}} \cdots \oplus_{\mathcal{A}} cl_i(\{s_n\})), \end{aligned}$$

for $n \geq 1$. Further, f is a homomorphism iff $f^\dagger$ satisfies

i. $f^\dagger(\{s_1, \cdots, s_n\}) = f^\dagger(\{s_1\}) \oplus_{\mathcal{A}} \cdots \oplus_{\mathcal{A}} f^\dagger(\{s_n\})$, for $n > 1$,

ii. $f^\dagger(\{\langle\varepsilon, 0\rangle\}) = 0_{\mathcal{A}}$,

iii. $f^\dagger(\{\langle\varepsilon, 1\rangle\}) = 1_{\mathcal{A}}$, and

iv. $f^\dagger(\{\langle\alpha\bar{\alpha}, j\rangle\}) = \alpha_{\mathcal{A}}(f^\dagger(\{\langle\bar{\alpha}, j\rangle\}))$,

and any such $f^\dagger$ determines f. The "only if" direction is straigthforward. For the converse direction, note first that any element in the range of $f^\dagger$ is equal to a sum $\sum X$, for X a finite, nonempty set of *traces* which are elements obtainable using only 0, 1 and operators α ∈Act. Next note that whenever $cl_i(S_1) = cl_i(S_2)$, for S_1, S_2 finite, nonempty, then $f^\dagger(S_1) = f^\dagger(S_2)$. Thirdly, if s_1, s_2 are paths, and $f^\dagger$ satisfies i.–iv., $f^\dagger(\{s_1 \cdot s_2\}) = f^\dagger(\{s_1\}) \cdot f^\dagger(\{s_2\})$. Then we check, that f is a homomorphism—we take only the case for $\cdot$:

$$\begin{aligned} f(S_1 \cdot S_2) &= f^\dagger(\{s_1 \cdot s_2 \mid s_1 \in S'_1, s_2 \in S'_2, S'_1(S'_2) \\ &\quad \text{is finite and generates } S_1(S_2)\}) \\ &= \textstyle\sum_{\mathcal{A}}\{f^\dagger(\{s_1 \cdot s_2\}) \mid s_1 \in S'_1, s_2 \in S'_2\} \\ &= \textstyle\sum_{\mathcal{A}}\{f^\dagger(\{s_1\}) \cdot f^\dagger(\{s_2\}) \mid s_1 \in S'_1, \\ &\quad s_2 \in S'_2\} \\ &= \textstyle\sum_{\mathcal{A}}\{s_1 \cdot s_2 \mid s_1(s_2) \text{ a trace of} \\ &\quad f^\dagger(S'_1)(f^\dagger(S'_2))\} \\ &= f^\dagger(S'_1) \cdot f^\dagger(S'_2) \\ &= f(S_1) \cdot f(S_2). \end{aligned}$$

Then we are done, for i.–iv. determines $f^\dagger$, and hence f, uniquely. The check, that f preserves $\leq_i$ is straightforward.

Proof of theorem 3: Thm. 2 gives homomorphisms $[\![\cdot]\!]_i : \mathcal{P} \to \mathcal{D}_i$ for each $i \in \{1, 2, 3\}$. We must show these semantics to be fully abstract, i.e. that for all $p, q \in \mathcal{P}$, $p \preceq_i q$ iff $[\![p]\!]_i \leq_i [\![q]\!]_i$, $i \in \{1, 2, 3\}$. For the if-direction it suffices to show the (in-)equations valid w.r.t. the $\preceq_i$ — this is straightforward. For the converse direction note first that there are obvious relations of "may" and "must", inducing the relations $\preceq_i^{\mathcal{D}}$, defined on the $\mathcal{D}_i$. Using thm. 2 and the if-direction it may be shown that $p \preceq_i q$ iff $[\![p]\!]_i \preceq_i^{\mathcal{D}} [\![q]\!]_i$. To complete the proof we assume $[\![p]\!]_i \not\leq_i [\![q]\!]_i$ and show that then $[\![p]\!]_i \not\preceq_i^{\mathcal{D}} [\![q]\!]_i$. The case for $i = 3$ reduces to those for $i \in \{1, 2\}$. We outline the proof for $i = 2$. So, let S_1, S_2 be upper, f.g. and nonempty sets of irredundant paths, and assume $S_1 \not\leq_2 S_2$. Then there is a path $s_2 \in S_2$ s.t. for all $s_1 \in S_1$, $s_1 \not\leq s_2$. We proceed by induction on the complexity of the least generating subset of S_2. We take only the case where $s_2 = \langle\alpha_2\bar{\alpha}_2, j_2\rangle$. Then every $s_1 \in S_1$ must have the form either $s_1 = \langle\varepsilon, 1\rangle$ or $s_1 = \langle\alpha_1\bar{\alpha}_1, j_1\rangle$. As S_1 is f.g. we can find a finite, nonempty set A ⊆Act s.t. S_1 **must** $\sum A$ and $\alpha_2 \in A$. For $k = 1, 2$ let $S'_k = \{\langle\bar{\alpha}, k\rangle \mid \langle\alpha_2\bar{\alpha}, k\rangle \in S_k\}$. If $S'_1 = \emptyset$ all we have to do is to find a test t_{α_2} s.t. not S'_2 **must** t_{α_2}. If $S'_1 \neq \emptyset$ note that $S'_1, S'_2 \in D_2$ and that $S'_1 \not\leq_2 S'_2$. It may be seen, that the complexity of S'_2 is strictly less than that of S_2, hence we can apply the inductive hypothesis to find a test t_{α_2} s.t. S'_1 **must** t_{α_2} and not S'_2 **must** t_{α_2}. This is then combined with A, obtaining a test separating S_1 and S_2, and we are done.

REFERENCES

[1] K. R. Abrahamson, "*Decidability and expressiveness of logics of processes*," Ph.D. thesis, University of Washington, 1981.

[2] A. Avron, "The semantics and proof theory of linear logic," tech. rep., University of Edinburgh, ECS-LFCS-87-27, 1987.

[3] A. R. Anderson and N. D. Belnap, "*Entailment, the logic of relevance and necessity*," Vol. 1, Princeton University Press, 1975.

[4] S. Brookes, C. Hoare and A. Roscoe, "A theory of communicating sequential processes," *J. ACM* 31 (1984) 560–599.

[5] H. Barringer, R. Kuiper and A. Pnueli, "Now you may compose temporal logic specification", *Proc. ACM Symp. on Theory of Computing* (1984), 51–63.

[6] R. de Nicola and M. Hennessy, "Testing equivalences for processes," *Theor. comp. sci.* 34 (1984), 83–133.

[7] J. M. Dunn, "Relevance logic and entailment," in: D. Gabbay, F. Guenthner (eds.), *Handbook of phil. logic*, Vol III (D. Reidel, Dordrecht, 1986), 117–224.

[8] J.-Y. Girard, "Linear logic," *Theor. comp. sci.* 50 (1987), 1–101.

[9] S. Graf and J. Sifakis, "A logic for the description of nondeterministic programs and their properties," *Inf. and contr.* 68 (1986), 254–270.

[10] M. Hennessy and R. Milner, "Algebraic laws for nondeterminism and concurrency," *J. ACM* 32 (1985), 137–162.

[11] M. Hennessy and G. Plotkin, "Finite conjunctive nondeterminism," in: K.Voss, H.J.Genrich, G.Rozenberg (eds.), *Concurrency and nets*, (Springer, Berlin, 1987).

[12] C. Jones, "Specification and design of (parallel) programs," *Proc. IFIP* (1983), 321–332.

[13] J.K.Kennaway, "*Formal semantics of nondeterminism and parallelism*," Ph.D. Thesis, Univ. of Oxford (1981).

[14] K. Larsen, "A context dependent equivalence between processes," *Lecture notes in comp. sci.* 194 (Springer, Berlin, 1985) 373–382.

[15] R. Milner, "A calculus of communicating systems,"

Lecture notes in comp. sci. 92 (Springer, Berlin, 1980).

[16] R. Milner, “Calculi for synchrony and asynchrony,” *Theor. comp. sci.* 25 (1983) 267–310.

[17] H. Ono and Y. Komori, “Logics without the contraction rule,” *J. symb. logic* 50 (1985) 169–201.

[18] G. Plotkin, “Algebras for internal and external nondeterminism,” manuscript, University of Edinburgh (1987).

[19] C. Stirling, “A generalization of Owicki-Gries’s Hoare logic for a concurrent while language,” *Theor. comp. sci.* To appear.

[20] C. Stirling, “Modal logics for communicating systems,” *Theor. comp. sci.* 49 (1987) 311–347.

[21] A. Urquhart, “Semantics for relevant logics,” *J. Symb. Logic* 37 (1972) 159–169.

[22] G. Winskel, “A complete proof system for SCCS with modal assertions,” *Lecture notes in comp. sci.* 206 (Springer, Berlin, 1985) 392–410.