Computing coproducts of finitely presented Goedel algebras

Vincenzo Marra

with Ottavio M. D'Antona

Dipartimento di Informatica e Comunicazione, Università degli Studi di Milano
via Comelico 39-41, 20100 Milan, Italy

marra@dico.unimi.it

A *Gödel algebra* (a.k.a. an *L-algebra* or a *Gödel-Dummett algebra*) is a Heyting algebra satisfying the prelinearity axiom $(\alpha \to \beta) \vee (\beta \to \alpha) = 1$. The variety of Gödel algebras is locally finite, whence finitely generated Gödel algebras coincide with finite or finitely presented ones. Our main result is an algorithm to compute finite coproducts of finitely generated Gödel algebras.

Let G denote the category of Gödel algebras, and G_{fp} the full subcategory of finitely presented algebras. A *forest* is a finite poset F such that for every $x \in F$, the set $\{y \in F \mid y \leq x\}$ is a chain (i.e. is totally ordered) when endowed with the order inherited from P. A *tree* is a forest with a bottom element. Let F denote the category of forests and open order-preserving maps, and T the full subcategory of trees. (Recall that an order-preserving map $f \colon A \to B$ between posets is *open* iff it carries down-sets to down-sets.) A straightforward development shows that the spectral functor **Spec** yields an equivalence between $\mathsf{G}_{\mathsf{fp}}^{\mathsf{op}}$ and F. Trees correspond to finitely presented Gödel algebras with a unique maximal filter. (Remark: As a matter of convention, we are ordering prime filters by *reverse* inclusion just to make our trees grow upwards.) Standard considerations allow one to reduce computation of (finite) coproducts in G_{fp} to computation of (finite) products in T. We remark in passing that equalisers in T can be effectively computed without difficulties, whence computation of fibred products in T (or fibred coproducts in G_{fp}) follows at once from our main result below.

The core of this piece of work is thus computation of products in T. An *ordered partition* σ is a finite chain of pairwise disjoint nonempty finite sets. We write $\sigma = \{S_1, \ldots, S_m\}$ to mean that S_i precedes S_j iff $i \leq j$. Given ordered partitions $\sigma = \{S_1, \ldots, S_m\}, \tau = \{T_1, \ldots, T_n\}$ with $m \leq n$, we let $\sigma \leq \tau$ iff $S_i = T_i$ for every $i \in \{1, \ldots, m\}$. A *foliage* is a set of mutually incomparable (according to $\leq$) ordered partitions. Given an ordered partition σ, its *support* is $\mathrm{supp}\sigma = \bigcup \sigma$. Similarly, if T is a foliage, we set $\mathrm{supp}T = \bigcup_{\sigma \in T} \mathrm{supp}\sigma$. For σ and τ ordered partitions with disjoint supports, we define a *merged shuffle* of σ and τ to be any ordered partition constructed in a certain (herein not detailed) manner from σ and τ. (Remark: the set of merged shuffles of σ and τ is finite and effectively computable.) If S and T are foliages with $\mathrm{supp}S \cap \mathrm{supp}T = \emptyset$, we call

$$S \times T = \{\theta \mid \theta \text{ is a merged shuffle of some } \sigma \in S, \tau \in T\}$$

the *product* of S and T. It is possible to show that $S \times T$ is a foliage.
Given a foliage T, we set

$$\mathrm{Tree}T = \{\sigma \mid \sigma \text{ is an ordered partition such that } \sigma \leq \tau \in T\}.$$

A moment's reflection shows $\mathrm{Tree}T$ is a tree for any foliage T. Given an ordered partition $\theta = \{B_1, \ldots, B_n\}$ and a set X, we let $\theta - X$ denote the ordered partition $\{B_1 \setminus X, \ldots, B_n \setminus X\} \setminus \{\emptyset\}$, where $\setminus$ is set-theoretic difference. Let S and T be foliages, and set $X = \mathrm{supp}S$, $Y = \mathrm{supp}T$. Assume $X \cap Y = \emptyset$. Let $A = \mathrm{Tree}S$, $B = \mathrm{Tree}T$, and $C = \mathrm{Tree}(S \times T)$. We define a function $\pi_S \colon C \to A$ by $\theta \mapsto \theta - Y$, and similarly $\pi_T \colon C \to B$ by $\theta \mapsto \theta - X$. We call π_S and π_T the *projections induced by* $S \times T$. It turns out that π_S and π_T are morphisms (in fact, epimorphisms) in T.

Main Theorem. Let S and T be foliages such that $\mathrm{supp}S \cap \mathrm{supp}T = \emptyset$. Then

$$\mathrm{Tree}S \overset{\pi_S}{\leftarrow} \mathrm{Tree}(S \times T) \overset{\pi_T}{\to} \mathrm{Tree}T$$

is the product of $\mathrm{Tree}S$ and $\mathrm{Tree}T$ in T.
On the basis of this theorem, it is an easy matter to set up an explicit algorithm to compute finite products of trees.

If time allows, we shall offer a small sample of applications of our main result.