Alternating States for Dual Nondeterminism in Imperative Programming

Wim H. Hesselink, June 3, 2009
Dept. of Mathematics and Computing Science, University of Groningen
P.O.Box 407, 9700 AK Groningen, The Netherlands
Email: `w.h.hesselink@rug.nl`, Web: `http://www.cs.rug.nl/~wim`

Abstract. The refinement calculus of Back, Morgan, Morris, and others is based on monotone predicate transformers (weakest preconditions) where conjunctions stand for demonic choices between commands and disjunctions for angelic choices. Arbitrary monotone predicate transformers cannot be modelled by relational semantics but can be modelled by so-called multirelations. Results of Morris indicate, however, that the natural domain for the combination of demonic and angelic choice is the free distributive completion (FDC) of the state space.
The present paper provides a new axiomatization and more explicit construction of the FDC of an arbitrary ordered set. The FDC concept is self-dual, but the construction is not. We therefore determine the duality function from the FDC to the dual of the FDC of the dual ordered set. The elements of the FDC are classified according to their conjunctivity and disjunctivity. The theory is applied to imperative programming with operators for sequential composition and demonic and angelic choice. The theory based on the FDC is shown to be equivalent to a weakest precondition theory for up-closed predicates. If the order is discrete (i.e. the equality relation), the FDC turns out to be the domain of the choice semantics of Back and Von Wright, whereas up-closed multirelations are functions towards this domain.

Keywords: free distributive completion, nondeterminism, angelic choice, demonic choice, refinement calculus, programming semantics

1 Introduction

The refinement calculus of Back, Morgan, Morris, and others is based on monotone predicate transformers where conjunctions stand for demonic choices between commands and disjunctions for angelic choices. The words demonic and angelic describe the opposite faces of the coin of nondeterminism, as explained in Section 1.1. In Section 1.2, we sketch the history of 50 years of nondeterminism, culminating in Morris' proposal of the free distributive completion. Section 1.3 contains an overview of the present paper.

1.1 The two faces of nondeterminism

Nondeterminism is the phenomenon that a command started in a given initial state allows more than one (or possibly less than one) final state. If X is the set of possible initial states and Y is the set of possible final states, the natural way to formalize this is by means of a relation $R \subseteq X \times Y$, where $(x, y) \in R$ means that the command, if started in initial state x, can result in final state y.

Let us now assume that the programmer aims at a certain postcondition for the command. The difference between demonic and angelic choice is the question whether *all* final states y should satisfy the postcondition, or that it suffices that the postcondition holds in *at least one* final state. In the first case, the nondeterminism may be resolved (in a later stage of design) in an arbitrary way; we therefore speak of a demonic choice. In the second case, the choice requires care, and we speak of angelic choice.

Since relation R cannot be used to differentiate between the two interpretations, we turn to the weakest precondition, denoted $wp.c.p$, such that command c is guaranteed to establish postcondition p. The function $wp.c$ that transforms the postcondition into its weakest precondition is a typical predicate transformer.

Restricting for the moment to binary choices, say between commands c and d, we write $c \mathbin{[\!]} d$ and $c \diamondsuit d$ for the demonic and angelic choices between c and d, respectively. Then we have

$$\begin{aligned} wp.(c \mathbin{[\!]} d).p &= wp.c.p \;\wedge\; wp.d.p \;, \\ wp.(c \diamondsuit d).p &= wp.c.p \;\vee\; wp.d.p \;. \end{aligned}$$

This can be understood as follows. In the demonic case, with $[\!]$, it is unknown whether c or d will be executed. Therefore, in order to guarantee postcondition p, we need to require both preconditions. In the angelic case, with $\diamondsuit$, the "angel" can choose c or d, and it is able to establish postcondition p if and only if at least one of the preconditions holds. This shows that demonic choices correspond to conjunctions of predicate transformers, while angelic choices correspond to disjunctions.

Although one can be content with weakest preconditions to describe the nondeterminism, one can also strive for a modelling of the commands as functions with some kind of virtual final states, in the same way as complex numbers were introduced in mathematics to model intermediate results in numerical computations.

In this paper, following [36,37], we show that the latter aim is realizable. The aim is to strengthen the foundations of [36,37], e.g., by providing axioms that characterize the categorical definition of the "free distributive completion".

The alternating states of the title are the virtual states. On the one hand, they are the elements of the free distributive completion of the state space. On the other hand, they can be regarded as Boolean functions on predicates, and they induce the same weakest precondition semantics as before. Once the formalism can handle alternating states, it can easily be extended to handle alternating expressions. We use the term *alternation* to refer to the combination of the two flavours of nondeterminism, following [9]. Allowing nondeterminism via alternating expressions makes it easier to postpone design decisions, and enables us to combine nondeterminism of both flavours with a functional programming style.

1.2 Fifty years of nondeterminism

Nondeterminism was introduced in computer science in 1959 for finite automata by Rabin and Scott [41]. This nondeterminism requires the angelic interpretation, because a nondeterministic finite automaton accepts a string w if and only if it has at least one accepting computation for w, regardless of possibly rejecting or nonterminating computations. In 1962, Chomsky [11] and Schützenberger introduced angelic nondeterminism in pushdown automata to be able to accept arbitrary context-free languages. In 1971, Cook [12] used angelic nondeterministic Turing machines to define the class $\mathcal{NP}$ of nondeterministic polynomial-time complexity.

The idea of program correctness can be traced back to Turing [44], but it became an active research area at the end of the 60s, with [22,30,39]. This was partly inspired by the growing field of parallelism and multiprogramming [16]. In 1975, Dijkstra [17] introduced demonic nondeterminism for his guarded command language, without mentioning that the intention of his nondeterminism differed from the established use in automata theory. Actually, Dijkstra used the term *nondeterminacy* instead of *nondeterminism*, but his adjective was *nondeterministic* and the subtlety was missed by most of his readers.

From 1977 onward, the CIP group used nondeterminism, roughly speaking, for syntax and nondeterminacy for semantics, in either case with the demonic interpretation, e.g., [8]. Indeed, in programming theory, demonic nondeterminism became the rule. This was partly justified by the actual occurrence of demonic nondeterminism in multiprogramming.

Initially, this nondeterminism usually was explicitly bounded, e.g. [2,13,15,18]. In particular, Dijkstra [18, p. 77] mentioned the difficulty of implementing a mechanism to choose within a finite time between infinitely many possibilities. After some years, however, unbounded nondeterminism was proposed and investigated, e.g., in [1,7,8,19,23]. In particular, Park [40] introduced the term *loose* nondeterminism to make it explicit that an implementation is not required to be able to produce all possible results. This idea was essential for the nondeterministic datatypes of [24]. Chandy and Misra [10] exploited demonic nondeterminism mitigated by fairness to eliminate sequential composition in their programming package UNITY for the construction of parallel programs.

The combination of demonic and angelic nondeterminism was first proposed in 1981, for the alternating Turing machines of [9]. Here both forms of nondeterminism were bounded. It seems that the term "angelic non-determinism" first occurs in [31]. Around 1989, angelic nondeterminism entered weakest precondition semantics via e.g. [3,25,34,35]. In programming theory, it is convenient to allow both flavours of nondeterminism to be unbounded. Full treatments of the semantics of unbounded angelic and demonic nondeterminism with recursion (and local variables) were given in [4,27,28].

Angelic nondeterminism is not often used in program correctness, but in [26], we used a weakest angelic precondition to develop a parsing algorithm, because finding a derivation according to a context-free grammar inherently requires angelic choices. In 1995, R.M. Dijkstra [21] used arbitrary monotone predicate transformers with their demonic conjunctions and angelic disjunctions to describe and analyse the semantics of UNITY.

As shown in Section 1.1, relational semantics do not distinguish demonic and angelic choices and therefore do not model arbitrary monotone predicate transformers. In [4,33,42], it was realized that monotone predicate transformers can be modelled by so-called multirelations. This multirelation model is essentially equivalent to one of the constructions of the free distributive completion (FDC), which Morris [36] defines for an arbitrary ordered set, following [43]. Following Morris, we regard the FDC of a state space as the natural domain for the combination of demonic and angelic choices.

1.3 The present paper

This paper is devoted to a deeper study of the FDC and to a more explicit application to imperative programming.

In Section 2, inspired by Morris [36], we present three axioms that characterise the FDC of an arbitrary ordered set. Morris' construction of the FDC is explicit, but requires two steps and a rather extended terminology. We "refactor" his construction, and prove its correctness in such a way that it also shows that our axioms *characterise* the FDC, whereas Morris only proved that the FDC is a *model* of his axioms.

The elements of the FDC are the *alternating states* of the title of this paper. They serve as states with additional information concerning demonic and angelic choices. Our construction of the FDC enables us to identify the alternating states with a kind of monotone predicate transformers. We give a new treatment of the duality of the FDC.

In Section 3, we investigate the junctivity properties of the alternating states, regarded as predicate transformers. In particular, we show that an alternating state is conjunctive (disjunctive) iff it is a conjunction (disjunction) of proper states. We also characterize the finitely conjunctive states.

Section 4 finally presents the alternating states for imperative programming. It shows that the semantics with the alternating states are equivalent to weakest precondition semantics. The junctivity properties of the predicate transformers are those of the alternating states in the image of the function. The well-known duality on predicate transformers turns out to correspond to the natural duality on the alternating states. Commands are then introduced as functions that yield alternating states. Conclusions are drawn in Section 5.

We begin with some preparations. In Section 1.4, we fix the nomenclature for ordered sets and completeness. Section 1.5 contains material on functions, power sets, and order. Complete distributivity is introduced in Section 1.6. Section 1.7 presents order in function spaces and provides the main examples of completely distributive ordered sets.

1.4 Order and completeness

Following [6], we use the term order, ordered set, and monotone where other authors use partial order, partially ordered set (poset), and monotonic.

For an ordered set X, the *dual* X° is defined as the same set with the opposite order $\leq^\circ$ given by $x \leq^\circ x' \equiv x' \leq x$. An ordered set X is called *discrete* if the order is the equality relation, or equivalently if it is equal to its dual as an ordered set.

Recall that an ordered set X is called *complete* if every subset A of X has a least upper bound (join, denoted by $\bigvee A$) and a greatest lower bound (meet, denoted by $\bigwedge A$). The unique greatest element of X is $\top_X = \bigwedge \emptyset$; the unique smallest element is $\bot_X = \bigvee \emptyset$.

1.5 Functions, power sets, and order

Function application is denoted by means of an infix operator "." (dot) that binds stronger than all binary and all prefix operators. It associates to the left, to allow currying.

For any set X, we write $\mathbb{P}.X$ to denote its power set, the set of its subsets ordered by inclusion. For any function $f : X \to Y$, we have the *direct image* function $f^* : \mathbb{P}.X \to \mathbb{P}.Y$ defined by $f^*.A = \{f.x \mid x \in A\} \in \mathbb{P}.Y$ for all $A \in \mathbb{P}.X$.

A function $f : X \to Y$ between ordered sets X and Y is *monotone* if it preserves the order. It is called *an order embedding* if additionally $f.x \leq f.x'$ implies $x \leq x'$ for all x, $x' \in X$. Function f is an order embedding if and only if f is injective and induces an isomorphism of X with the image $f^*.X$ ordered as a subset of Y. This condition implies that f is monotone and injective, but the inverse implication is not valid.

A function $f : X \to Y$ between complete ordered sets is called *disjunctive* if it preserves least upper bounds, i.e. if $x = \bigvee A$ implies $f.x = \bigvee f^*.A$ for all $x \in X$ and $A \subseteq X$. It is called *conjunctive* if it preserves greatest lower bounds. It is called *junctive* if it is both disjunctive and conjunctive [20].

Our application of the terms disjunctivity and conjunctivity rather than join-homomorphism and meet-homomorphism is justified by the fact that our ordered sets are mostly sets of predicates where join and meet coincide with disjunction and conjunction.

1.6 Complete distributivity

A complete ordered set X is called *completely distributive* if it is complete and, for every family of index sets $(i \in I : M.i)$ and families of elements $(j \in M.i : a.j)$ in X, using $R = \prod_{i \in I} M.i$, we have

$$\bigwedge_{i \in I} \bigvee_{j \in M.i} a.j = \bigvee_{r \in R} \bigwedge_{i \in I} a.(r.i) \ .$$

We assume validity of the Axiom of Choice. This is nothing but the axiom that, for every family of nonempty sets $(i \in I : M.i)$, the product set $R = \prod_{i \in I} M.i$ is nonempty. The elements of R are called *choice functions*. The Axiom of Choice thus enables us to combine many choices $r.i \in M.i$ in a single choice r.

This axiom has the consequence that complete distributivity is self-dual: if an ordered set X is completely distributive, its dual X° is also completely distributive. See e.g. [5,29].

The set $\mathbb{B}$ of the Booleans is ordered with *false* $<$ *true*, so that $\leq$ is the same as $\Rightarrow$. The operators $\bigvee$ and $\bigwedge$ correspond to $\exists$ and $\forall$. Complete distributivity of $\mathbb{B}$ boils down to

$$\forall\, i \in I : \exists\, j \in M.i : a.j \quad \equiv \quad \exists\, r \in \textstyle\prod_{i \in I} M.i : \forall\, i \in I : a.(r.i) \ .$$

This is nothing but an unusual form of the Axiom of Choice. In other words, by postulating the Axiom of Choice, we postulated complete distributivity of $\mathbb{B}$.

1.7 Order in function spaces

If X is a set and Z is an ordered set, the set of functions $(X \to Z)$ is regarded as an ordered set with the argumentwise ordering: $f \leq g \equiv (\forall\, x \in X : f.x \leq g.x)$. If Z is complete, $(X \to Z)$ is complete and the least upper bounds and greatest lower bounds in $(X \to Z)$ can be obtained argumentwise. If Z is completely distributive, then so is $(X \to Z)$.

If X and Z are ordered sets, the set of the monotone functions from X to Z is denoted by $(X \xrightarrow{m} Z)$. It is easy to see that least upper bounds and greatest lower bounds of monotone functions are monotone. Therefore, if Z is complete (or completely distributive), the subset $(X \xrightarrow{m} Z)$ of $(X \to Z)$ is by itself complete (completely distributive, respectively), and the injection of $(X \xrightarrow{m} Z)$ into $(X \to Z)$ is junctive.

If X is a set, its power set $\mathbb{P}.X$ is identified with ordered set $(X \to \mathbb{B})$ via the rule $p.x \equiv (x \in p)$. Because $\mathbb{B}$ is completely distributive, $\mathbb{P}.X$ is also completely distributive.

For an ordered set X, a function $p \in \mathbb{P}.X$ is monotone if and only if, as a subset, it is *up-closed* $(x \leq x' \ \wedge \ x \in p \ \Rightarrow \ x' \in p)$. We therefore define $\mathbb{U}.X = (X \xrightarrow{m} \mathbb{B})$ to be the set of the up-closed subsets or monotone elements of $\mathbb{P}.X$. For any $a \in X$, we define $up.a = \{x \in X \mid a \leq x\} \in \mathbb{U}.X$. For every ordered set X, the ordered set $\mathbb{U}.X$ is completely distributive. Note that the function $up : X \to \mathbb{U}.X$ is antimonotone and, in general, not monotone. This is the reason that Morris [36, 4.2] gives his set $\mathcal{U}X$ the opposite order, so that $\mathcal{U}X = (\mathbb{U}.X)^\circ$.

2 Free Distributive Completions of Ordered Sets

A *free distributive completion* (FDC) of an ordered set X is defined to be a monotone function f from X to a completely distributive set Y such that, for every monotone function g from X to a completely distributive ordered set Z, there is a unique junctive function $h : Y \to Z$ with $g = h \circ f$ [36,43].

We present an internal characterization of an FDC of X in Section 2.1. In Section 2.2, we prove Tunnicliffe's result [43] that every ordered set has an FDC, which is essentially unique. We compare our presentation with the work of Morris and Tyrrell in Section 2.3. In Section 2.4, we use that every monotone function on X has a unique junctive extension to the FDC of X. This extension or lifting plays a crucial role later on. In Section 2.5, we show self-duality of the FDC concept: the dual of an FDC of X is an FDC of the dual of X.

2.1 Axioms for an FDC

We start with an axiomatic characterization of an FDC Y of an ordered set X in terms of the ordered structures of X and Y, and their relationships. For the sake of the argument, we define a function $f : X \to Y$ with Y completely distributive to be an *fd-completion* if it satisfies the conditions

(fd0) $\quad y = \bigvee\{\bigwedge f^*.A \mid A : \bigwedge f^*.A \le y\}$ for all $y \in Y$,
(fd1) $\quad \bigwedge f^*.A \le f.x \quad \equiv \quad (\exists\, a \in A : a \le x)$ for all $x \in X$, $A \in \mathbb{P}.X$,
(fd2) $\quad \bigwedge f^*.A \le \bigvee B \quad \equiv \quad (\exists\, y \in B : \bigwedge f^*.A \le y)$ for all $A \in \mathbb{P}.X$, $B \in \mathbb{P}.Y$.

Condition (fd0) expresses a kind of density. The inequality $\bigvee\{\bigwedge f^*.A \mid A : \bigwedge f^*.A \le y\} \le y$ holds trivially. Therefore (fd0) just says that every y can be approximated from below by terms of the form $\bigwedge f^*.A$.

Condition (fd1) is the only one that mentions the order of the set X (in the right-hand side). The other two conditions only mention the order of Y. By taking $A := \{x'\}$ with $x' \in X$, we see that condition (fd1) implies that function f is an order embedding and, hence, monotone and injective. Taking for A the empty set, we see that $f.x \ne \top_Y$ for all $x \in X$, so that f is not surjective.

Condition (fd2) expresses a kind of atomicity of the terms $\bigwedge f^*.A$. Taking $B := \emptyset$ and $A = \{x\}$, it implies that $f.x \ne \bot_Y$ for all $x \in X$.

Theorem 1. *Let X be an ordered set and Y a completely distributive ordered set. A function $f : X \to Y$ is a free distributive completion if and only if it is an fd-completion.*

Proof. The proof of "only if" is postponed to the end of Section 2.2.

Here, we only give the proof of "if". Let $f : X \to Y$ be an fd-completion. In order to prove that f is a free distributive completion, we need to factor an arbitrary monotone function to a completely distributive ordered set Z over f and a junctive function $h : Y \to Z$. Let $g : X \to Z$ be a monotone function to a completely distributive ordered set Z. If $h : Y \to Z$ is junctive with $g = h \circ f$, then condition (fd0) with junctivity of h and $g = h \circ f$ implies that, for all y,

(0) $\quad h.y = \bigvee\{\bigwedge g^*.A \mid A : \bigwedge f^*.A \le y\}$.

This proves uniqueness of h. In order to prove existence, we therefore define function $h : Y \to Z$ by formula (0). It remains to prove that $h \circ f = g$ and that function h is junctive.

The equality $h \circ f = g$ is proved by the observation that, for every x,

$$
\begin{array}{ll}
 & h.(f.x) \\
= & \{\ (0)\ \} \\
 & \bigvee\{\bigwedge g^*.A \mid A : \bigwedge f^*.A \le f.x\} \\
= & \{\ (\mathrm{fd1})\} \\
 & \bigvee\{\bigwedge g^*.A \mid A : (\exists\, a \in A : a \le x)\} \\
= & \{\ \le g.x \text{ because } g \text{ is monotone, } \ge g.x \text{ from } A := \{x\}\ \} \\
 & g.x\ .
\end{array}
$$

For the verification that function h is junctive, we let B be a subset of Y. Disjunctivity of h is proved in

$$\begin{array}{ll} & h.(\bigvee B) \\ = & \{\ (0)\ \} \\ & \bigvee\{\bigwedge g^*.A \mid A : \bigwedge f^*.A \leq \bigvee B\} \\ = & \{\ (\text{fd2})\ \} \\ & \bigvee\{\bigwedge g^*.A \mid A : (\exists\, y \in B : \bigwedge f^*.A \leq y)\} \\ = & \{\ \text{splitting joins}\ \} \\ & \bigvee_{y\in B} \bigvee\{\bigwedge g^*.A \mid A : \bigwedge f^*.A \leq y\} \\ = & \{\ (0)\ \} \\ & \bigvee_{y\in B} h.y\ . \end{array}$$

Since h is disjunctive, it is monotone and hence satisfies $h.(\bigwedge_{y\in B} y) \leq \bigwedge_{y\in B} h.y$. The other inequality for conjunctivity is proved in

$$\begin{array}{ll} & \bigwedge_{y\in B} h.y \\ = & \{\ (0)\ \} \\ & \bigwedge_{y\in B} \bigvee\{\bigwedge g^*.A \mid A : \bigwedge f^*.A \leq y\} \\ = & \{\ Z \text{ is completely distributive;} \\ & \quad \text{take } R = \prod_{y\in B}\{A \mid \bigwedge f^*.A \leq y\}\ \} \\ & \bigvee_{r\in R} \bigwedge_{y\in B} \bigwedge g^*.(r.y) \\ = & \{\ \text{definitions of } \bigwedge \text{ and } g^*\text{; take } A_r := \bigcup_{y\in B} r.y\ \} \\ & \bigvee_{r\in R} \bigwedge g^*.A_r \\ \leq & \{\ (*)\text{, see below}\ \} \\ & \bigvee\{\bigwedge g^*.A \mid A : \bigwedge f^*.A \leq \bigwedge_{y\in B} y\} \\ = & \{\ (0)\ \} \\ & h.(\bigwedge_{y\in B} y)\ . \end{array}$$

Step (*) is justified by the observation for every $r \in R$, that $\bigwedge f^*.(r.y) \leq y$ for all $y \in B$, and that therefore $A_r = \bigcup_{y\in B} r.y$ satisfies $\bigwedge f^*.A_r \leq \bigwedge_{y\in B} y$.

This concludes the proof of conjunctivity of h, and thus of junctivity of h. □

Remark. Once condition (fd0) has been postulated, the need for conditions like (fd1) and (fd2) emerges naturally in the proof.

2.2 Construction of an FDC

Given an ordered set X, we now construct an fd-completion $f : X \to Y$.

The heuristical starting point is (fd0), which says that every element $y \in Y$ is the least upper bound of the elements $\bigwedge f^*.A$ that are below y, with A ranging over the subsets of X. If a subset A is replaced by its up-closure $\{x \mid \exists\, a \in A : a \leq x\}$, the greatest lower bound $\bigwedge f^*.A$ is unchanged when f is monotone. Therefore (fd0) should remain valid when A only ranges over the set $\mathbb{U}.X$; see Section 1.7. This implies $y = \bigvee\{\bigwedge f^*.A \mid A \in R(y)\}$ for $R(y) = \{A \in \mathbb{U}.X \mid \bigwedge f^*.A \leq y\}$. We notice now that $R(y)$ determines the element y, and that it is an up-closed subset of $\mathbb{U}.X$, that is $R(y) \in \mathbb{U}.(\mathbb{U}.X)$. We therefore try and construct the ordered set Y as $\mathbb{U}.(\mathbb{U}.X)$. Indeed, the ordered set $\mathbb{U}.(\mathbb{U}.X)$ is completely distributive. Recall from 1.7, that in the set $\mathbb{U}.(\mathbb{U}.X)$ we can use $\bigcup$ and $\bigcap$ for $\bigvee$ and $\bigwedge$.

Let $j : X \to \mathbb{U}.(\mathbb{U}.X)$ be defined by $j.x = \{K \in \mathbb{U}.X \mid x \in K\}$. Indeed, for any $x \in X$, we have $j.x \in \mathbb{U}.(\mathbb{U}.X)$ because $K \subseteq K'$ and $K \in j.x$ implies $K' \in j.x$.

Theorem 2. *For every ordered set X, the function $j : X \to \mathbb{U}.(\mathbb{U}.X)$ is an fd-completion and, hence, also a free distributive completion of X.*

Proof. In this proof, we let K (and K') always range over $\mathbb{U}.X$. Function j is monotone because for all x, $x' \in X$,

$$\begin{array}{ll} & x \le x' \\ \Rightarrow & (\forall\, K : x \in K \;\Rightarrow\; x' \in K) \\ \equiv & j.x \subseteq j.x' \;. \end{array}$$

We next observe, for any $A \in \mathbb{P}.X$, that

$$\bigcap j^*.A = \{K \mid A \subseteq K\} \;, \tag{1}$$

because of

$$\begin{array}{ll} & K \in \bigcap j^*.A \\ \equiv & \forall\, x \in A : K \in j.x \\ \equiv & \forall\, x \in A : x \in K \\ \equiv & A \subseteq K \;. \end{array}$$

In order to verify (fd0): $y = \bigcup\{\bigcap j^*.A \mid A : \bigcap j^*.A \subseteq y\}$, we first note that, when A is replaced by its up-closure, the set $\bigcap j^*.A = \{K \mid A \subseteq K\}$ does not change. We can therefore change (fd0) by letting A range over the up-closed subsets of X. For up-closed A, formula (1) implies that $(\bigcap j^*.A \subseteq y) \equiv (A \in y)$. Formula (fd0) therefore follows from

$$y = \bigcup\{\bigcap j^*.A \mid A \in y\} \;, \tag{2}$$

which is proved in

$$\begin{array}{ll} & K \in \bigcup\{\bigcap j^*.A \mid A \in y\} \\ \equiv & \{ \text{ definition union and (1) } \} \\ & \exists\, A \in y : A \subseteq K \\ \equiv & \{ \; y \text{ up-closed } \} \\ & K \in y \;. \end{array}$$

We verify (fd1) by first observing for $x \in X$ and $A \in \mathbb{P}.X$ that

$$\bigcap j^*.A \subseteq j.x \quad \Leftarrow \quad (\exists\, a \in A : a \le x) \;,$$

because j is monotone. The other implication of (fd1) is proved in

$$\begin{array}{ll} & \bigcap j^*.A \subseteq j.x \quad \Rightarrow \quad (\exists\, a \in A : a \le x) \\ \equiv & \{ \text{ (1) and definition of } j \text{ } \} \\ & (\forall\, K : A \subseteq K \;\Rightarrow\; x \in K) \quad \Rightarrow \quad (\exists\, a \in A : a \le x) \\ \equiv & \{ \text{ contraposition } \} \\ & (\forall\, a \in A : a \not\le x) \quad \Rightarrow \quad (\exists\, K : A \subseteq K \;\wedge\; x \notin K) \\ \equiv & \{ \text{ take } K_0 := \{z \in X \mid z \not\le x\} \in \mathbb{U}.X \text{ } \} \\ & \mathit{true} \;. \end{array}$$

We verify condition (fd2) by first observing that the existence of $y \in B$ with $\bigcap j^*.A \subseteq y$ clearly implies $\bigcap j^*.A \subseteq \bigcup B$. The other implication is proved in

$$\begin{array}{ll} & \bigcap j^*.A \subseteq \bigcup B \\ \equiv & \{ \text{ (1) } \} \\ & \forall\, K : A \subseteq K \;\Rightarrow\; K \in \bigcup B \\ \Rightarrow & \{ \text{ take } K_1 := \{x \mid \exists\, a \in A : a \le x\}, \text{ then } K_1 \in \mathbb{U}.X \text{ } \} \\ & \exists\, y \in B : K_1 \in y \\ \equiv & \{ \; A \subseteq K \;\Rightarrow\; K_1 \subseteq K \text{ for all } K, \text{ and } y \in \mathbb{U}.(\mathbb{U}.X) \text{ } \} \\ & \exists\, y \in B : \forall\, K : A \subseteq K \;\Rightarrow\; K \in y \\ \equiv & \{ \text{ (1) } \} \\ & \exists\, y \in B : \bigcap j^*.A \subseteq y \;. \end{array}$$

This proves that f is an fd-completion. As proved in the above partial proof of Theorem 1, this implies that f is a free distributive completion. $\square$

For any ordered set X, we write $\mathbb{F}.X = \mathbb{U}.(\mathbb{U}.X)$ and $j : X \to \mathbb{F}.X$ for the free distributive completion thus constructed. If necessary, we give j an index X to indicate that it depends on X. One referee proposed to write $\mathbb{U}^2$ instead of $\mathbb{F}$. We prefer to use $\mathbb{F}$, because we regard $\mathbb{F} = \mathbb{U}^2$ as only one of several alternative constructions of the free distributive completion. Indeed, the paper [38, Section 4.5] following [43] contains two alternatives.

Because of (fd1), function $j : X \to \mathbb{F}.X$ is an order embedding. It is never surjective, because both the bottom element $\perp$ and the top element $\top$ of $\mathbb{F}.X$ are not in the image of j.

Example. Assume that X is finite and linear. Then $\mathbb{U}.X$ consists of the sets $up.x$ for $x \in X$ and the empty set. Therefore, $\mathbb{U}.X$ is also finite and linear. Therefore $\mathbb{F}.X$ is also finite and linear. It consists of the elements $\perp$, $\top$, and $j.x$ with $x \in X$.

By a standard argument in category theory, the free distributive completion is essentially unique:

Theorem 3. *Let $f : X \to Y$ be a free distributive completion of an ordered set X. Then there is a unique isomorphism $g : Y \cong \mathbb{F}.X$ with $j = g \circ f$.*

Proof. For convenience, we do provide the standard argument. Since $j : X \to \mathbb{F}.X$ is monotone and $f : X \to Y$ is an FDC, there is a unique junctive function $g : Y \to \mathbb{F}.X$ with $j = g \circ f$. Since f is monotone and j is an FDC, there is a unique junctive function $g' : \mathbb{F}.X \to Y$ with $f = g' \circ j$. The composition $g' \circ g : Y \to Y$ is junctive and satisfies $g' \circ g \circ f = f$, just as the identity function of Y does. Since f is an FDC, this implies that $g' \circ g$ is the identity function of Y. By symmetry, $g \circ g'$ is the identity function of $\mathbb{F}.X$. Therefore, g and g' are inverse to each other. So they are isomorphisms. □

Proof of "only if" for Theorem 1. Let $f : X \to Y$ be a free distributive completion. By Theorem 2, $j : X \to \mathbb{F}.X$ is an fd-completion. By Theorem 3, there is an isomorphism $g : Y \cong \mathbb{F}.X$ with $j = g \circ f$. One can now use the isomorphim g (and its inverse g') to show that $f : X \to Y$ is an fd-completion as well. □

2.3 Comparison with the presentation of Morris and Tyrrell

Morris and Tyrrell [36,37,38] proposed the use of Tunnicliffe's free distributive completion [43] to model dual nondeterminacy. The construction of the FDC in [36,38] is a two-step procedure following [43].

Our conditions (fd0), (fd1), (fd2) were inspired by the proof theory of [36] and [38, Section 3]. These papers contain condition (fd2) more or less literally, and condition (fd0) in the form:

$$y \le y' \quad \equiv \quad (\forall\, A : \bigwedge f^*.A \le y \;\Rightarrow\; \bigwedge f^*.A \le y')\ .$$

They do not mention condition (fd1). Yet, this condition cannot be omitted. Indeed, let $f : X' \to X$ be an arbitrary surjective function. The composition $g = j \circ f : X' \to \mathbb{F}.X$ satisfies the conditions (fd0) and (fd2) because of $\{\bigwedge g^*.B \mid B \in \mathbb{P}.X'\} = \{\bigwedge j^*.A \mid A \in \mathbb{P}.X\}$. Yet g is a free distributive completion only if $f : X' \to X$ is an isomorphism of ordered sets.

Our characterizing conditions (fd0), (fd1), (fd2) enable us to simplify the construction of the FDC, by combining it with the proofs of soundness and completeness for these conditions.

In our view, the papers [36,37,38] suffer from the choice to treat X as a subset of $\mathbb{F}.X$, so that the elements of X and $\mathbb{F}.X$ need to be distinguished as "proper" and "improper", while the orders of X and $\mathbb{F}.X$ are distinguished as $\le$ and $\sqsubseteq$.

2.4 Lifting to the FDC

The universal property in the definition of the FDC immediately induces the following definition of lifting. For any monotone function $f : X \to Z$ where Z is completely distributive, the *lifting* $\varphi.f$ of f is defined as the unique junctive function $\varphi.f : \mathbb{F}.X \to Z$ with $f = \varphi.f \circ j$.

Theorem 4. *Let X, Y, and Z be ordered sets, and let Z be completely distributive. Let $f : X \to \mathbb{F}.Y$ and $g : Y \to Z$ be monotone. Then $\varphi.f$ and $\varphi.g$ are well-defined and $\varphi.g \circ \varphi.f = \varphi.(\varphi.g \circ f)$.*

Proof. First, note that $\mathbb{F}.Y$ and Z are completely distributive. Therefore, $\varphi.f : \mathbb{F}.X \to \mathbb{F}.Y$ and $\varphi.g : \mathbb{F}.Y \to Z$ are well-defined. We have $\varphi.g \circ f : X \to Z$. Therefore, $\varphi.(\varphi.g \circ f)$ is the unique junctive function $\mathbb{F}.X \to Z$ with $\varphi.g \circ f = \varphi.(\varphi.g \circ f) \circ j_X$. So, it suffices to observe that $\varphi.g \circ \varphi.f$ is junctive and satisfies $\varphi.g \circ f = \varphi.g \circ \varphi.f \circ j_X$. □

While the definition of liftings is based on the universal property in the definition of the FDC, we can also use the construction $\mathbb{F}.X = \mathbb{U}.(\mathbb{U}.X)$ to interpret elements $y \in \mathbb{F}.X$ as monotone functions $y : \mathbb{U}.X \xrightarrow{m} \mathbb{B}$, or up-closed subsets $y \subseteq \mathbb{U}.X$. Indeed, since $\varphi.f$ is junctive and satisfies $\varphi.f \circ j = f$, formula (2) implies, for all $y \in \mathbb{F}.X$, that

$$\varphi.f.y = \bigvee\{\bigwedge f^*.K \mid K \in y\} \ . \tag{3}$$

2.5 Duality

Recall that X° is the ordered set X with the opposite order. If X is completely distributive then so is X°. Any monotone function $f : X \to Y$ is also a monotone function $f : X^\circ \to Y^\circ$. The function $f : X^\circ \to Y^\circ$ is conjunctive if and only if $f : X \to Y$ is disjunctive; and conversely. Therefore $f : X^\circ \to Y^\circ$ is junctive if and only if $f : X \to Y$ is junctive. It follows that the free distributive completion is self-dual:

Theorem 5. *Let $f : X \to Y$ be a free distributive completion. Then $f : X^\circ \to Y^\circ$ is a free distributive completion.*

Proof. Let $g : X^\circ \to Z$ be a monotone function to a completely distributive Z. Then $g : X \to Z^\circ$ is monotone and Z° is completely distributive. Because $f : X \to Y$ is a free distributive completion, it follows that there is a unique junctive function $h : Y \to Z^\circ$ with $g = h \circ f$. Consequently, $h : Y^\circ \to Z$ is junctive with $g = h \circ f$, and it is the only junctive function with this property. □

The combination of Theorems 1 and 5 yields:

Corollary 1. *Let $f : X \to Y$ be a free distributive completion of an ordered set X. Then $f : X^\circ \to Y^\circ$ is an fd-completion, i.e, function f satisfies the duals of the conditions (fd0), (fd1), (fd2).*

Another consequence of Theorem 5 is that there is a canonical isomophism $\psi : \mathbb{F}.X \to (\mathbb{F}.X^\circ)^\circ$. Using $j : X \to \mathbb{F}.X$ and $j_\circ : X^\circ \to \mathbb{F}.X^\circ$, we have $\psi = \varphi.j_\circ$. Therefore, by (3), we have

$$\psi.y = \bigvee\{\bigwedge j_\circ^*.K \mid K \in y\} \ .$$

In this formula, the conjunction and disjunction are those of $(\mathbb{F}.X^\circ)^\circ$. Using a set-theoretical interpretation of $\mathbb{F}.X^\circ$, we therefore get:

$$\psi.y = \bigcap\{\bigcup j_\circ^*.K \mid K \in y\} \ .$$

We have $j_\circ(x) = \{L \in \mathbb{U}.X^\circ \mid x \in L\}$. It follows that, for any $L \in \mathbb{U}.X^\circ$,

$$\begin{array}{ll} & L \in \bigcup j_\circ^*.K \\ \equiv & \exists\, x \in K : x \in L \\ \equiv & L \cap K \neq \emptyset \ . \end{array}$$

We finally observe

$$\begin{array}{ll} & L \in \psi.y \\ \equiv & \{ \text{ above results } \} \\ & \forall\, K \in y : L \cap K \neq \emptyset \\ \equiv & \{ \text{ calculus } \} \\ & \forall\, K : K \subseteq X \setminus L \ \Rightarrow \ K \notin y \\ \equiv & \{ \ X \setminus L \in \mathbb{U}.X;\ y \text{ is up-closed } \} \\ & X \setminus L \notin y \ . \end{array}$$

This proves that the duality isomorphism $\psi : \mathbb{F}.X \to (\mathbb{F}.X^\circ)^\circ$ satisfies

(4) $$\psi.y = \{L \in \mathbb{U}.X^\circ \mid X \setminus L \notin y\} \ .$$

3 Alternating States

The elements of $\mathbb{F}.X$ are called *alternating states* in view of [9], or *states* (for brevity). The elements of $\mathbb{F}.X$ of the form $j.x$ for some $x \in X$ are called *proper states.*

In this section we concentrate on the properties of alternating states that reflect the particular construction of $\mathbb{F}.X$ chosen in Section 2.2. We therefore use the operators $\bigcap$ and $\bigcup$ rather than $\bigwedge$ and $\bigvee$. In Section 3.1, we resolve a potential ambiguity in the interpretation of states as functions. In 3.2, we determine necessary and sufficient conditions for a state to be conjunctive, disjunctive, or junctive. Section 3.3 is devoted to finitely conjunctive states.

3.1 The functional interpretation of states

An alternating state $y \in \mathbb{F}.X = \mathbb{U}.(\mathbb{U}.X)$ is a monotone Boolean function that can be applied to elements $p \in \mathbb{U}.X$. On the other hand, an element $p \in \mathbb{U}.X$ is a monotone function $X \xrightarrow{m} \mathbb{B}$ and has therefore a lifting $\varphi.p : \mathbb{F}.X \to \mathbb{B}$, which can be applied to alternating states $y \in \mathbb{F}.X$. The two interpretations coincide since, for every $p \in \mathbb{U}.X$ and $y \in \mathbb{F}.X$, we have

(5) $$\begin{array}{ll} & \varphi.p.y \\ \equiv & \{ \ (3) \ \} \\ & \bigvee\{\bigwedge p^*.K \mid K \in y\} \\ \equiv & \{ \ \exists \text{ and } \forall \text{ are } \bigvee \text{ and } \bigwedge \text{ for } \mathbb{B}, \text{ definition } p^* \ \} \\ & \exists\, K \in y : \forall\, x \in K : p.x \\ \equiv & \{ \text{ regard } p \text{ as a subset } \} \\ & \exists\, K \in y : K \subseteq p \\ \equiv & \{ \ y \text{ is up-closed } \} \\ & p \in y \ . \end{array}$$

3.2 Junctivity of alternating states

An alternating state $y \in \mathbb{F}.X$ is an element of $\mathbb{U}.X \xrightarrow{m} \mathbb{B}$. We can therefore investigate its conjunctivity and disjunctivity.

Theorem 6. *Let $y \in \mathbb{F}.X$ be regarded as a function $y : \mathbb{U}.X \xrightarrow{m} \mathbb{B}$.*
(a) Function y is conjunctive if and only if $y = \{K \mid A \subseteq K\}$ for some $A \in \mathbb{P}.X$.
(b) Function y is disjunctive if and only if $y = \{K \mid K \cap B \neq \emptyset\}$ for some $B \in \mathbb{P}.X$.
(c) Function y is junctive if and only if state y is proper.

Proof. (a) If y is conjunctive, the set $A = \bigcap\{K \mid K \in y\}$ satisfies $y.A = (\forall\ K \in y : y.K) = true$, so that $A \in y$ and hence $\{K \mid A \subseteq K\} = y$. Conversely, if $y = \{K \mid A \subseteq K\}$ for some $A \in \mathbb{U}.X$, then $(\bigcap_i K.i \in y) \equiv (\forall\ i : K.i \in y)$ for any family $(i : K.i)$.

(b) If y is disjunctive, define $B = \{x \in X \mid up.x \in y\}$. Every $K \in \mathbb{U}.X$ satisfies $K = \bigcup\{up.x \mid x \in K\}$, so that $y.K = (\exists\ x \in K : up.x \in y) = (K \cap B \neq \emptyset)$, so that $y = \{K \mid K \cap B \neq \emptyset\}$. Conversely, if $y = \{K \mid K \cap B \neq \emptyset\}$ for some $B \in \mathbb{U}.X$, then $(\bigcup_i K.i \in y) \equiv (\exists\ i : K.i \in y)$ for any family $(i : K.i)$.

(c) By (a) and (b), we have A and B with $y = \{K \mid A \subseteq K\} = \{K \mid K \cap B \neq \emptyset\}$. Put $A' = \{x \mid \exists\ a \in A : a \leq x\}$. Then $A \subseteq A'$ and hence $A' \in y$. Therefore, there is some $x \in A' \cap B$. Since $x \in B$, we have $j.x \subseteq y$. Since $x \in A'$, we have $y \subseteq j.x$. This proves $y = j.x$, so that y is proper. Conversely, if $y = j.x$, part (a) with $A := up.x$ and part (b) with $B := \{x\}$ imply that y is junctive. $\Box$

According to (1), we have $\{K \mid A \subseteq K\} = \bigcap j^*.A$. Therefore, an alternating state is conjunctive if and only if it is a meet of proper states. It can also be proved that $\{K \mid K \cap B \neq \emptyset\} = \bigcup j^*.B$. Therefore, an alternating state is disjunctive if and only if it is a join of proper states.

3.3 Finitely conjunctive alternating states

An state $y \in \mathbb{F}.X$ is called *finitely conjunctive* if $y.(\bigcap S) = \bigcap\{y.K \mid K \in S\}$ for every finite set S of elements of $\mathbb{U}.X$. Since y is monotone, i.e., an up-closed subset of $\mathbb{U}.X$, finite conjunctivity of y is equivalent to the conditions that $X \in y$ and that $K \cap K' \in y$ for all K and $K' \in y$. Indeed, the first condition comes from empty S, the second condition applies to pairs. The equality for bigger finite sets is proved by mathematical induction.

Recall that an ordered set B is called *directed* iff it is nonempty and, for every y, $y' \in B$, there is some $y'' \in B$ with $y \leq y''$ and $y' \leq y''$. We now show that finite conjunctivity is preserved under directed unions and that every finitely conjunctive element is a directed union of conjunctive ones.

Theorem 7. *(a) Let B be a directed subset of $\mathbb{F}.X$ and assume that all $y \in B$ are finitely conjunctive. Then $\bigcup B$ is finitely conjunctive.*
(b) Let $y \in \mathbb{F}.X$ be finitely conjunctive. Then there is a directed set B of conjunctive elements of $\mathbb{F}.X$ with $y = \bigcup B$.

Proof. (a) Firstly, $X \in \bigcup B$ since $X \in y \in B$ for some y, because B is nonempty and every $y \in B$ is finitely conjunctive. Next, assume K and $K' \in \bigcup B$. Then there are y and $y' \in B$ with $K \in y$ and $K' \in y'$. There is some $y'' \in B$ with $y \subseteq y''$ and $y' \subseteq y''$. Then both K and K' are in y''. Since y'' is finitely conjunctive, this implies $K \cap K' \in y'' \subseteq \bigcup B$.

(b) Define $r.A = \{K \in \mathbb{U}.X \mid A \subseteq K\}$. Then $y = \bigcup_{A \in y} r.A$ and each $r.A$ is conjunctive by Theorem 6(a). If A, $B \in y$, then $r.A \cup r.B \subseteq r.(A \cap B)$. Therefore, if y is finitely conjunctive, the set $\{r.A \mid A \in y\}$ is directed. $\Box$

Example. The prototypical example is $X = \mathbb{N}$ with the discrete ordering. Let $f : \mathbb{N} \to \mathbb{F}.X$ be given by $f.n = \{K \mid [n, \infty) \subseteq K\}$. Then all states $f.n$ are conjunctive by Theorem 6. For $k \leq n$, we have $f.k \subseteq f.n$. Therefore, the family $(n \in \mathbb{N} : f.n)$ is directed, so that Theorem 7 implies that the alternating state $y = \bigcup_n f.n$ is finitely conjunctive. The set y is the set of the cofinite subsets of $\mathbb{N}$ (recall that $U \subseteq \mathbb{N}$ is called *cofinite* iff its complement $\mathbb{N} \setminus U$ is finite). It is easy to see that, indeed, this set y is finitely conjunctive and not conjunctive. In this case, $\bigcap y = \emptyset$. □

4 Alternation in an Imperative Setting

We now apply the theory to imperative programming with operators for sequential composition and demonic and angelic choice. Modifiers to specify state changes are introduced in Section 4.1. In Section 4.2, we give the weakest precondition semantics of modifiers, which turns out to be an order isomorphism that also preserves sequential composition. In Section 4.3, we show that the duality of predicate transformers corresponds to the duality of the FDC of Section 2.5. Section 4.4 contains a proposal for a syntax for possibly recursive commands with semantics in an ordered set X. Finally, in Section 4.5, we come down to imperative programming on an unordered state space.

At this point, it is convenient to abstract from the specific representation of $\mathbb{F}.X$ as a set of sets ordered by inclusion. We therefore use the ordinary lattice operations $\wedge$, $\bigwedge$, $\vee$, $\bigvee$ rather than $\cap$, $\bigcap$, $\cup$, $\bigcup$. The smallest and largest states are denoted $\bot$, also called *abort*, and $\top$, also called *miracle*, respectively.

4.1 Modifiers

We take *modifiers* as the generic term to specify steps that transform states from a state space X into alternating states from a state space Y. The specification may allow the steps a certain freedom (demonic choice) or may reckon with future user requirements (angelic choice). The idea goes back to e.g. [3,35], and has been exposed in the text book [4], which uses the term *contract* instead of *modifier*.

We define a *modifier* from state space X to state space Y to be a monotone function $X \to \mathbb{F}.Y$. The set of the modifiers from X to Y is thus the ordered set of the functions $(X \xrightarrow{m} \mathbb{F}.Y)$. Since $\mathbb{F}.Y$ is completely distributive, $(X \xrightarrow{m} \mathbb{F}.Y)$ with the induced order is also completely distributive.

Sequential composition of modifiers $c : X \xrightarrow{m} \mathbb{F}.Y$ and $d : Y \xrightarrow{m} \mathbb{F}.Z$ is defined by $c; d = \varphi.d \circ c : X \xrightarrow{m} \mathbb{F}.Z$. Using that $\varphi.j_X$ is the identity function of $\mathbb{F}.X$, and similarly for Y, one can easily verify that the modifiers j_X form unit elements for this operation in the sense that $j_Y; d = d$, and $c; j_X = c$. Sequential composition is associative because for any $c : X \xrightarrow{m} \mathbb{F}.Y$, $d : Y \xrightarrow{m} \mathbb{F}.Z$, and $e : Z \xrightarrow{m} \mathbb{F}.W$,

$$
\begin{array}{ll}
& (c; d); e = c; (d; e) \\
\equiv & \{ \text{ definition of sequential composition } \} \\
& \varphi.e \circ (\varphi.d \circ c) = \varphi.(\varphi.e \circ d) \circ c \\
\equiv & \{ \text{ associativity of } \circ \text{ and Theorem 4 } \} \\
& \textit{true} \ .
\end{array}
$$

In this way, the modifiers form a category with state spaces as objects and modifiers as morphisms. Note that function φ gives an order isomorphism from the set of the modifiers $(X \xrightarrow{m} \mathbb{F}.Y)$ to the set of the junctive functions $\mathbb{F}.X \to \mathbb{F}.Y$. Therefore, alternatively, modifiers could have been defined as the elements of the latter set.

4.2 Weakest precondition semantics

We now take the step to predicate transformers and weakest preconditions and show that the ordered set of the modifiers from X to Y is isomorphic to the ordered set of the monotone predicate transformers from $\mathbb{U}.Y$ to $\mathbb{U}.X$.

Indeed, for arbitrary ordered sets X and Y, the ordered set of modifiers from X to Y satisfies

$$\begin{aligned}&(X \xrightarrow{m} \mathbb{F}.Y) = (X \xrightarrow{m} \mathbb{U}.(\mathbb{U}.Y)) = (X \xrightarrow{m} (\mathbb{U}.Y \xrightarrow{m} \mathbb{B}))\\ &\cong (\mathbb{U}.Y \xrightarrow{m} (X \xrightarrow{m} \mathbb{B})) = (\mathbb{U}.Y \xrightarrow{m} \mathbb{U}.X)\ ,\end{aligned}$$

where the isomorphism comes from swapping the two arguments of curried functions. This gives us an isomorphism $wp : (X \xrightarrow{m} \mathbb{F}.Y) \to (\mathbb{U}.Y \xrightarrow{m} \mathbb{U}.X)$ of completely distributive ordered sets. In order to see how function wp acts on modifiers, we observe that any element $c : X \xrightarrow{m} \mathbb{F}.Y$ equals $\lambda x : \lambda p : (p \in c.x)$. Equality (5) therefore implies $c = \lambda x : \lambda p : \varphi.p.(c.x)$. It follows that $wp.c = \lambda p : \lambda x : \varphi.p.(c.x)$ and hence $wp.c.p = \varphi.p \circ c$. The predicate $wp.c.p \in \mathbb{U}.X$ is called the *weakest precondition* for modifier c and postcondition $p \in \mathbb{U}.Y$.

Remark. In [33,42], monotone predicate transformers are modelled by means of up-closed multirelations, where a *multirelation* M is a subset of $X \times \mathbb{P}.Y$, which is called *up-closed* if $A \subseteq B$ and $(x, A) \in M$ always implies $(x, B) \in M$. In other words, a multirelation can be regarded as a function that associates to x an up-closed subset of $\mathbb{P}.Y$. If X and Y are now regarded as discrete ordered sets, an up-closed multirelation is precisely the same as a function $X \to \mathbb{U}.(\mathbb{U}.Y)$. With less emphasis, the same point of view is expressed in Section 15.1 of [4]. Our treatment here has the additional aspects that the state spaces can be ordered and that $\mathbb{F}.X$ is the free distributive completion of X. □

Function wp commutes with sequential composition in the sense that, for any pair of modifiers $c : X \xrightarrow{m} \mathbb{F}.Y$ and $d : Y \xrightarrow{m} \mathbb{F}.Z$, we have

$$\begin{aligned}(6)\qquad & wp.c \circ wp.d = wp.(c; d)\\ \equiv\ & \{\ \text{take arbitrary } p \in \mathbb{P}.Z\ \}\\ & wp.c.(wp.d.p) = wp.(c; d).p\\ \equiv\ & \{\ \text{definition } wp \text{ and sequential composition}\ \}\\ & \varphi.(\varphi.p \circ d) \circ c = \varphi.p \circ (\varphi.d \circ c)\\ \equiv\ & \{\ \text{Theorem 4, associativity of} \circ\ \}\\ & true\ .\end{aligned}$$

This shows that wp is an isomorphism of ordered sets from $X \xrightarrow{m} \mathbb{F}.Y$ to $\mathbb{U}.Y \xrightarrow{m} \mathbb{U}.X$ that preserves composition.

The next result shows that, as one might expect, the junctivity properties of $wp.c$ are the same as the junctivity properties of the resulting states of modifier c.

Theorem 8. *Let $c : X \xrightarrow{m} \mathbb{F}.Y$. The predicate transformer $wp.c$ is conjunctive (disjunctive, finitely conjunctive), if and only if, for every $x \in X$, the resulting state $c.x \in \mathbb{F}.Y$ is conjunctive (disjunctive, finitely conjunctive).*

Proof. For any set of predicates $B \in \mathbb{P}.Y$, we observe

$$\begin{aligned}& \textstyle\bigwedge_{p\in B} wp.c.p = wp.c.(\bigwedge_{p\in B} p)\\ \equiv\ & \{\ \text{definition } wp\ \}\\ & \textstyle\bigwedge_{p\in B} \varphi.p \circ c = \varphi.(\bigwedge_{p\in B} p) \circ c\\ \equiv\ & \{\ \text{equality of functions}\ \}\\ & \textstyle\forall\, x \in X : \bigwedge_{p\in B} \varphi.p.(c.x) = \varphi.(\bigwedge_{p\in B} p).(c.x)\\ \equiv\ & \{\ \text{formula (5)}\ \}\\ & \textstyle\forall\, x \in X : \bigwedge_{p\in B} (c.x).p = (c.x).(\bigwedge_{p\in B} p)\ .\end{aligned}$$

Therefore, conjunctivity of $wp.c$ implies conjunctivity of all alternating states $c.x$, and vice versa. The proofs for disjunctivity and finite conjunctivity are completely analogous. □

4.3 Duality for modifiers

For a modifier $c : X \xrightarrow{m} \mathbb{F}.Y$, we can form the composition $\psi \circ c : X \xrightarrow{m} (\mathbb{F}.Y^\circ)^\circ$, which can also be regarded as the modifier $\psi \circ c : X^\circ \xrightarrow{m} \mathbb{F}.Y^\circ$. This is called the *dual modifier*.

For a monotone predicate $p \in \mathbb{U}.X$, we can form the negation or complement $\neg\, p \in \mathbb{U}.X^\circ$, and conversely. For any transformer $m : \mathbb{U}.Y \to \mathbb{U}.X$, we can therefore form the *dual* transformer $\sim m : \mathbb{U}.Y^\circ \to \mathbb{U}.X^\circ$ by $(\sim m).p = \neg\, m.(\neg p)$. In the context of unordered state spaces, this dual is introduced as the conjugate m^* in [20].

The duality of transformers corresponds to the duality of modifiers, in the sense that $wp.(\psi \circ c) = \sim(wp.c)$. This is proved by observing that, for any $p \in \mathbb{U}.Y^\circ$ and $x \in X$,

$$
\begin{array}{ll}
 & wp.(\psi \circ c).p.x \quad = \quad \neg\, wp.c.(\neg\, p).x \\
\equiv & \{ \text{ definition } wp \ \} \\
 & (\varphi.p \circ \psi \circ c).x \quad = \quad \neg(\varphi.(\neg\, p) \circ c).x \\
\equiv & \{ \text{ definition of composition and (5) } \} \\
 & p \in \psi.(c.x) \quad = \quad \neg\, p \notin c.x \\
\equiv & \{ \text{ formula (4) } \} \\
 & \mathit{true}\ .
\end{array}
$$

4.4 Commands

Up to this point, the ordered sets are the state spaces and the modifiers hold the semantics of the commands. We now want to briefly introduce a command syntax that allows recursive commands. Since we don't want to complicate matters with recursive domain equations, we fix a single ordered set X as the common state space. We write $M.X = (X \xrightarrow{m} \mathbb{F}.X)$ for the ordered set of modifiers of X. Note, however, that $M.X$ is isomorphic to $\mathbb{U}.X \xrightarrow{m} \mathbb{U}.X$ so that the latter space can be used just as well.

We use the abstract syntax for commands introduced in [27]. Let A be a set of symbols to be called *commands*. We assume that A contains all simple commands and procedure names that may be needed. Starting from A, we define the class *Cmd* of *command expressions* inductively by the clauses

- $A \subseteq \mathit{Cmd}$,
- if c and $d \in \mathit{Cmd}$ then $(c; d) \in \mathit{Cmd}$,
- if $(i \in I : c.i)$ is a family in *Cmd* then $(\,[\!]\, i \in I : c.i) \in \mathit{Cmd}$ and $(\,\diamondsuit\, i \in I : c.i) \in \mathit{Cmd}$.

If the family in the third clause is a pair, we use $[\!]$ and $\diamondsuit$ is infix operators. Just as in Section 1.1, they are the syntactic operators for the demonic and angelic choice, respectively.

The semantics are expressed by means of the modifier set $M.X$. The semantic function from the syntactic domain *Cmd* to $M.X$ must primarily transform the syntactic operators into the corresponding semantic ones. We therefore define a function $u : \mathit{Cmd} \to M.X$ to be a *homomorphism* if and only if

$$
\begin{array}{l}
u.(c\,; d) = u.c\,; u.d\ , \\
u.(\,[\!]\, i \in I : c.i) = \bigwedge_{i \in I} u.(c.i)\ , \\
u.(\,\diamondsuit\, i \in I : c.i) = \bigvee_{i \in I} u.(c.i)\ .
\end{array}
$$

Every function $f : A \to M.X$ permits precisely one homomorphism $f^{\odot} : Cmd \to M.X$ with restriction $(f^{\odot}|A) = f$, which is called the homomorphic extension of f. In fact, $f^{\odot}$ is easily constructed inductively.

We assume that the set of commands A is the disjoint union of two sets S and H, which may be infinite. The elements of S are called simple commands. We assume that their semantics is given by a function $sem0 : S \to M.X$. The elements of H are called procedure names. Every $h \in H$ is supposed to be equipped with a declaration $\mathbf{body}.h \in Cmd$.

We define $sem1 : A \to M.X$ to be the least solution $u : A \to M.X$ of the system of equations $(u|S) = sem0$ and $u.h = u^{\odot}.(\mathbf{body}.h)$. The semantics of a command expression c is now defined by $[\![\, c \,]\!] = sem1^{\odot}.c \in M.X$. For a command expression c that does not contain procedure names, $[\![\, c \,]\!]$ is completely determined by $sem0$ and the fact that $sem1^{\odot}$ is a homomorphism that coincides with $sem0$ on S.

We assume that S contains the set of the monotone functions $X \xrightarrow{m} X$ and the set of the monotone predicates $\mathbb{U}.X = (X \xrightarrow{m} \mathbb{B})$. For a monotone function $s \in (X \xrightarrow{m} X) \subseteq S$ the semantics is given by $sem0.s = j \circ s \in M.X$. Let the identity function of X be called $skip$. It follows that $sem0.skip = j$. We define $i : \mathbb{B} \xrightarrow{m} \mathbb{F}.X$ to be the order embedding given by $i.false = \bot$ and $i.true = \top$. For a monotone predicate $p \in \mathbb{U}.X \subseteq S$, the semantics is given by $sem0.p = i \circ p$. If we use conditional expressions like in the programming language C, it follows that

$$[\![\, skip \,[\!]\, p \,]\!].x = (p.x ?\, j.x : \bot) \ ,$$
$$[\![\, skip \diamondsuit p \,]\!].x = (p.x ?\, \top : j.x) \ .$$

Command $skip \,[\!]\, p$ is called the *assertion* of p and command $skip \diamondsuit p$ is called the *guard* of $\neg\, p$. Notice, however, that predicate $\neg\, p$ is antimonotone and is usually not monotone.

4.5 The case of the discrete state space

We now specialize further by assuming that the order of the state space X is discrete. This has the effect that all functions on X are monotone. In particular, we can assign meanings $sem0.f$ to all functions $f \in X \to X$, and, for every predicate $p \in \mathbb{P}.X$, we can define $!\,p = (skip \,[\!]\, p)$ and $?p = (skip \diamondsuit \neg\, p)$. Either form can be used for the construction of the conditional choice between command expressions c and d. The usual definition is

$$\mathbf{if}\ p\ \mathbf{then}\ c\ \mathbf{else}\ d\ \mathbf{end} \quad = \quad (?p\, ; c \,[\!]\, ?\neg\, p\, ; d) \ .$$

According to [14, p. 176], this formula goes back to [32].

One can easily show that the same semantics is obtained by taking $(!\,p; c \diamondsuit !\neg\, p\, ; d)$, or, even simpler, by $(p \,[\!]\, c) \diamondsuit (\neg\, p \,[\!]\, d)$ or $(\neg\, p \diamondsuit c) \,[\!]\, (p \diamondsuit d)$.

Specializing further, we assume that the state space is spanned by a set V of programming variables, and that every variable $v \in V$ has values in the set (or type) $T.v$. This means that $X = \prod_{v \in V} T.v$. For every variable $v \in V$, a state $s \in X$ has the component $s.v \in T.v$, which is regarded as the value of v in state s. If $r \in X \to T.v$ for some $v \in V$, the assignment $v := r$ is regarded as a denotation of the function $f \in X \to X$ given by $f.s.v = r.s$ and $f.s.w = s.w$ for all variables $w \neq v$. So, we have $[\![\, v := r \,]\!] = j \circ f$.

In the case of a discrete state space X, it holds that $X^{\circ} = X$, so that the duality function ψ has the type $\mathbb{F}.X \xrightarrow{m} (\mathbb{F}.X)^{\circ}$. One might conjecture that, in this case, all fixpoints of function ψ are proper, but this is not the case, provided X has at least three elements.

Example. Assume that X has three elements. Then $\mathbb{F}.X$ has three different proper states, say a, b, and c. One can verify that the alternating state $y = (a \vee b) \wedge (a \vee c) \wedge (b \vee c)$ satisfies $\psi.y = y$, but y is not proper. In total, $\mathbb{F}.X$ has 20 elements: four self-dual ones a, b, c, y, and eight pairs of duals, like $\perp$ and $\top$, $a \wedge b \wedge c$ and $a \vee b \vee c$, etc. The easiest way to verify such assertions seems to be to represent $\mathbb{F}.X$ as $\mathbb{U}.(\mathbb{P}.X)$, where $\mathbb{P}.X$ is represented by means of bit vectors.

5 Conclusions

In comparison with [36], we have removed much of the terminology, like terms as "join dense" and "completely join prime", etc. By not regarding the state space X a priori as an ordered subset of its free distributive completion (FDC), we have obtained an axiomatization that characterizes the FDC. Indeed, Morris [36] explicitly states that the FDC is a model for his axioms, but he does not claim that his axioms characterize the FDC.

The concrete construction of the FDC of X as $\mathbb{U}.(\mathbb{U}.X)$ where $\mathbb{U}.X = (X \xrightarrow{m} \mathbb{B})$, offers relevant information for applications to programming semantics. It allows purely state based semantics that combine the two opposite interpretations of nondeterminism, even in the presence of an order on the proper state space.

The FDC concept is self-dual, but the concrete construction is not. This construction therefore induces a duality into the theory. This duality interchanges the opposite interpretations of nondeterminism.

We have kept the theory general, so that it can be applied to various kinds of semantic proposals, like functional, imperative, or process-algebraic.

References

1. K.R. Apt and G.D. Plotkin. Countable nondeterminism and random assignment. *Journal ACM*, 33:724–767, 1986.
2. A. Arnold and M. Nivat. Non deterministic recursive program schemes. In *Fundamentals of Computation Theory*, volume 56 of *LNCS*, pages 12–21, New York, 1977. Springer.
3. R.J.R. Back and J. von Wright. A lattice theoretical basis for a specification language. In J.L.A. van de Snepscheut, editor, *Mathematics of Program Construction*, volume 375 of *LNCS*, pages 139–156, New York, 1989. Springer.
4. R.J.R. Back and J. von Wright. *Refinement Calculus, a systematic approach.* Springer, New York, 1998.
5. R. Balbes and Ph. Dwinger. *Distributive Lattices.* University of Missouri Press, 1974.
6. M. Barr and C. Wells. *Category Theory for Computing Science.* Prentice Hall International, 1990.
7. H.J. Boom. A weaker precondition for loops. *ACM Trans. Program. Lang. Syst.*, 4:668–677, 1982.
8. M. Broy, R. Gratz, and M. Wirsing. Semantics of non-deterministic and noncontinuous constructs. In F.L. Bauer and M. Broy, editors, *Program Construction, International Summer School Marktoberdorf*, volume 69 of *LNCS*, pages 553–591, New York, 1979. Springer.
9. A.K. Chandra, D.C. Kozen, and L.J. Stockmeyer. Alternation. *J. ACM*, 28(1):114–133, 1981.
10. K.M. Chandy and J. Misra. *Parallel Program Design, A Foundation.* Addison–Wesley, 1988.
11. N. Chomsky. Context-free grammar and pushdown storage. *Quarterly Progress Report, MIT Research lab. in Electronics*, 65:187–194, 1962.
12. S.A. Cook. The complexity of theorem proving procedures. In *Proc. Third Annual ACM Symposium on the Theory of Computing*, pages 151–158, 1971.

13. J.W. de Bakker. *Mathematical Theory of Program Correctness.* Prentice-Hall Int., Englewood Cliffs, 1980.
14. J.W. de Bakker and W.P. de Roever. A calculus for recursive program schemes. In M. Nivat, editor, *Automata, Languages and Programming 1972*, pages 167–196. North Holland, 1973.
15. W.P. de Roever. Dijkstra's predicate transformer, non-determinism, recursion, and termination. In *Mathematical Foundations of Computer Science*, volume 45 of *LNCS*, pages 472–481, New York, 1976. Springer.
16. E.W. Dijkstra. Co-operating sequential processes. In *Programming Languages*, pages 31–39, London, etc., 1968. NATO Advanced Study Institute, Academic Press.
17. E.W. Dijkstra. Guarded commands, nondeterminacy and formal derivation of programs. *Commun. ACM*, 18:453–457, 1975.
18. E.W. Dijkstra. *A Discipline of Programming.* Prentice Hall, 1976.
19. E.W. Dijkstra and C.S. Scholten. The operational interpretation of extreme solutions. Tech. Rept., Tech. Univ. Eindhoven, EWD 883, see `www.cs.utexas.edu/users/EWD`, 1984.
20. E.W. Dijkstra and C.S. Scholten. *Predicate Calculus and Program Semantics.* Springer Verlag, 1990.
21. R.M. Dijkstra. DUALITY: a simple formalism for the analysis of UNITY. *Formal Aspects of Comput.*, 7:353–388, 1995.
22. R.W. Floyd. Assigning meanings to programs. In *Proceedings of the Symposium on Applied mathematics*, volume 19, pages 19–32. AMS, 1967.
23. W.H. Hesselink. Interpretations of recursion under unbounded nondeterminacy. *Theor. Comput. Sci.*, 59:211–234, 1988.
24. W.H. Hesselink. A mathematical approach to nondeterminism in data types. *ACM Trans. Program. Lang. Syst.*, 10:87–117, 1988.
25. W.H. Hesselink. Modalities of nondeterminacy. In W.H.J. Feijen et al., editors, *Beauty is our business, a birthday salute to Edsger W. Dijkstra*, pages 182–192. Springer, 1990.
26. W.H. Hesselink. LR-parsing derived. *Sci. Comput. Program.*, 19:171–196, 1992.
27. W.H. Hesselink. Nondeterminacy and recursion via stacks and games. *Theor. Comput. Sci.*, 124:273–295, 1994.
28. W.H. Hesselink. Predicate transformers for recursive procedures with local variables. *Formal Aspects of Computing*, 11:616–636, 1999.
29. W.H. Hesselink. Universally distributive ordered sets –several known results–. `http://www.cs.rug.nl/~wim/pub/mans.html`, January 2007.
30. C.A.R. Hoare. An axiomatic basis for computer programming. *Commun. ACM*, 12:576–583, 1969.
31. C.A.R. Hoare. *Communicating Sequential Processes.* Prentice Hall, 1985.
32. R.M. Karp. *Some applications of logical syntax to digital computer programming.* Thesis, Harvard University, 1959.
33. C.E. Martin, S.A. Curtis, and I. Rewitzky. Modelling nondeterminism. In *Mathematics of Program Construction*, volume 3125 of *LNCS*, pages 228–251. Springer, 2004.
34. C. Morgan. *Programming from Specifications.* Prentice Hall, Englewood Cliffs, NJ, 1990.
35. C. Morgan and P.H.B. Gardiner. Data refinement by calculation. *Acta Inf.*, 27:481–503, 1990.
36. J.M. Morris. Augmenting types with unbounded demonic and angelic nondeterminacy. In *Mathematics of Program Construction*, volume 3125 of *LNCS*, pages 274–288, New York, 2004. Springer.
37. J.M. Morris and M. Tyrrell. Dual unbounded nondeterminacy, recursion, and fixpoints. *Acta Inf.*, 44:323–344, 2007.
38. J.M. Morris and M. Tyrrell. Terms with unbounded demonic and angelic nondeterminacy. *Sci. Comput. Program.*, 65:159–172, 2007.
39. P. Naur. Proof of algorithms by general snapshots. *BIT*, 6:310–316, 1966.
40. D. Park. On the semantics of fair parallelism. In *Abstract Software Specifications*, volume 86 of *LNCS*, pages 504–526. Springer, 1980.
41. M.O. Rabin and D. Scott. Finite automata and their decision problems. *IBM J. Res.*, 3:115–125, 1959.

42. I. Rewitzky. Binary multirelations. In H. de Swart, E. Orlowska, G. Schmidt, and M. Roubens, editors, *Theory and Application of Relational Structures as Knowledge Instruments*, volume 2929 of *LNCS*, pages 259–274, New York, 2004. Springer.
43. W.R. Tunnicliffe. The free completely distributive lattice over a poset. *Algebra Universalis*, 21:133–135, 1985.
44. A.M. Turing. On checking a large routine. In *Report of a Conference on High-Speed Automatic Calculating Machines*, pages 67–69. University Mathematical Laboratory, Cambridge, 1949.