Angelicism in the Theory of Reactive Processes

Pedro Ribeiro and Ana Cavalcanti

Department of Computer Science, University of York, UK
{pfr,alcc}@cs.york.ac.uk

Abstract The concept of angelic nondeterminism has traditionally been employed in the refinement calculus. Despite different notions having been proposed in the context of process algebras, namely Communicating Sequential Processes (CSP), the analogous counterpart to the angelic choice operator of the monotonic predicate transformers, has been elusive. In order to consider this concept in the context of reactive processes, we introduce a new theory in the setting of Hoare and He's Unifying Theories of Programming (UTP). Based on a theory of designs with angelic nondeterminism previously developed, we show how these processes can be similarly expressed as reactive designs. Furthermore, a Galois connection is established with the existing theory of reactive processes and a bijection is also found with respect to the subset of non-angelic processes.

Keywords: formal specification, reactive processes, CSP, UTP

1 Introduction

In the refinement calculus [1,2,3], angelic nondeterminism is defined as the least upper bound of the lattice of monotonic predicate transformers and is the dual operator of demonic nondeterminism. The angelic nature pertains to the embodied notion of nondeterminism that is aversive to failure. In theories of correctness for sequential programs, this corresponds to evading abortion, if possible.

In the context of reactive and concurrent systems, however, the notions of angelic nondeterminism considered so far in the literature, have been notably different. Tyrrell et al. [4] have proposed an axiomatized algebra of processes resembling CSP where external choice is referred to as angelic choice, however, in their model deadlock is not distinguishable from divergence.

Roscoe [5] has proposed an angelic choice operator $P \boxdot Q$ through operational combinator semantics for CSP. It is an alternative to the external choice operator that behaves as follows: as long as the environment chooses events offered by both P and Q, then the choice between P and Q is unresolved. The possibility of divergence or otherwise has no effect on the choice. A suitable notion of angelic nondeterminism for reactive processes would ideally also avoid divergence.

The UTP of Hoare and He [6] is a suitable framework in order to study the concept of angelic nondeterminism in a theory of reactive processes. Although characterising both demonic and angelic nondeterminism in a relational setting is not trivial [7], an encoding of upward-closed binary multirelations [8] can be used in order to define a theory of designs with both as we showed in [9].

In this paper we propose a natural extension to the UTP theories of reactive processes that characterises CSP in the UTP, using the principles of the theory in [9] in order to support both notions of nondeterminism. In this new theory, the angelic choice $(a \rightarrow \mathit{Skip}) \sqcup (a \rightarrow \mathit{Chaos})$ is actually resolved in favour of $a \rightarrow \mathit{Skip}$. Angelic nondeterminism corresponds to the least upper bound of the lattice, while demonic nondeterminism is the greatest lower bound.

We show how processes in this new theory can similarly be expressed as reactive designs with angelic nondeterminism [9], just like processes in the theory of [6,10] can be expressed as reactive designs. Furthermore, a Galois connection is also established with the existing theory and a bijection is found with respect to the subset of our theory that does not exhibit angelic nondeterminism.

2 Preliminaries

The UTP [6] is an alphabetized, predicative theory of relations suitable for modelling different programming paradigms. Theories are characterised by three components: an alphabet, a set of healthiness conditions and a set of operators. The alphabet $\alpha(P)$ of a relation P is split into $in\alpha(P)$, which contains undashed variables corresponding to the initial observations, and $out\alpha(P)$ containing the dashed counterparts for after or final observations.

Refinement is defined as universal reverse implication. In the UTP, total correctness is characterised through the theory of designs [6,11], whose healthiness conditions are **H1** and **H2**. Every design P can be expressed in terms of pre and postcondition pairs, $(\neg\ P^f \vdash P^t)$, where $P^o = P[o/ok']$ and t and f correspond to *true* and *false*, respectively.

2.1 Angelic Designs

As discussed earlier, modelling of both angelic and demonic nondeterminism in the UTP can be achieved through an encoding of upward-closed binary multirelations [8] with non-homogeneous relations as proposed by Cavalcanti et al. [7]. In that theory, the alphabet consists of input program variables and a sole output variable ac' that is a set of final states available for angelic choice. Intuitively, the angelic choice over states corresponds to those in ac', while the demonic choice corresponds to the choice over the value of ac' itself.

Upward closure is enforced by the following healthiness condition, where v and v' refer to every variable other than ac and ac', respectively.

Definition 1. $\mathbf{PBMH}(P) \mathrel{\widehat{=}} P\ ;\ ac \subseteq ac' \wedge v' = v$

A fixed point of **PBMH** requires that if it is possible for P to provide some set of final states ac' for angelic choice, then any superset can also be established. In the theory in [7], there are no other variables v', and here we consider a more general class of theories. **PBMH** can be restated as shown in Lemma 1.

Lemma 1. $\mathbf{PBMH}(P) = \exists\, ac_0 \bullet P[ac_0/ac'] \wedge ac_0 \subseteq ac'$

This function commutes with both **H1** and **H2** of the theory of designs. Proofs of these and all other results can be found in [12].

Following this approach we have defined a theory of angelic designs [9]. Its alphabet includes ok and ok', a single input state s and a set of final states ac'. A state is a record whose components are program variables.

The healthiness conditions of our theory of angelic designs are **H1** and **H2** from Hoare and He's theory of designs [6], and **A**, whose definition is the functional composition of **A0** and **A1** as reproduced below [9].

Definition 2.

$$\mathbf{A0}(P) \mathrel{\widehat{=}} P \wedge ((ok \wedge \neg P^f) \Rightarrow (ok' \Rightarrow ac' \neq \emptyset))$$
$$\mathbf{A1}(P) \mathrel{\widehat{=}} (\neg\, \mathbf{PBMH}(P^f) \vdash \mathbf{PBMH}(P^t))$$
$$\mathbf{A}(P) \mathrel{\widehat{=}} \mathbf{A0} \circ \mathbf{A1}(P)$$

The healthiness condition **A0** requires that when a design terminates successfully, then there must be some final state in ac' available for angelic choice. While **A1** requires that the final set of states in both the postcondition and the negation of the precondition are upward closed. We observe that **A1** can also be expressed as the application of **PBMH** to the whole of the design P.

Since **H1**, **H2** and **A** commute, and these functions are all idempotent and monotonic [9], so is the functional composition of **H1**, **H2** and **A**. Furthermore, because **A** is idempotent and monotonic, and the theory of designs is a complete lattice, so is our theory of **A**-healthy designs [6].

Amongst the operators introduced in [9] we single out sequential composition as the least trivial due to the non-homogeneous nature of the relations. Its definition is layered upon the sequential composition operator $;_{\mathcal{A}}$ of [7], whose definition, in the context of this theory, we reproduce below.

Definition 3. $P \;;_{\mathcal{A}}\; Q \mathrel{\widehat{=}} P[\{s \mid Q\}/ac']$

The resulting set of angelic choices is that of Q, such that they can be reached from an initial state of Q that is available for P as a set ac' of angelic choices. This use of substitution can be interpreted as back propagating the necessary information concerning the final states.

For instance, consider the following example, where angelic choice ($\sqcup$) is the least upper bound of the lattice. The choice is between the assignment of *true* or *false* to the program variable b, as denoted by t and f, respectively. This is sequentially composed with the program that maintains the value of b provided that the initial value of b is *true*, and otherwise aborts.

Example 1.

$$\begin{aligned}
&(\{b \mapsto t\} \in ac' \sqcup \{b \mapsto f\} \in ac') \;;_{\mathcal{A}}\; (s.b \Rightarrow s \in ac') && \{\text{Definition of } \sqcup\}\\
&= (\{b \mapsto t\} \in ac' \wedge \{b \mapsto f\} \in ac') \;;_{\mathcal{A}}\; (s.b \Rightarrow s \in ac') && \{\text{Definition of } ;_{\mathcal{A}}\}\\
&= (\{b \mapsto t\} \in ac' \wedge \{b \mapsto f\} \in ac')[\{s \mid s.b \Rightarrow s \in ac'\}/ac'] && \{\text{Substitution}\}
\end{aligned}$$

$$= \{b \mapsto t\} \in \{s \mid s.b \Rightarrow s \in ac'\} \wedge \{b \mapsto f\} \in \{s \mid s.b \Rightarrow s \in ac'\}$$
$$\{\text{Property of sets and value of record component } b\}$$
$$= (\mathit{true} \Rightarrow \{b \mapsto t\} \in ac') \wedge (\mathit{false} \Rightarrow \{b \mapsto f\} \in ac') \qquad \{\text{Predicate calculus}\}$$
$$= \{b \mapsto t\} \in ac'$$

The only possible result is the assignment of *true* to b, since this avoids aborting.

2.2 Reactive Processes

Programs characterised by continuous interactions with their environment are modelled in the UTP using the theory of reactive processes [6,10]. In addition to the variables, ok and ok' of the theory of designs, this theory includes the variables *wait*, *tr*, *ref* and their dashed counterparts, that record information about interactions with the environment.

This is a theory where there are observations of intermediate states. The variable *wait* records whether the previous process is waiting for an interaction from the environment or, alternatively, has terminated. Similarly, *wait*$'$ ascertains this for the current process. The variable ok indicates whether the previous process is in a stable state, while ok' records this information for the current process. If a process is not in a stable state, then it is said to have diverged. A process only starts executing in a state where ok and $\neg$ *wait* are *true*. Successful termination is characterised by ok' and $\neg$ *wait*$'$ being *true*.

The actual interactions with the environment are represented using sequences of events, recorded by *tr* and *tr*$'$. The variable *tr* records the sequence of events that took place before the current process started, while *tr*$'$ records the intermediate or final sequence of events that can be observed. Finally, *ref* and *ref*$'$ record the set of events that may be refused by the process. Refusal sets allow the appropriate modelling of deadlock [13].

Healthiness Conditions The theory of reactive processes **R** is characterised by the functional composition of three healthiness conditions [6,10] below.

Definition 4 (Reactive Process).

$$\mathbf{R1}(P) \mathrel{\widehat{=}} P \wedge tr \leq tr'$$
$$\mathbf{R2}(P) \mathrel{\widehat{=}} P[\langle\rangle, tr' - tr/tr, tr']$$
$$\mathbf{R3}(P) \mathrel{\widehat{=}} \mathbb{I}_{rea} \lhd \mathit{wait} \rhd P$$
$$\mathbf{R}(P) \mathrel{\widehat{=}} \mathbf{R3} \circ \mathbf{R1} \circ \mathbf{R2}(P)$$

R1 requires that in all circumstances the only change that can be observed in the final trace of events *tr*$'$ is an extension of the initial sequence *tr*, while **R2** requires that a process must not impose any restriction on the initial value of *tr*. Finally, **R3** requires that if the previous process is waiting for an interaction with the environment, that is *wait* is *true*, then the process behaves as the identity of the theory $\mathbb{I}_{rea}$ [6,10], otherwise it behaves as P.

CSP Processes as Reactive Designs The theory of CSP can be described by reactive processes that in addition also satisfy two other healthiness conditions, **CSP1** and **CSP2**, whose definitions are reproduced below [6,10].

Definition 5 (CSP).

$$\mathbf{CSP1}(P) \mathrel{\widehat{=}} P \vee \mathbf{R1}(\neg\, ok)$$
$$\mathbf{CSP2}(P) \mathrel{\widehat{=}} P \;;\; ((ok \Rightarrow ok') \wedge tr' = tr \wedge ref' = ref \wedge wait' = wait)$$

The first healthiness condition **CSP1** requires that if the previous process has diverged, that is, *ok* is *false*, then extension of the trace is the only guarantee. **CSP2** is **H2** restated with the extended alphabet of reactive processes.

A process that is **R**, **CSP1** and **CSP2**-healthy can be described in terms of a design as proved in [6,10]. We reproduce this result below, where we use the notation $P^o_w = P[o, w/ok', wait]$.

Theorem 1 (Reactive Design). *For every CSP process* P, $\mathbf{R}(\neg\, P^f_f \vdash P^t_f) = P$

This result is important as it allows CSP processes to be specified in terms of pre and postconditions, such as is the case for sequential programs, while the healthiness condition **R** enforces the required reactive behaviour.

3 A Natural Extension of the Theory of Reactive Processes

Based on the concept of states, as introduced in the theory of angelic designs [9], we explore a new model where the observational variables of the theory of reactive processes are encoded as state components.

Definition 6 (Alphabet).

$$ok, ok' : \{true, false\}, s : State, ac' : \mathbb{P}\, State$$
$$\mathrm{dom}\, State = \{tr, ref, wait\}$$

In addition to a single initial state s, a set of final states ac', and the observational variables *ok* and ok' that record stability, we require that every *State* has record components of name *tr*, *wait* and *ref*. This enables the angelic choice over the final or intermediate observations of *tr*, *ref* and *wait*.

3.1 Healthiness Conditions for Reactive Angelic Processes

Since this is a theory with angelic nondeterminism, relations need to satisfy **PBMH**, that is the set of final states ac' must be upward-closed. Furthermore, reactive processes must also satisfy the counterpart properties to **R** in the new model. In this section, we restate all the properties enforced by **R**, namely we define healthiness conditions **RA1**, **RA2** and **RA3**.

RA1 The first property of interest that underpins the theory of reactive processes is the notion that the history of events observed cannot be undone. In general, for any initial state x, the set of all final states that satisfy this property is given by $States_{tr \leq tr'}(x)$ as defined below.

Definition 7. $States_{tr \leq tr'}(x) \mathrel{\widehat{=}} \{z : State \mid x.tr \leq z.tr\}$

This definition is used for introducing the first healthiness condition, **RA1**, that not only enforces this notion for final states in ac', but also requires that there is some final state satisfying this property available for angelic choice.

Definition 8. $\mathbf{RA1}(P) \mathrel{\widehat{=}} (P \wedge ac' \neq \emptyset)[States_{tr \leq tr'}(s) \cap ac'/ac']$

A consequence of the definition of **RA1** is that it also enforces **A0**.

Theorem 2. $\mathbf{RA1} \circ \mathbf{A0}(P) = \mathbf{RA1}(P)$

Although **A0** only requires ac' not to be empty in the postcondition of a design, **RA1** requires this under all circumstances.

The function **RA1** distributes through conjunction and disjunction.

Theorem 3. $\mathbf{RA1}(P \wedge Q) = \mathbf{RA1}(P) \wedge \mathbf{RA1}(Q)$

Theorem 4. $\mathbf{RA1}(P \vee Q) = \mathbf{RA1}(P) \vee \mathbf{RA1}(Q)$

Furthermore, the operator $;_{\mathcal{A}}$ is closed under **RA1**, provided that both operands are upward-closed and **RA1**-healthy. This is an important property as the definition for sequential composition in our theory is also based on $;_{\mathcal{A}}$.

Theorem 5. *Provided P and Q are* **RA1** *and* **PBMH***-healthy.*

$$\mathbf{RA1}(P ;_{\mathcal{A}} Q) = P ;_{\mathcal{A}} Q$$

For every healthiness condition of the theory, the upward-closure enforced by **PBMH** must be maintained. Theorem 6 establishes this for **RA1**.

Theorem 6. *Provided P is* **PBMH***-healthy.* $\mathbf{PBMH} \circ \mathbf{RA1}(P) = \mathbf{RA1}(P)$

However, **PBMH** and **RA1** do not commute in general. We consider the following counter-example where the healthiness conditions are applied to the relation $ac' = \emptyset$, which is not **PBMH**-healthy.

Example 2.

$$\begin{aligned}
&\mathbf{RA1} \circ \mathbf{PBMH}(ac' = \emptyset) && \{\text{Definition of } \mathbf{PBMH} \text{ (Lemma 1)}\}\\
&= \mathbf{RA1}(\exists\, ac_0 \bullet ac_0 = \emptyset \wedge ac_0 \subseteq ac') && \{\text{One-point rule and property of sets}\}\\
&= \mathbf{RA1}(\mathit{true})
\end{aligned}$$

$$\begin{aligned}
&\mathbf{PBMH} \circ \mathbf{RA1}(ac' = \emptyset) && \{\text{Definition of } \mathbf{RA1}\}\\
&= \mathbf{PBMH}((ac' = \emptyset \wedge ac' \neq \emptyset)[States_{tr \leq tr'}(s) \cap ac'/ac']) && \{\text{Predicate calculus}\}\\
&= \mathbf{PBMH}(\mathit{false}) && \{\text{Definition of } \mathbf{PBMH} \text{ (Lemma 1)}\}\\
&= \mathit{false}
\end{aligned}$$

In the first case, the application of **PBMH** yields *true*. The result of the functional composition is then $\mathbf{RA1}(\mathit{true})$. On the other hand, in the second case, there is a contradiction that yields *false*.

RA2 The next healthiness condition of interest is **RA2**, that requires a process to be insensitive to the initial trace of events $s.tr$. It is the counterpart to **R2** of the original theory of reactive processes, and is also defined using substitution.

Definition 9 (RA2).

$$\mathbf{RA2}(P) \mathrel{\widehat{=}} P\left[s \oplus \{tr \mapsto \langle\rangle\}, \left\{ z \;\middle|\; \begin{array}{l} z \in ac' \wedge s.tr \leq z.tr \\ \bullet\ z \oplus \{tr \mapsto z.tr - s.tr\} \end{array} \right\} \middle/ s, ac'\right]$$

It sets the component tr in the initial state s to the empty sequence, and consequently changes ac' as follows: the set of final states ac' is restricted to those states z whose traces are a suffix of $s.tr$, and furthermore, their trace is set to the difference with respect to the initial trace $s.tr$.

Since substitution distributes through conjunction and disjunction, so does the healthiness condition **RA2**.

Theorem 7. $\mathbf{RA2}(P \wedge Q) = \mathbf{RA2}(P) \wedge \mathbf{RA2}(Q)$

Theorem 8. $\mathbf{RA2}(P \vee Q) = \mathbf{RA2}(P) \vee \mathbf{RA2}(Q)$

Furthermore, the operator $;_{\mathcal{A}}$ is also closed under **RA2**.

Theorem 9. *Provided P and Q are* **RA2***-healthy.*

$$\mathbf{RA2}(P ;_{\mathcal{A}} Q) = P ;_{\mathcal{A}} Q$$

A consequence of the definition of **RA2** is that applying it to the non-empty set of final states ac' is equivalent to applying **RA1** to the relation *true*.

Theorem 10. $\mathbf{RA2}(ac' \neq \emptyset) = \mathbf{RA1}(true)$

This results sheds light on the relationship between **RA2** and **RA1**, as in fact, these functions are commutative.

Theorem 11. $\mathbf{RA1} \circ \mathbf{RA2}(P) = \mathbf{RA2} \circ \mathbf{RA1}(P)$

Finally, Theorem 12 establishes that **RA2** maintains the upward-closure.

Theorem 12. *Provided P is* **PBMH***-healthy.* $\mathbf{PBMH} \circ \mathbf{RA2}(P) = \mathbf{RA2}(P)$

This concludes our discussion of **RA2** and its most important properties.

RA3 As in the theory of reactive processes, it is necessary to ensure that a process cannot start before the previous process has finished interacting with the environment. The counterpart to **R3** in the new theory is **RA3**. Before exploring its definition, we introduce the identity $\mathbb{I}_{\mathcal{R}ac}$.

Definition 10. $\mathbb{I}_{\mathcal{R}ac} \mathrel{\widehat{=}} (\mathbf{RA1}(\neg\, ok) \vee (ok' \wedge s \in ac'))$

Similarly to the reactive identity $\mathbb{I}_{rea}$, the behaviour for an unstable state $\neg\, ok$ is given by **RA1**, that is, there must be at least one final state in ac' whose trace is a suffix of the initial trace $s.tr$. Otherwise, the process is stable, with ok' being *true*, and the initial state s is in the set of final states ac'.

Having defined the identity, we introduce the definition of **RA3** below.

Definition 11. $\mathbf{RA3}(P) \mathrel{\widehat{=}} \mathbb{I}_{\mathcal{R}ac} \lhd s.wait \rhd P$

This definition resembles that of the original theory, the difference being in the identity $\mathbb{I}_{\mathcal{R}ac}$ and the fact that *wait* is a component of the initial state s. Using Leibniz's substitution, it is possible to prove the following Lemma 2, where $P^o_w = P[o, s \oplus \{wait \mapsto w\}/s, ok']$.

Lemma 2. $\mathbf{RA3}(P) = \mathbf{RA3}(P_f)$

The function **RA3** also distributes through both conjunction and disjunction.

Theorem 13. $\mathbf{RA3}(P \wedge Q) = \mathbf{RA3}(P) \wedge \mathbf{RA3}(Q)$

Theorem 14. $\mathbf{RA3}(P \vee Q) = \mathbf{RA3}(P) \vee \mathbf{RA3}(Q)$

In addition, the operator $;_{\mathcal{A}}$ is also closed under **RA3** provided that the second process is also **RA1**-healthy. This is not a problem since the theory of interest is characterised by the functional composition of all the healthiness conditions.

Theorem 15. *Provided P and Q are* **RA3***-healthy and Q is* **RA1***-healthy.*

$$\mathbf{RA3}(P ;_{\mathcal{A}} Q) = P ;_{\mathcal{A}} Q$$

Furthermore, as required, **RA3** maintains the upward-closure.

Theorem 16. *Provided P is* **PBMH***-healthy.* $\mathbf{PBMH} \circ \mathbf{RA3}(P) = \mathbf{RA3}(P)$

The identity $\mathbb{I}_{\mathcal{R}ac}$ is a fixed point of every healthiness condition, including **RA1**, **RA2**, **RA3** and **PBMH**. Finally, **RA3** commutes with both **RA1** and **RA2**.

Theorem 17. $\mathbf{RA3} \circ \mathbf{RA1}(P) = \mathbf{RA1} \circ \mathbf{RA3}(P)$

Theorem 18. $\mathbf{RA3} \circ \mathbf{RA2}(P) = \mathbf{RA2} \circ \mathbf{RA3}(P)$

This concludes our discussion of the most important properties of **RA3**.

RA The new theory of reactive processes that we define here is characterised by the functional composition of the healthiness conditions **RA3**, **RA2**, **RA1** and **PBMH**. In order to maintain the parallel with the original theory of reactive processes, we define part of this composition as **RA** below.

Definition 12. $\mathbf{RA}(P) \mathrel{\widehat{=}} \mathbf{RA1} \circ \mathbf{RA2} \circ \mathbf{RA3}(P)$

The order of the functional composition is not important since these functions commute, except for **PBMH** that does not necessarily commute with every function, so it must be applied in the first instance.

Since all of the healthiness conditions **RA1**, **RA2** and **RA3** are idempotent and monotonic, so is **RA**. Similarly, since all those functions distribute through conjunction and disjunction, so does **RA**. Finally, **RA** maintains upward-closure since all of the **RA** healthiness conditions do so.

3.2 CSP Processes with Angelic Nondeterminism

As mentioned before, in the UTP, CSP processes are characterised as reactive processes that, in addition, satisfy the healthiness conditions **CSP1** and **CSP2**. In order to define a theory for CSP processes with angelic nondeterminism we follow a similar approach by introducing two healthiness conditions.

CSPA1 The first healthiness condition of interest is **CSPA1**, that is, the counterpart to **CSP1** in the original theory of CSP processes.

Definition 13. $\mathbf{CSPA1}(P) \mathrel{\widehat{=}} P \vee \mathbf{RA1}(\neg\ ok)$

A CSP process with angelic nondeterminism P is required to observe **RA1** when in an unstable state. For a **RA**-healthy process, this property is already enforced by **RA1** under all circumstances. Theorem 19 shows that this behaviour can also be described as the functional composition of **RA1** after **H1**.

Theorem 19. $\mathbf{RA1} \circ \mathbf{CSPA1}(P) = \mathbf{RA1} \circ \mathbf{H1}(P)$

Proof.

$$
\begin{aligned}
&\mathbf{RA1} \circ \mathbf{H1}(P) && \{\text{Definition of } \mathbf{H1}\}\\
&= \mathbf{RA1}(ok \Rightarrow P) && \{\text{Predicate calculus and Theorem 4}\}\\
&= \mathbf{RA1}(\neg\ ok) \vee \mathbf{RA1}(P) && \{\mathbf{RA1}\text{-idempotent}\}\\
&= \mathbf{RA1} \circ \mathbf{RA1}(\neg\ ok) \vee \mathbf{RA1}(P) && \{\text{Theorem 4}\}\\
&= \mathbf{RA1}(\mathbf{RA1}(\neg\ ok) \vee P) && \{\text{Definition of } \mathbf{CSPA1}\}\\
&= \mathbf{RA1} \circ \mathbf{CSPA1}(P) && \square
\end{aligned}
$$

The function **CSPA1** is idempotent and monotonic.

Theorem 20. *Provided P is* **PBMH***-healthy.*

$$\mathbf{PBMH} \circ \mathbf{CSPA1}(P) = \mathbf{CSPA1}(P)$$

Furthermore, it preserves the upward closure as required by **PBMH**.

CSPA2 The last healthiness condition of interest is the counterpart to **CSP2**. This is defined as **H2** with the extended alphabet that includes s and ac'.

Definition 14. $\mathbf{CSPA2}(P) \mathrel{\widehat{=}} \mathbf{H2}(P)$

This healthiness condition satisfies the same properties as **H2**. It can alternatively be defined using the J-split of [11].

RAP The theory of CSP processes in the new model is defined by **RAP**, the functional composition of all the healthiness conditions of interest.

Definition 15 (Reactive Angelic Process).

$$\mathbf{RAP}(P) \mathrel{\widehat{=}} \mathbf{RA} \circ \mathbf{CSPA1} \circ \mathbf{CSPA2} \circ \mathbf{PBMH}(P)$$

The fixed points of **RAP** are the reactive angelic processes. Since **PBMH** and **RA1** do not commute, **PBMH** is applied first. Every such process P can be expressed as $\mathbf{RA} \circ \mathbf{A}(\neg\, P^f_f \vdash P^t_f)$ as shown by the following Theorem 21.

Theorem 21. $\mathbf{RAP}(P) = \mathbf{RA} \circ \mathbf{A}(\neg\, P^f_f \vdash P^t_f)$

Proof.

$$\begin{aligned}
&\mathbf{RAP}(P) && \{\text{Definition of } \mathbf{RAP}\}\\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1} \circ \mathbf{CSPA1} \circ \mathbf{CSPA2} \circ \mathbf{PBMH}(P) && \{\text{Theorem 19}\}\\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1} \circ \mathbf{H1} \circ \mathbf{CSPA2} \circ \mathbf{PBMH}(P) && \{\mathbf{CSPA2} \text{ is } \mathbf{H2}\}\\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1} \circ \mathbf{H1} \circ \mathbf{H2} \circ \mathbf{PBMH}(P) && \{\text{Theorem 2}\}\\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1} \circ \mathbf{A0} \circ \mathbf{H1} \circ \mathbf{H2} \circ \mathbf{PBMH}(P) && \{\text{Theorems 34 and 35}\}\\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1} \circ \mathbf{A0} \circ \mathbf{PBMH} \circ \mathbf{H1} \circ \mathbf{H2}(P) && \{\text{Definition of design}\}\\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1} \circ \mathbf{A0} \circ \mathbf{PBMH}(\neg\, P^f \vdash P^t) && \{\text{Definition of } \mathbf{A}\}\\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1} \circ \mathbf{A}(\neg\, P^f \vdash P^t) && \{\text{Theorems 11, 17 and 18}\}\\
&= \mathbf{RA1} \circ \mathbf{RA2} \circ \mathbf{RA3} \circ \mathbf{A}(\neg\, P^f \vdash P^t) && \{\text{Lemmas 2 and 6}\}\\
&= \mathbf{RA1} \circ \mathbf{RA2} \circ \mathbf{RA3} \circ \mathbf{A}((\neg\, P^f \vdash P^t)_f) && \{\text{Substitution}\}\\
&= \mathbf{RA1} \circ \mathbf{RA2} \circ \mathbf{RA3} \circ \mathbf{A}(\neg\, P^f_f \vdash P^t_f) && \{\text{Definition of } \mathbf{RA}\}\\
&= \mathbf{RA} \circ \mathbf{A}(\neg\, P^f_f \vdash P^t_f) && \square
\end{aligned}$$

That is, such processes can be specified as the image of an **A**-healthy design through the function **RA**. This is a result similar to that obtained for CSP processes as the image of designs through **R** [6,10]. Since both **RA** and **A** are monotonic and idempotent, and the theory of designs is a complete lattice [6], so is the theory of reactive angelic processes.

3.3 Operators

Having discussed the healthiness conditions, in this section we present the corresponding definition of the most important operators of CSP in the new model. The original operators of CSP are distinguished with the subscript $_{\mathcal{CSP}}$.

Chaos The first process of interest is that corresponding to divergence.

Definition 16. $\mathit{Chaos} \mathrel{\widehat{=}} \mathbf{RA} \circ \mathbf{A}(\mathit{false} \vdash ac' \neq \emptyset)$

This is characterised by *Chaos*, whose precondition is always *false* and postcondition requires ac' not to be empty.

Stop The following process captures the notion of deadlock.

Definition 17. $\mathit{Stop} \mathrel{\widehat{=}} \mathbf{RA} \circ \mathbf{A}(\mathit{true} \vdash \circledcirc{\in}^{y}_{ac'}(y.tr = s.tr \wedge y.\mathit{wait}))$

The precondition is *true*, while the postcondition requires the process to always be waiting for the environment and keep the trace of events unchanged. In this new model we introduce the following auxiliary predicate.

Definition 18. $\circledcirc{\in}^{y}_{ac'}(P) \mathrel{\widehat{=}} \exists\, y \bullet y \in ac' \wedge P$

This definition requires that there is a state y available for angelic choice in ac' satisfying P. In the upward-closed binary multirelational encoding of our theory, it is the distributed intersection over all possible values of ac' which constitutes the actual final states available to the angel. Using this notation, the definitions of the CSP operators are very similar. It can be further extrapolated to other important CSP operators, such as external choice, parallelism and hiding.

Event Prefixing Prefixing is defined in a similar form as in the theory of CSP.

Definition 19.

$$a \rightarrow \mathit{Skip} \mathrel{\widehat{=}} \mathbf{RA} \circ \mathbf{A}\left(\mathit{true} \vdash \circledcirc{\in}^{y}_{ac'}\left(\begin{array}{l}(y.tr = s.tr \wedge a \notin y.\mathit{ref}) \\ \lhd y.\mathit{wait} \rhd \\ (y.tr = s.tr \frown \langle a \rangle)\end{array}\right)\right)$$

The precondition is *true*, while the postcondition is split into two cases. When the process is waiting for an interaction from the environment, that is, *wait* is true, then a is not in the set of refusals and the trace is kept unchanged. While in the second case, the process has interacted with the environment, and so the only guarantee is that the event a is part of the trace.

Demonic Choice The internal choice, also known as demonic choice, is defined using the greatest lower bound of the lattice, which is disjunction.

Definition 20. $P \sqcap Q \mathrel{\widehat{=}} P \vee Q$

For processes that are **RAP**-healthy, this result can also be turned into a **RAP** process that depends on the pre and postconditions of P and Q, respectively [12].

Sequential Composition The operator for sequential composition is perhaps the most challenging due to the use of non-homogeneous relations. We follow the approach used for the theory of angelic designs [9].

Definition 21. $P \mathbin{;_{\mathcal{R}ac}} Q = \exists\, ok_0 \bullet P[ok_0/ok'] \mathbin{;_{\mathcal{A}}} Q[ok_0/ok]$

This definition is layered upon the sequential composition operator $;_{\mathcal{A}}$ of [7] as introduced earlier. Finally, for processes that are **RAP**-healthy, sequential composition also yields a **RAP** process as shown in Theorem 22.

Theorem 22. *Provided P and Q are* **RAP***-healthy.*

$$
\begin{array}{l}
P \mathbin{;_{\mathcal{R}ac}} Q \\
= \\
\mathbf{RA} \circ \mathbf{A} \left(\begin{array}{l} \left(\begin{array}{l} \neg\, (\mathbf{RA1}(P_f^f) \mathbin{;_{\mathcal{A}}} \mathbf{RA1}(\mathit{true})) \\ \wedge \\ \neg\, (\mathbf{RA1}(P_f^t) \mathbin{;_{\mathcal{A}}} (\neg\, s.\mathit{wait} \wedge \mathbf{RA2} \circ \mathbf{RA1}(Q_f^f))) \end{array} \right) \\ \vdash \\ \mathbf{RA1}(P_f^t) \mathbin{;_{\mathcal{A}}} (s \in ac' \lhd s.\mathit{wait} \rhd (\mathbf{RA2} \circ \mathbf{RA1}(\neg\, Q_f^f \Rightarrow Q_f^t))) \end{array} \right)
\end{array}
$$

This is a result that resembles that for CSP, apart from the postcondition of the design. When $s.\mathit{wait}$ is *false*, and hence P_f^t has finished its interaction with the environment, the behaviour is given by the composition with $\mathbf{RA2} \circ \mathbf{RA1}(\neg\, Q_f^f \Rightarrow Q_f^t)$. In contrast with the result in CSP, this is an implication between the pre and postcondition of Q, instead of its postcondition.

In the theory of angelic designs, the sequential composition operator also has a similar implication in the postcondition that acts as a filter by eliminating final states of P that fail to satisfy the precondition of Q. In this theory, the implication only has a significant role when Q's precondition is not necessarily *true* and when there is angelic nondeterminism in P.

3.4 Angelic Choice

Following from the theory of angelic designs [9], we define angelic choice as the least upper bound of the lattice, which is conjunction.

Definition 22. $P \sqcup Q \mathrel{\widehat{=}} P \wedge Q$

Similarly, for processes that are **RAP**-healthy, this result is stated as follows.

Theorem 23. *Provided P and Q are* **RAP***-healthy.*

$$P \sqcup Q = \mathbf{RA} \circ \mathbf{A}(\neg\, P_f^f \vee \neg\, Q_f^f \vdash (\neg\, P_f^f \Rightarrow P_f^t) \wedge (\neg\, Q_f^f \Rightarrow Q_f^t))$$

The resulting process has as precondition the disjunction of the preconditions of P and Q, while the postcondition is the conjunction of two implications. In both cases, if either precondition of P or Q holds, then the corresponding postcondition is established. This is a result that follows closely that observed for the least upper bound of designs [6,11].

Futhermore, Theorem 24 establishes that *Chaos* is the unit with respect to the least upper bound of the lattice.

Theorem 24. $\mathit{Chaos} \sqcup \mathbf{RA} \circ \mathbf{A}(\neg\, P_f^f \vdash P_f^t) = \mathbf{RA} \circ \mathbf{A}(\neg\, P_f^f \vdash P_f^t)$

In order to understand the behaviour of angelic choice we consider the following examples. In Example 3 there is a choice between terminating and, deadlocking following event a, sequentially composed with *Chaos*.

Example 3. $((a \rightarrow \mathit{Skip} \mathbin{;_{\mathcal{R}ac}} \mathit{Stop}) \sqcup \mathit{Skip}) \mathbin{;_{\mathcal{R}ac}} \mathit{Chaos} = a \rightarrow \mathit{Skip} \mathbin{;_{\mathcal{R}ac}} \mathit{Stop}$

In this case, the angel avoids diverging by choosing not to terminate, but instead allowing the environment to perform event a and then deadlocking. In Example 4 there is a choice between terminating or diverging upon performing an a.

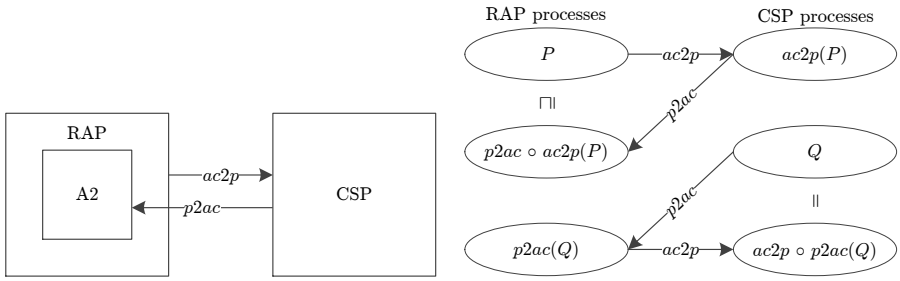
Example 4.

$$
\begin{aligned}
&(a \rightarrow \mathit{Skip}) \sqcup (a \rightarrow \mathit{Chaos}) && \{\text{Definition of prefixing}\}\\
&= \left(\begin{array}{l} \mathbf{RA} \circ \mathbf{A}\left(\mathit{true} \vdash \left(\begin{array}{l} \circledin^{y}_{ac'}(y.\mathit{wait} \wedge y.\mathit{tr} = s.\mathit{tr} \wedge a \notin y.\mathit{ref}) \\ \vee \\ \circledin^{y}_{ac'}(\neg\, y.\mathit{wait} \wedge y.\mathit{tr} = s.\mathit{tr} \frown \langle a \rangle) \end{array}\right)\right) \\ \sqcup \\ \mathbf{RA} \circ \mathbf{A}\left(\begin{array}{l} \neg \circledin^{y}_{ac'}(s.\mathit{tr} \frown \langle a \rangle \leq y.\mathit{tr}) \\ \vdash \\ \circledin^{y}_{ac'}(y.\mathit{wait} \wedge y.\mathit{tr} = s.\mathit{tr} \wedge a \notin y.\mathit{ref}) \end{array}\right) \end{array}\right) \\
& && \{\text{Theorem 23 and predicate calculus}\}\\
&= \mathbf{RA} \circ \mathbf{A}\left(\mathit{true} \vdash \left(\begin{array}{l} \left(\begin{array}{l} \circledin^{y}_{ac'}(y.\mathit{wait} \wedge y.\mathit{tr} = s.\mathit{tr} \wedge a \notin y.\mathit{ref}) \\ \vee \\ \circledin^{y}_{ac'}(\neg\, y.\mathit{wait} \wedge y.\mathit{tr} = s.\mathit{tr} \frown \langle a \rangle) \end{array}\right) \\ \wedge \\ \left(\begin{array}{l} \circledin^{y}_{ac'}(s.\mathit{tr} \frown \langle a \rangle \leq y.\mathit{tr}) \\ \vee \\ \circledin^{y}_{ac'}(y.\mathit{wait} \wedge y.\mathit{tr} = s.\mathit{tr} \wedge a \notin y.\mathit{ref}) \end{array}\right) \end{array}\right)\right) \\
& && \{\text{Predicate calculus}\}\\
&= \mathbf{RA} \circ \mathbf{A}\left(\mathit{true} \vdash \left(\begin{array}{l} \circledin^{y}_{ac'}(y.\mathit{wait} \wedge y.\mathit{tr} = s.\mathit{tr} \wedge a \notin y.\mathit{ref}) \\ \vee \\ \circledin^{y}_{ac'}(\neg\, y.\mathit{wait} \wedge y.\mathit{tr} = s.\mathit{tr} \frown \langle a \rangle) \end{array}\right)\right) \\
& && \{\text{Definition of prefixing}\}\\
&= a \rightarrow \mathit{Skip}
\end{aligned}
$$

The result is a process that following event a can only terminate, and thus avoids divergence. This property is an intuitive counterpart to the angelic choice operator of the refinement calculus, that instead considers choices over interactions.

4 Relationship with CSP

The theory that we propose can be related with the original UTP theory for CSP through a pair of linking functions that we introduce in this section: $ac2p$, that maps predicates from the theory of angelic reactive processes to predicates of the theory of CSP, and $p2ac$, mapping in the opposite direction. The relationship between the models of interest is illustrated in Figure 1(a), where each theory is labelled according to its healthiness conditions. The subset of reactive angelic processes that correspond exactly to CSP processes is characterised by **A2**, a healthiness condition we introduce in this section.



(a) Theories and links (b) Predicates and links

Figure 1: Relationship between theories

In Figure 1(b) the relationship between the predicates of each theory is illustrated. For a predicate P of the theory of angelic processes, the functional composition $p2ac \circ ac2p(P)$ yields a stronger predicate, while for a predicate Q of the CSP theory, the composition $ac2p \circ p2ac(Q)$ yields exactly the same predicate Q. Thus a Galois connection exists between the theories.

The definition of $ac2p$ is introduced in Section 4.1, while the definition of $p2ac$ is introduced in Section 4.2. In Section 4.3 we discuss the results pertaining to the functional composition of $p2ac$ and $ac2p$. Finally, in Section 4.4 we characterise the subset of angelic processes that do not exhibit angelic nondeterminism by introducing the healthiness condition **A2**. Furthermore, we establish that this subset is isomorphic to the CSP theory as suggested in Figure 1(a).

4.1 From Reactive Angelic Processes

The first function of interest is $ac2p$, whose goal is to collapse the set of final states into a single final state, and re-introduce the variables tr, ref and $wait$, and their dashed counterparts by performing appropriate substitutions. Its definition is presented below, where $in\alpha = \{tr, ref, wait\}$ and $out\alpha = \{tr', ref', wait'\}$.

Definition 23.

$$ac2p(P) \mathrel{\widehat{=}} \mathbf{PBMH}(P)[State_{\amalg}(in\alpha)/s] \mathbin{;_{\mathcal{A}}} \bigwedge x : out\alpha \bullet dash(s).x = x$$

First it enforces upward-closure by applying **PBMH** and then performs a substitution on the initial state s. This substitution introduces the initial variables of the CSP theory, which in the angelic theory are collected as fields of the record s. The variables ok and ok' are not changed as their meaning in both theories is exactly the same. For a set of variables $S\alpha$, $State_{\amalg}(S\alpha)$ is an identity record, whose components s_i are mapped to the respective variables s_i.

Definition 24. $State_{\amalg}(S\alpha) \mathrel{\widehat{=}} \{s_0 \mapsto s_0, \ldots, s_n \mapsto s_n\}$

As an example, we consider $(s.tr = \langle a \rangle \wedge ok)[State_{\amalg}(in\alpha)/s]$ whose result is $tr = \langle a \rangle \wedge ok$. If we consider the definition of **PBMH** and $;_{\mathcal{A}}$, then $ac2p$ can be rewritten as shown in the following Lemma 3.

Lemma 3. $ac2p(P) = \exists\, ac' \bullet \begin{pmatrix} P[State_{\Pi}(in\alpha)/s] \\ \wedge \\ \forall\, z \bullet z \in ac' \Rightarrow (\bigwedge x : out\alpha \bullet dash(z).x = x) \end{pmatrix}$

That is, the variable ac' is quantified away, and for each state z in the set ac', the output variables in $out\alpha$ are introduced and set to the respective values of the components of z. Since in our encoding, the components of a state are always undashed, we apply the function $dash(z)$ to z: its only purpose is to rename the components of z to their dashed counterparts. If there is more than one state in ac', then $ac2p$ yields *false* as no x variable introduced can take on more than one value. In general, this function maps predicates with more than one state in ac' to *false*. We consider the following example, where $ac2p$ is applied to the angelic choice between a prefixing on the event a or b, followed by deadlock.

Example 5. $ac2p(a \rightarrow Stop \sqcup b \rightarrow Stop) = a \rightarrow Stop \sqcup_{\mathcal{CSP}} b \rightarrow Stop$

The result is the least upper bound of the corresponding CSP processes.

Application of $ac2p$ after the healthiness conditions of the theory of reactive angelic processes yields healthy counterparts in the original theory as established by the following Theorem 25.

Theorem 25. *Provided P is* **PBMH***-healthy.* $ac2p \circ \mathbf{RA}(P) = \mathbf{R} \circ ac2p(P)$

Finally, these results allow us to establish the following result: the application of $ac2p$ to a reactive angelic process yields a reactive design.

Theorem 26. $ac2p \circ \mathbf{RA} \circ \mathbf{A}(\neg\, P_f^f \vdash P_f^t) = \mathbf{R}(\neg\, ac2p(P_f^f) \vdash ac2p(P_f^t))$

Proof.

$$
\begin{aligned}
& ac2p \circ \mathbf{RA} \circ \mathbf{A}(\neg\, P_f^f \vdash P_f^t) && \{\text{Theorem 36}\} \\
&= ac2p \circ \mathbf{RA} \circ \mathbf{PBMH}(\neg\, P_f^f \vdash P_f^t) && \{\text{Theorem 25}\} \\
&= \mathbf{R} \circ ac2p \circ \mathbf{PBMH}(\neg\, P_f^f \vdash P_f^t) && \{\text{Lemma 7}\} \\
&= \mathbf{R} \circ ac2p(\neg\, P_f^f \vdash P_f^t) && \{\text{Lemma 8}\} \\
&= \mathbf{R}(\neg\, ac2p(P_f^f) \vdash ac2p(P_f^t)) && \square
\end{aligned}
$$

This is a pleasing result that supports the reuse of results across the theories.

4.2 From CSP Processes

The mapping in the opposite direction, that is, from the theory of CSP to our theory is achieved through the function $p2ac$.

Definition 25. $p2ac(P) \mathrel{\widehat{=}} \exists\, z \bullet P[\mathbf{s}, \mathbf{z}/in\alpha, out\alpha] \wedge undash(z) \in ac'$

First, each variable in the set of input and output variables is replaced with the corresponding component of the initial state s and a final state z from the set of final states available for angelic choice. In general, for an arbitrary set of variables $S\alpha$, this substitution is defined as follows.

Definition 26. $P[\mathbf{z}/S\alpha] \mathrel{\widehat{=}} P[z.s_0, \ldots, z.s_n/s_0, \ldots, s_n]$

Each variable s_i in $S\alpha$ is replaced with $z.s_i$. As an example, we consider the substitution $(tr' = tr \wedge ok')[\mathbf{s}, \mathbf{z}/in\alpha, out\alpha]$, whose result is $z.tr' = s.tr \wedge ok'$. Since in our encoding states have undashed components, we require $undash(z)$ to be in ac'. The function $undash$ is the inverse of $dash$.

A consequence of the definition of $p2ac$ is that it requires ac' not to be empty. Furthermore, the result of $p2ac$ is also upward-closed as established by Lemma 4.

Lemma 4. $\mathbf{PBMH} \circ p2ac(P) = p2ac(P)$

The application of $ac2p$ to the healthiness conditions of the theory of reactive processes yields the corresponding healthiness conditions of our theory. As a result, we can establish that, in general, the application of $p2ac$ to a process P, characterised by $\mathbf{R}$, can be described by the functional composition of $\mathbf{RA}$ after $p2ac$ to the original process P, as established by Theorem 27.

Theorem 27. $p2ac \circ \mathbf{R}(P) = \mathbf{RA} \circ p2ac(P)$

The result of applying $p2ac$ to a reactive design is established below; $p2ac$ can be applied to the pre and postconditions separately, followed by $\mathbf{A}$ and $\mathbf{RA}$.

Theorem 28. $p2ac \circ \mathbf{R}(\neg\, P_f^f \vdash P_f^t) = \mathbf{RA} \circ \mathbf{A}(\neg\, p2ac(P_f^f) \vdash p2ac(P_f^t))$

Proof.

$$
\begin{aligned}
& p2ac \circ \mathbf{R}(\neg\, P_f^f \vdash P_f^t) && \{\text{Theorem 27 and definition of } \mathbf{RA}\} \\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1} \circ p2ac(\neg\, P_f^f \vdash P_f^t) && \{\text{Definition of } \mathbf{RA1}\} \\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1}((p2ac(\neg\, P_f^f \vdash P_f^t) \wedge ac' \neq \emptyset) && \{\text{Theorem 37}\} \\
&= \mathbf{RA3} \circ \mathbf{RA2} \circ \mathbf{RA1}((\neg\, p2ac(P_f^f) \vdash p2ac(P_f^t)) \wedge ac' \neq \emptyset) && \{\mathbf{RA1} \text{ and } \mathbf{RA}\} \\
&= \mathbf{RA}(\neg\, p2ac(P_f^f) \vdash p2ac(P_f^t)) && \{\text{Lemma 4}\} \\
&= \mathbf{RA}(\neg\, \mathbf{PBMH} \circ p2ac(P_f^f) \vdash \mathbf{PBMH} \circ p2ac(P_f^t)) && \{\text{Definition of } \mathbf{A1}\} \\
&= \mathbf{RA} \circ \mathbf{A1}(\neg\, p2ac(P_f^f) \vdash p2ac(P^t)) && \{\text{Definition of } \mathbf{RA} \text{ and Theorem 2}\} \\
&= \mathbf{RA} \circ \mathbf{A0} \circ \mathbf{A1}(\neg\, p2ac(P_f^f) \vdash p2ac(P^t)) && \{\text{Definition of } \mathbf{A}\} \\
&= \mathbf{RA} \circ \mathbf{A}(\neg\, p2ac(P_f^f) \vdash p2ac(P^t)) && \square
\end{aligned}
$$

This proof relies on the fact that **RA1** requires ac' not to be empty, and the fact that $p2ac$ ensures that this is the case. Furthermore, as already mentioned, the predicate resulting from applying $p2ac$ is upward-closed. This result enables CSP processes to be easily mapped into our theory by considering the mapping of the pre and postcondition of reactive designs separately.

4.3 A Galois Connection

The linking functions we have defined establish a Galois connection between the theories. In fact, when considering the mapping from the original theory of reactive processes, followed by the mapping in the opposite direction, we obtain an exact correspondence as described in Theorem 29.

Theorem 29. $ac2p \circ p2ac(P) = P$

This result establishes that our theory can accommodate the existing reactive processes appropriately, that is, those without angelic nondeterminism.

When considering the mapping in opposite direction we obtain the following.

Lemma 5. $p2ac \circ ac2p(P) = \exists\, ac_0, y \bullet P[ac_0/ac'] \wedge ac_0 \subseteq \{y\} \wedge y \in ac'$

The functional composition behaves as follows: if the set of final states ac_0 in P has more than one state, then the result of this composition is *false*, otherwise ac_0 is either a singleton or if, ac_0 is empty, any final state is in ac'. In other words, the mapping only preserves predicates whose set of angelic choices is either empty or a singleton, otherwise the result is *false*. We consider the following example, where Lemma 5 is applied to the process $a \rightarrow Stop \sqcup b \rightarrow Stop$.

Example 6.

$$
\begin{array}{l}
p2ac \circ ac2p(a \rightarrow Stop \sqcup b \rightarrow Stop) \\
= \\
\mathbf{RA} \circ \mathbf{A}\left(true \vdash \circledin{}^{y}_{ac'}(y.wait \wedge y.tr = s.tr \wedge a \notin y.ref \wedge b \notin y.ref)\right)
\end{array}
$$

This process corresponds to the application of $p2ac$ to the result obtained in the previous Example 5. In this case, the process is always waiting for the environment and keeps the trace of events unchanged, however it also requires that neither event a nor b are refused. This is a process whose behaviour would not be describable using the standard operators of CSP.

If we consider the result of Lemma 5 in the context of the predicates of our theory, that is, those which are **PBMH**-healthy, then we obtain an inequality as shown in the following Theorem 30.

Theorem 30. *Provided P is* **PBMH***-healthy.* $p2ac \circ ac2p(P) \sqsupseteq P$

Proof.

$$
\begin{array}{lr}
p2ac \circ ac2p(P) & \{\text{Lemma 5}\} \\
= \exists\, ac_0, y \bullet P[ac_0/ac'] \wedge ac_0 \subseteq \{y\} \wedge y \in ac' & \{\text{Predicate calculus}\} \\
\sqsupseteq \exists\, ac_0 \bullet P[ac_0/ac'] \wedge ac_0 \subseteq ac' & \{\text{Definition of } \mathbf{PBMH} \text{ (Lemma 1)}\} \\
= \mathbf{PBMH}(P) & \{\text{Assumption: } P \text{ is } \mathbf{PBMH}\text{-healthy}\} \\
= P & \square
\end{array}
$$

These results establish the existence of a Galois connection [6] between the theories. In particular, these results also hold between the reactive processes, characterised by **R**, and those with angelic nondeterminism characterised by $\mathbf{RA} \circ \mathbf{A}$, that in general, the Galois connection is not restricted to CSP processes.

Using these results, we have established the relationship between operators of CSP and their counterparts in our theory [12]. For instance, in the case of the external choice operator of CSP we have the following results.

Theorem 31. *Provided P and Q are reactive angelic processes.*

$$p2ac(ac2p(P) \,\Box_{\mathcal{CSP}}\, ac2p(Q)) \sqsupseteq P \,\Box\, Q$$

Theorem 32. $ac2p(p2ac(P) \,\Box\, p2ac(Q)) = P \,\Box_{\mathcal{CSP}}\, Q$

These are important in validating our intuitive definitions of the operators using $\circledin^{y}_{ac'}(P)$ and the existing definitions of CSP operators as reactive designs.

4.4 Subset of Non-Angelic Processes

As mentioned before, in the setting of upward-closed binary multirelations, the actual choices available to the angel are those available in every possible demonic choice of the set of final states. This corresponds to the distributed intersection over all possible choices of the set of final states.

Therefore, when we consider the upward-closure of a singleton, that is, a set of final states with only one state, then this must be the only state available for angelic choice. In other words, there is no angelic choice to be made, and the relation can be represented in the original relational model that considers a single final state. This subset of non-angelic processes is characterised in our theory by the following healthiness condition.

Definition 27. $\mathbf{A2}(P) \mathrel{\widehat{=}} \mathbf{PBMH}(P \mathbin{;_{\mathcal{A}}} \{s\} = ac')$

The predicate $P \mathbin{;_{\mathcal{A}}} \{s\} = ac'$ requires the set of final states in P to be either empty or a singleton, otherwise it becomes *false*. Since this purposedly breaks the upward-closure, **PBMH** must be applied as a result. If we consider the application of **A2** to the process $a \rightarrow \mathit{Stop} \sqcup b \rightarrow \mathit{Stop}$, we obtain exactly the same result as in Example 6. In other words, for reactive angelic processes, **A2** characterises exactly the same fixed points as $p2ac \circ ac2p$. We observe, however, that in general, **A2** permits an empty set of final states, whereas in this theory, both **RA1** and $p2ac$ require the set of final states not to be empty. The function **A2** is idempotent and monotonic.

Finally, we establish the following Theorem 33 for reactive angelic processes.

Theorem 33. *Provided P^f_f and P^t_f are* **A2***-healthy.*

$$p2ac \circ ac2p \circ \mathbf{RA} \circ \mathbf{A}(\neg\, P^f_f \vdash P^t_f) = \mathbf{RA} \circ \mathbf{A}(\neg\, P^f_f \vdash P^t_f)$$

That is, when we consider the theory of reactive angelic processes that are **A2**-healthy, then we find that there is a bijection with the original theory of reactive processes. Thus this subset is isomorphic to the theory of CSP.

5 Conclusion

Angelic nondeterminism has traditionally been studied in the refinement calculus [1,2,3] through the universal monotonic predicate transformers. The characterisation of both types of nondeterminism in a relational setting can use

multirelational models [7]. In [8], Rewitzky presents several of these of which the upward-closed model is the most important due to its lattice theoretic structure.

The concept of angelic nondeterminism has also been considered in the context of functional languages by Morris and Tyrrel [14,15], and Hesselink [16] who have modelled both types of nondeterminism at the expression or term level. A generalised algebraic structure has been proposed by Guttmann [17], where existing computational models, such as the monotonic predicate transformers and multirelations, are characterised as instances.

In the context of process algebras such as CSP, however, the notions of angelic nondeterminism considered so far [4,5] have been rather different from that of the refinement calculus. In order to provide a counterpart notion of angelic nondeterminism in CSP, we have developed an encoding of the CSP theory based on the underlying principles of the model of angelic designs previously developed in [9], which itself is an encoding of upward-closed binary multirelations.

The approach we have followed consists of a natural extension of the existing CSP model. We have shown that reactive angelic processes can be specified through angelic designs, in a similar fashion to the CSP theory, where processes can be specified as reactive designs. In addition, we have proposed a natural way to specify CSP operators in the new theory by use of a suitable predicate.

We have established that our theory forms a Galois connection with the CSP theory. Furthermore, when considering the subset of processes that do not exhibit angelic nondeterminism, there is a bijection with the existing CSP theory. A number of operators have also been proved to correspond exactly to their CSP counterparts, thus providing a reassuring result.

Finally, a number of examples have been presented to illustrate its relationship with angelic choice. It remains to be seen what consequences arise from combining angelic choice with other fundamental CSP operators, such as hiding, interleaving and parallel composition. Algebraic laws of the new theory is our main avenue for future work.

References

1. Morris, J.M.: A theoretical basis for stepwise refinement and the programming calculus. Sci. Comput. Program. **9** (December 1987) 287–306
2. Morgan, C.: Programming from specifications. Prentice Hall (1994)
3. Back, R., Wright, J.: Refinement calculus: a systematic introduction. Graduate texts in computer science. Springer (1998)
4. Tyrrell, M., Morris, J., Butterfield, A., Hughes, A.: A Lattice-Theoretic Model for an Algebra of Communicating Sequential Processes. In Barkaoui, K., Cavalcanti, A., Cerone, A., eds.: Theoretical Aspects of Computing - ICTAC 2006. Volume 4281 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2006) 123–137
5. Roscoe, A.W.: Understanding concurrent systems. Springer (2010)
6. Hoare, C.A.R., Jifeng, H.: Unifying Theories of Programming. Prentice Hall International Series in Computer Science (1998)
7. Cavalcanti, A., Woodcock, J., Dunne, S.: Angelic nondeterminism in the unifying theories of programming. Formal Aspects of Computing **18** (2006) 288–307

8. Rewitzky, I.: Binary Multirelations. In de Swart, H., Orlowska, E., Schmidt, G., Roubens, M., eds.: Theory and Applications of Relational Structures as Knowledge Instruments. Volume 2929 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2003) 1964–1964
9. Ribeiro, P., Cavalcanti, A.: Designs with Angelic Nondeterminism. In: Theoretical Aspects of Software Engineering (TASE), 2013 International Symposium on. (2013) 71–78
10. Cavalcanti, A., Woodcock, J.: A Tutorial Introduction to CSP in *Unifying Theories of Programming*. In Cavalcanti, A., Sampaio, A., Woodcock, J., eds.: Refinement Techniques in Software Engineering. Volume 3167 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2006) 220–268
11. Woodcock, J., Cavalcanti, A.: A Tutorial Introduction to Designs in Unifying Theories of Programming. In Boiten, E., Derrick, J., Smith, G., eds.: Integrated Formal Methods. Volume 2999 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2004) 40–66
12. Ribeiro, P.: Reactive angelic processes. Technical report, University of York (February 2014) http://www-users.cs.york.ac.uk/pfr/reports/rac.pdf.
13. Roscoe, A.W.: The Theory and Practice of Concurrency. Prentice Hall (1998)
14. Morris, J.: Augmenting Types with Unbounded Demonic and Angelic Nondeterminacy. In Kozen, D., ed.: Mathematics of Program Construction. Volume 3125 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2004) 274–288
15. Morris, J.M., Tyrrell, M.: Terms with unbounded demonic and angelic nondeterminacy. Science of Computer Programming **65**(2) (2007) 159 – 172
16. Hesselink, W.H.: Alternating states for dual nondeterminism in imperative programming. Theoretical Computer Science **411**(22-24) (2010) 2317 – 2330
17. Guttmann, W.: Algebras for correctness of sequential computations. Science of Computer Programming **85, Part B**(0) (2014) 224 – 240 Special Issue on Mathematics of Program Construction 2012.

A Auxiliary Results

Theorem 34. $\mathbf{PBMH} \circ \mathbf{H1}(P) = \mathbf{H1} \circ \mathbf{PBMH}(P)$

Theorem 35. $\mathbf{PBMH} \circ \mathbf{H2}(P) = \mathbf{H2} \circ \mathbf{PBMH}(P)$

Theorem 36. $\mathbf{RA} \circ \mathbf{A}(P) = \mathbf{RA} \circ \mathbf{PBMH}(P)$

Theorem 37. $ac' \neq \emptyset \land p2ac(\neg\ P^f \vdash P^t) = (\neg\ p2ac(P^f) \vdash p2ac(P^t))$

Lemma 6. $\mathbf{A}(P)_w = \mathbf{A}(P_w)$

Lemma 7. $ac2p \circ \mathbf{PBMH}(P) = ac2p(P)$

Lemma 8. $ac2p(P \vdash Q) = (\neg\ ac2p(\neg\ P) \vdash ac2p(Q))$