Foundation of a Computable Solid Modelling

Abbas Edalat
Department of Computing, Imperial College, London, UK

André Lieutier
Dassault Systemes Provence and
LMC/IMAG, Aix-en-Provence and Grenoble, France

Abstract

Solid modelling and computational geometry are based on classical topology and geometry in which the basic predicates and operations, such as membership, subset inclusion, union and intersection, are not continuous and therefore not computable. But a sound computational framework for solids and geometry can only be built in a framework with computable predicates and operations. In practice, correctness of algorithms in computational geometry is usually proved using the unrealistic Real RAM machine model of computation, which allows comparison of real numbers, with the undesirable result that correct algorithms, when implemented, turn into unreliable programs. Here, we use a domain-theoretic approach to recursive analysis to develop the basis of an effective and realistic framework for solid modelling. This framework is equipped with a well-defined and realistic notion of computability which reflects the observable properties of real solids. The basic predicates and operations on solids are computable in this model which admits regular and non-regular sets and supports a design methodology for actual robust algorithms. Moreover, the model is able to capture the uncertainties of input data in actual CAD situations.

1 Introduction

The current frameworks for solid modelling and computational geometry are based, on the one hand, on discontinuous predicates and Boolean operations, and, on the other hand, on comparison of real numbers. These essential foundations of the existing theory and implementations are both unjustified and unrealistic.

Topology and geometry, as mainstream mathematical disciplines, have been developed to study continuous transformations on spaces. It is therefore ironical that the main building blocks of these subjects, namely the membership predicate of a set, the subset inclusion predicate and the basic operations such as union and intersections, are generally not continuous and therefore non-computable. For example, in any Euclidean space the membership predicate of any proper subset is discontinuous at the boundary of the subset; whereas the binary intersection, as an operator on compact subsets, is discontinuous with respect to the Hausdorff metric whenever the two input compact subsets touch each other. This non-continuity creates a foundational problem in computation, which has so far been essentially neglected. In fact, in order to construct a sound computational model for solids and geometry, one needs a framework in which these elementary building blocks are continuous and computable.

In practice, correctness of algorithms in computational geometry is usually proved using the Real RAM machine [27] model of computation, in which comparison of real numbers is considered to be decidable. Since this model is not realistic, correct algorithms, when implemented, turn into unreliable programs. In CAGD modelling operators, the effect of rounding errors on consistency and robustness of actual implementations is an open question, which is handled in industrial software by various unreliable and expensive "up to epsilon" heuristics that remain very unsatisfactory.

The solid modelling framework provided by classical analysis, which allows discontinuous behaviour and comparison of exact real numbers, is not realistic as a model of our interaction with the physical world in terms of measurement and manufacturing. Nor is it realistic as a basis for the design of algorithms implemented on realistic machines, which can only deal with finite data. Industrial solid modelling software used for CAGD (Computer Aided Geometric Design), CAM (Computer Aided Manufacturing) or robotics is therefore infected by the disparity between the classical analysis paradigm and feasible computations. This disparity, as well as the representation of uncertainties in the geometry of the solid objects, is handled case by case, by various expensive and unsatisfactory "up to epsilon" ad hoc heuristics. It is difficult, if at all possible, to improve and generalize these techniques, since their relatively poor success depends on the skill and experience of software engineers rather than on a well formalised methodology. In practice, the maintenance cost of some central geometric operators such as the Boolean operations or some specific variants of the Minkowski sum has always remained critical [24, 14, 23].

The authors claim that a robust algorithm is one whose correctness is proved with the assumption of a realistic machine model [22]. Recursive analysis defines precisely what it means, in the context of the realistic Turing machine model of computation, to compute objects belonging to non-countable sets such as the set of real numbers. There are various approaches to computable analysis, including the Type 2 Theory of Effectivity (TTE) [21, 35, 36], based on a computation with a machine, the algebraic domain approach [33, 34], based on embedding classical spaces into algebraic domains, the continuous domain approach [9, 10, 13, 12], based on embedding classical spaces into the set of maximal elements of continuous domains, and the more recent approach by Equilogical Spaces [31, 4, 5], based on taking quotients of T_0 topological spaces. In recent years, Brattka and Weihrauch have also studied the question of computability of closed and compact subsets of Euclidean spaces in the context of TTE [6].

In this paper, which is based on the preliminary work in [11], we use a domain-theoretic approach to recursive analysis to develop the foundation of an effective framework for solid modelling. We introduce the continuous domain of solid objects which gives a concrete model of computation on solids close to the actual practice by CAD engineers. In this model, the basic predicates, such as membership and subset inclusion, and operations, such as union and intersection, are continuous and computable. The set-theoretic aspects of solid modelling are revisited, leading to a theoretically motivated model that shows some interesting similarities with the Requicha Solid Model [28, 29]. Within this model, some unavoidable limitations of solid modelling computations are proved and a sound framework to design specifications for feasible modelling operators is provided. Some consequences in computation with the boundary representation paradigm are sketched that can incorporate existing methods [16, 32, 19, 17, 18] into a general, mathematically well-founded theory. Moreover, the model is able to capture the uncertainties of input data [7, 25] in actual CAD situations.

We need the following requirements for the mathematical model:

(1) the notion of computability of solids has to be well defined,

(2) the model has to reflect the observable properties of real solids,

(3) it has to be closed under the Boolean operations and all basic predicates and operations have to be computable,

(4) non-regular sets[1] have to be captured by the model as well as regular solids,

(5) the model has to support a design methodology for actual robust algorithms.

A general methodology for the specification of feasible operators and the design of robust algorithms should rely on a sound mathematical model. This is why the authors believe that the domain-theoretic approach is a powerful framework both to model partial or uncertain data and to guide the design of robust software.

The paper is organised as follows. In Section 2, we introduce the solid domain, a mathematical model for computable rigid solids, which satisfies the above properties. Section 3 shows that the basic predicates and Boolean operations are continuous in this model. Using a standard theory of computability for domains, Section 4 presents a computability theory for our model, which is consistent with computing solids with a realistic machine. Sections 5 and 6 enrich the domain-theoretic notion of computability with a quantitative measure of convergence with respect to the Hausdorff metric and the Lebesgue measure respectively. Section 7 presents our conclusion and sketches the outline of future work including the implementation of this framework. Finally, in the Appendix, we collect together the basic notions of domain theory that we use in this paper.

2 The Solid Domain

In this section, we introduce the solid domain, a mathematical model for representing rigid solids. We focus here on the set-theoretic aspects of solid modelling as Requicha

[1] An open set is regular if it is the interior of its closure.

did in introducing the r-sets model [28]. Our model is motivated by requirements (1) to (5) given in the introduction.

For any subset A of a topological space, $\overline{A}$, A°, ∂A and A^c denote respectively the closure, the interior, the boundary and the complement of A. Recall, for example from [8, page 92], that an open set is *regular* if it is the interior of its closure; dually, a closed set is regular if it is the closure of its interior. The complement of a regular open set then is a regular closed set and vice versa. The interior of a regular closed set is a regular open set, whereas the closure of a regular open set is a regular closed set. Finally, the intersection of two regular open sets is regular. The *regularization* of an open set is the interior of its closure; the regularization of a closed set is the closure of its interior. Therefore, the regularized binary union of open sets O_1 and O_2 is the set $(\overline{O_1 \cup O_2})^\circ$.

Definition 2.1 *The* solid domain $(\mathbf{S}X, \sqsubseteq)$ *of a topological space X is the set of ordered pairs (A, B) of disjoint open subsets of X endowed with the information order:* $(A_1, B_1) \sqsubseteq (A_2, B_2) \iff A_1 \subseteq A_2$ *and* $B_1 \subseteq B_2$.

An element (A, B) of $\mathbf{S}X$ is called a *partial solid*: A and B are intended to capture, respectively, the interior and the exterior (interior of the complement) of a solid object, possibly, at some finite stage of computation. Note that $(\mathbf{S}X, \sqsubseteq)$ is a directed complete partial order; the least upper bound (lub) of a directed family of partial solid objects $(A_i, B_i)_{i \in I}$ is given by $\bigsqcup_{i \in I}(A_i, B_i) = (\bigcup_{i \in I} A_i, \bigcup_{i \in I} B_i)$. The solid domain is isomorphic with the function space $X \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$, i.e. the collection of continuous functions $f : X \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$ ordered pointwise. Here, $\{\mathsf{tt}, \mathsf{ff}\}_\perp$ is the lift of $\{\mathsf{tt}, \mathsf{ff}\}$ equipped with its Scott topology. By duality of open and closed sets, $(\mathbf{S}X, \sqsubseteq)$ is also isomorphic with the collection of ordered pairs (A, B) of closed subsets of X with $A \cup B = X$ with the information ordering: $(A_1, B_1) \sqsubseteq (A_2, B_2)$ $\iff A_2 \subseteq A_1$ *and* $B_2 \subseteq B_1$.

In fact, $\mathbf{S}$ is a contravariant functor on the category **TOP** of topological spaces and continuous maps. Given a continuous function $f : X \to Y$ of topological spaces X and Y, we have a continuous function $\mathbf{S}f : \mathbf{S}Y \to \mathbf{S}X$ defined by $(\mathbf{S}f)(A, B) = (f^{-1}A, f^{-1}B)$.

Proposition 2.2 *The partial solid $(A, B) \in (\mathbf{S}X, \sqsubseteq)$ is a maximal element iff $A = B^{c\circ}$ and $B = A^{c\circ}$.*

Proof Let (A, B) be maximal. Since A and B are disjoint open sets, it follows that $A \subseteq B^{c\circ}$. Hence, $(A, B) \sqsubseteq (B^{c\circ}, B)$ and thus $A = B^{c\circ}$. Similarly, $B = A^{c\circ}$. This proves the "only if" part. For the "if part", suppose that $A = B^{c\circ}$ and $B = A^{c\circ}$. Then, any proper open superset of A will have non-empty intersection with B and any proper open superset of B will have non-empty intersection with A. It follows that (A, B) is maximal. $\square$

Corollary 2.3 *If (A, B) is a maximal element, then A and B are regular open sets. Conversely, for any regular open set A, the partial solid $(A, A^{c\circ})$ is maximal.*

Proof For the first part, note that A is the interior of the closed set B^c and is, therefore, regular; similarly B is regular. For the second part, observe that $A^{c\circ c\circ} = (A^{c\circ c})^\circ = (\overline{A})^\circ = A$. $\square$

We define $(A, B) \in \mathbf{S}X$ to be a *classical* solid object if $\overline{A} \cup \overline{B} = X$.

Proposition 2.4 *Any maximal element is a classical solid object.*

Proof Suppose (A, B) is maximal. Then $X = A \cup \partial A \cup A^{c\circ} = \overline{A} \cup \overline{B}$, since $\overline{A} = A \cup \partial A$ and $A^{c\circ} \subseteq \overline{A^{c\circ}} = \overline{B}$. $\square$

We need the following lemma on regular open sets for later results.

Lemma 2.5 *If A is a regular open set of a topological space, then $\partial A = \partial(A^{c^\circ})$.*

Proof Suppose $x \in \partial A$. Then any neighbourhood of x contains an element of $A \subseteq A^{c^{\circ c}}$. Assume now that O is a neighbourhood of x which does not contain any element of A^{c°. Then $O \subseteq \overline{A}$, and hence by regularity of A, we have $O \subseteq A$ which contradicts $x \in \partial A$. This shows that $\partial A \subseteq \partial(A^{c^\circ})$. By symmetry we also get $\partial A \supseteq \partial(A^{c^\circ})$, since A^{c° is also a regular open set with $A^{c\circ c\circ} = A$. $\square$

We next show that the solid domain is continuous for a suitable class of topological spaces.

Theorem 2.6 *Let X be a locally compact Hausdorff space. Then the solid domain $(\mathbf{S}X, \sqsubseteq)$ is a bounded complete continuous domain and $(A_1, B_1) \ll (A_2, B_2)$ iff $\overline{A_1}$ and $\overline{B_1}$ are compact subsets of A_2 and B_2 respectively. If X is second countable, then $(\mathbf{S}X, \sqsubseteq)$ is ω-continuous.*

Proof This is a simple exercise which can be proved directly or it can be deduced from more general results as follows. From [15, page 129, II.4.6], it follows that $(\mathbf{S}X, \sqsubseteq)$ is a continuous domain with $(A_1, B_1) \ll (A_2, B_2)$ iff there are pairs (A_3, B_3) of compact sets such that $A_1 \subseteq A_3 \subseteq A_2$ and $B_1 \subseteq B_3 \subseteq B_2$, which gives us the desired condition since a closed subset of a compact set in a Hausdorff space is compact. If X is second countable, then it will have a countable basis, which contains the empty set, is closed under binary intersection and regularized binary union, and consists of regular open sets whose closures are compact. The collection of pairs of disjoint elements of this basis will provide a countable basis for $(\mathbf{S}X, \sqsubseteq)$. $\square$

Proposition 2.7 *Any classical solid object $(A, B) \in \mathbf{S}X$, with $A \neq \emptyset \neq B$, of a connected, locally compact Hausdorff space X is maximal with respect to the way-below relation.*

Proof If $(A, B) \ll (A', B') \in \mathbf{S}X$, then we must have $\overline{A} \subseteq A'$ and $\overline{B} \subseteq B'$. Therefore, $A' \cup B' = X$ with $A' \neq \emptyset \neq B'$. This contradicts the connectedness of X, since A' and B' are disjoint open sets. Hence, (A, B) is maximal with respect to the way-below relation. $\square$

Remark 2.8 *In fact if the lattice of open sets of a topological space X is continuous, then [15, II.4.6, page 129] implies that $(\mathbf{S}X, \sqsubseteq)$ is a continuous domain. In particular, it follows that one does not need X to be Hausdorff: the solid domain of any locally compact space is continuous. In that case we have: $(A_1, B_1) \ll (A_2, B_2)$ iff there are pairs (A_3, B_3) of compact saturated sets[2] such that $A_1 \subseteq A_3 \subseteq A_2$ and $B_1 \subseteq B_3 \subseteq B_2$. In this paper, however, we will restrict our attention to locally compact Hausdorff spaces only.*

In practice, we are often interested in the subdomain $\mathbf{S}_b X$ of *bounded* partial solids which is defined as $\mathbf{S}_b X = \{(A, B) \in \mathbf{S}X | B^c \text{ is compact}\} \cup \{(\emptyset, \emptyset)\}$, ordered by inclusion. It is easy to see that $\mathbf{S}_b X$ is a subdcpo of $\mathbf{S}X$. Moreover:

Proposition 2.9 *If X is a (second countable) locally compact Hausdorff space, $\mathbf{S}_b X$ is (ω)-continuous with the way-below relation given by $(A_1, B_1) \ll (A_2, B_2)$ iff $\overline{A_1}$ is a compact subset of A_2 and $B_2^c \subseteq {B_1}^{c\circ}$.*

From now on, unless otherwise stated, X is a locally compact second countable Hausdorff space.

3 Predicates and Operations on Solids

We will next define the membership predicate on $\mathbf{S}X$. In order to motivate our definition, assume for the discussion below that $X = \mathbb{R}^d$. Given any proper subset of $S \subseteq \mathbb{R}^d$, the classical membership predicate $\in_S : \mathbb{R}^d \to \{\mathsf{tt}, \mathsf{ff}\}$ is continuous except on ∂S. In fact, if S is an open or closed set, then its boundary has empty interior and it is not decidable that a point is on ∂S. For example if $X = \mathbb{R}$ and S is the set of positive numbers, then a real number $x \in \mathbb{R}$ is on the boundary of S iff $x = 0$ which is not decidable in computable analysis [26, page 23]. It therefore makes sense from a computational viewpoint to redefine the membership predicate as the continuous function: $\in'_S : \mathbb{R}^d \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$ where the value $\perp$ is taken on ∂S. We call this the *continuous membership predicate*. Then, two subsets are equivalent if and only if they have the same continuous membership predicate, i.e. if they have the same interior and the same exterior (interior of complement). By analogy with general set theory for which a set is completely defined by its membership predicate, the solid domain can be seen as the collection of subsets that can be distinguished by their continuous membership predicates. The definition of the solid domain is then consistent with requirement (1) since a computable membership predicate has to be continuous.

Our definition is also consistent with requirement (2) in a closely related way. We consider the idealization of a machine used to measure mechanical parts. Two parts corresponding to equivalent subsets cannot be distinguished by such a machine. Moreover, partial solids, and, more generally, domain-theoretically defined

[2] A set is saturated if it is upper closed with respect to the specialisation ordering.

data types allow us to capture partial, or uncertain input data [7, 25] encountered in realistic CAD situations. In order to be able to compute the continuous membership predicate on X, we extend it to the upper space $\mathbf{U}X$ by defining $- \in - : \mathbf{U}X \times \mathbf{S}X \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$ with:

$$C \in (A, B) = \begin{cases} \mathsf{tt} & \text{if } C \subseteq A \\ \mathsf{ff} & \text{if } C \subseteq B \\ \perp & \text{otherwise} \end{cases}$$

Note that we use the infix notation for predicates and Boolean operations. When $X = \mathbb{R}^d$, it is more convenient to use the interval domain $\mathbf{I}\mathbb{R}^d$ instead of the upper space and define the membership predicate as: $- \in - : \mathbf{I}\mathbb{R}^d \times \mathbf{S}\mathbb{R}^d \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$.

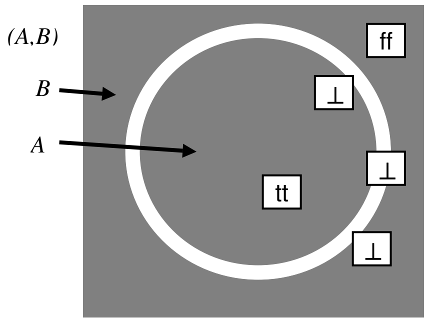

Figure 1: The membership predicate of a partial solid object of the unit square.

We define the predicate $- \subseteq - : \mathbf{S}_b X \times \mathbf{S}X \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$, by

$$(A, B) \subseteq (C, D) = \begin{cases} \mathsf{tt} & \text{if } B \cup C = X \\ \mathsf{ff} & \text{if } A \cap D \neq \emptyset \\ \perp & \text{otherwise} \end{cases}$$

The restriction to $\mathbf{S}_b X$ will ensure that $- \subseteq -$ is continuous, as we will see below. Starting with the continuous membership predicate, the natural definition for the complement would be to swap the values tt and ff. This means that the complement of (A, B) is (B, A), cf. requirement (3).

As for requirement (4), the figure below represents a subset S of $X = [0, 1]^2$ that is not regular (Fig. 2). Its regularization removes both the external and internal "dangling edge". This set can be captured in our framework but not in the Requicha model. Here and in subsequent figures, the two components A and B of the partial solid are, for clarity, depicted separately below each picture.

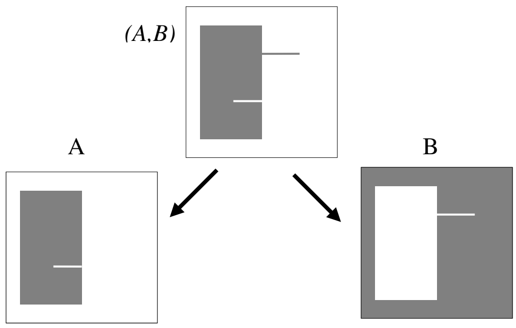

Figure 2: Representation of a non-regular solid.

Next we consider the Boolean operators. First note that the regularization operator $R : \mathbf{S}X \to \mathbf{S}X$ defined by $R((A,B)) = ((\overline{A})^\circ, (\overline{B})^\circ)$ is not continuous, and hence not computable. To see this, suppose $X = \mathbb{R}$ and consider the partial solid $(\mathbb{R} \setminus \{0\}, \emptyset)$. Then

$$\bigsqcup_{n\geq 1} (\mathbb{R} \setminus [-\frac{1}{n}, \frac{1}{n}], \emptyset) = (\mathbb{R} \setminus \{0\}, \emptyset),$$

but

$$\bigsqcup_{n\geq 1} R(\mathbb{R}\setminus[-\frac{1}{n}, \frac{1}{n}], \emptyset)) = \bigsqcup_{n\geq 1} (\mathbb{R}\setminus[-\frac{1}{n}, \frac{1}{n}], \emptyset) = (\mathbb{R}\setminus\{0\}, \emptyset) \neq (\mathbb{R}, \emptyset) = R((\mathbb{R}\setminus\{0\}, \emptyset)).$$

Furthermore, the regularized union [28, 29] of two adjacent three dimensional boxes (i.e. product of intervals) is not computable, since, to decide whether the adjacent faces are in contact or not, one would have to decide the equality of two real numbers which is not computable. Requirements (1) and (3) entail the existence of Boolean operators which are computable with respect to a realistic machine model (e.g. the Turing machine).

In order to define Boolean operators on the solid domain, we obtain the truth table of logical Boolean operators on $\{\mathsf{tt}, \mathsf{ff}, \bot\}$. Consider the logical Boolean operator "or", which, applied to the continuous membership predicates of two partial solids, would define their union.

$\vee$	tt	ff	$\bot$
tt	tt	tt	tt
ff	tt	ff	$\bot$
$\bot$	tt	$\bot$	$\bot$

This is indeed the truth table for *parallel or* in domain theory; see [2, page 133]. One can likewise build the truth table for "and". Note the similarities with the (In,On,Out) points classifications used in some boundary representation based algorithms [30, 3]. From these truth tables, we can deduce the definition of Boolean operators on partial solids:

$$
\begin{aligned}
(A_1, B_1) \cup (A_2, B_2) &= (A_1 \cup A_2, B_1 \cap B_2) \\
(A_1, B_1) \cap (A_2, B_2) &= (A_1 \cap A_2, B_1 \cup B_2).
\end{aligned}
$$

One can likewise define the n-ary union and the n-ary intersection of partial solids. Note that, given two partial solids representing adjacent boxes, their union would not represent the set-theoretic union of the boxes, as illustrated in Fig. 3.

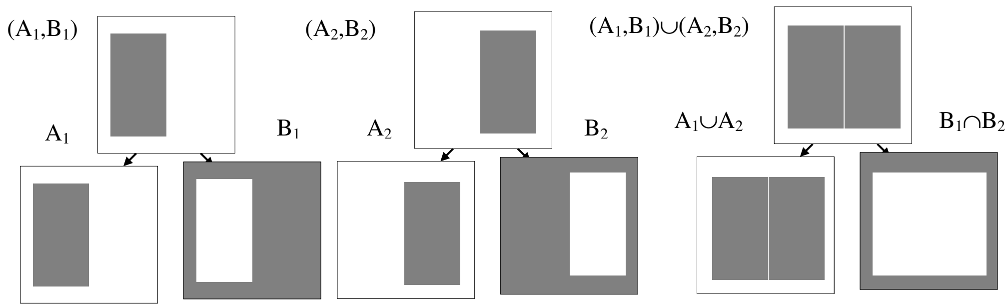

Figure 3: The union operation on the solid domain.

Theorem 3.1 *The following maps are continuous:*

(i) The predicates, $- \in - : \mathbf{U}X \times \mathbf{S}X \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$ *and* $- \in - : \mathbf{I}\mathbb{R}^d \times \mathbf{S}\mathbb{R}^d \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$.

(ii) The binary union $- \cup - : \mathbf{S}X \times \mathbf{S}X \to \mathbf{S}X$ *and more generally the n-ary union* $\bigcup : (\mathbf{S}X)^n \to \mathbf{S}X$ *for any topological space X.*

(iii) The binary intersection $- \cap - : \mathbf{S}X \times \mathbf{S}X \to \mathbf{S}X$ *and more generally the n-ary intersection* $\bigcap : (\mathbf{S}X)^n \to \mathbf{S}X$ *for any topological space X.*

(iv) $- \subseteq - : \mathbf{S}_b X \times \mathbf{S}X \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$, *for any Hausdorff space X.*

Proof (i) The proof is similar in both cases. A function of two variables on domains is continuous iff it is continuous in each variable separately when the other variable is fixed [2, page 12]. From this, we obtain the required continuity, in both cases, by observing that a non-empty compact set is contained in the union of an increasing sequence of open sets iff it is contained in one such open set.

(ii) This follows from the distributivity of $\cup$ over $\cap$.

(iii) Follows from (ii) by duality.

(iv) The function $\sqsubseteq$ is clearly monotone. To see that it is continuous in the first argument when the second argument (C, D) is held fixed, let $(A_i, B_i)_{i\in I}$ be a directed family in $\mathbf{S}_b X$. Then, $(\bigcup_{i\in I} B_i) \cup C = X$ iff $\bigcap_{i\in I} B_i^c \subseteq C$ iff there exists $i \in I$ such that $B_i^c \subseteq C$, by compactness of B_i^c [8, page 226]. On the other hand $(\bigcup_{i\in I} A_i) \cap D \neq \emptyset$ iff there exists $i \in I$ such that $A_i \cap D \neq \emptyset$. To show that it is continuous in the second argument when the first argument (A, B) is held fixed, let $(C_i, D_i)_{i\in I}$ be a directed family in $\mathbf{S}X$. Then, $B \cup \bigcup_{i\in I} C_i = X$ iff $B^c \subseteq \bigcup_{i\in I} C_i$ iff there exists $i \in I$ such that $B^c \subseteq C_i$, by compactness of B^c. Moreover, $A \cap (\bigcup_{i\in I} D_i) \neq \emptyset$ iff there exists $i \in I$ such that $A \cap D_i \neq \emptyset$. $\square$

We can also show the stability of classical solids under Boolean operations:

Theorem 3.2 *In any topological space, classical solid objects are stable under the Boolean operations.*

Proof We show that $- \cap -$ preserves classical solids; the case of $- \cup -$ follows by duality. Let (A_1, B_1) and (A_2, B_2) be two classical solids of the topological space X, so that $\overline{A_i} \cup \overline{B_i} = X$ for $i = 1, 2$. To show that $\overline{(A_1 \cap A_2)} \cup \overline{(B_1 \cup B_2)} = X$, assume $x \in X$, with $x \notin \overline{B_1 \cup B_2}$. Then, there exist open neighbourhoods D_i of x with $D_i \cap \overline{B_i} = \emptyset$, $i = 1, 2$. Hence, $D_1 \cap D_2 \subseteq \overline{A_1} \cap \overline{A_2}$. Let O be any neighbourhood of x. We will show that $O \cap A_1 \cap A_2 \neq \emptyset$. Put $D = O \cap D_1 \cap D_2$. From $D \subseteq \overline{A_1} \cap \overline{A_2}$, it follows that there exists a non-empty open set $D' \subseteq D$ with $D' \subseteq A_1$. Since $D' \subseteq \overline{A_2}$, there exists a non-empty open set $D'' \subseteq D'$ with $D'' \subseteq A_2$. We conclude that $O \cap A_1 \cap A_2 \supset D'' \neq \emptyset$, as required. $\square$

3.1 Minkowski Sum

We now introduce the Minkowski sum operation for partial solids of $X = \mathbb{R}^d$. Recall that the Minkowski sum of two subsets $S_1, S_2 \subseteq \mathbb{R}^d$ is defined as

$$S_1 \oplus S_2 = \{x + y | x \in S_1, y \in S_2\}$$

where $x + y$ is the vector addition in $\mathbb{R}^d$. For convenience we will use the same notation $\oplus$ for the Minkowski sum on the solid domain, which is defined as a function $- \oplus - : (\mathbf{S}_b\mathbb{R}^d) \times (\mathbf{S}\mathbb{R}^d) \to \mathbf{S}\mathbb{R}^d$ by:

$$(A_1, B_1) \oplus (A_2, B_2) = ((A_1 \oplus A_2), (B_1^c \oplus B_2^c)^c).$$

Lemma 3.3 $- \oplus - : (\mathbf{S}_b\mathbb{R}^d) \times (\mathbf{S}\mathbb{R}^d) \to \mathbf{S}\mathbb{R}^d$ *is well-defined.*

Proof Since the Minkowski sum of an open set with any other set is always open, $A_1 \oplus A_2$ is open. We show that the Minkowski sum $K \oplus L$ of any compact set K and any closed set L is always closed. It is sufficient to show that $K \oplus L$ contains all its limit points. Let $x_n + y_n$ with $x_n \in K$ and $y_n \in L$ be a convergent sequence with limit $z \in \mathbb{R}^d$. Since K is compact, x_n has a convergent subsequence x_{n_k} with limit $a \in K$. Since the subsequence $x_{n_k} + y_{n_k}$ converges to z, it follows that

$\lim_{k\to\infty} x_{n_k} = z - a \in L$, as L is closed. Hence $z = a + (z-a) \in K \oplus L$ and, thus, $K \oplus L$ is closed. Since, by assumption, B_1^c is compact, we conclude that $(B_1^c \oplus B_2^c)^c$ is open. It remains to show that $(A_1 \oplus A_2)$ and $(B_1^c \oplus B_2^c)^c$ are disjoint. This follows easily as $A_1 \subseteq B_1^c$ and $A_2 \subseteq B_2^c$ implies $(A_1 \oplus A_2) \subseteq (B_1^c \oplus B_2^c)$. $\square$

Corollary 3.4 *The Minkowski sum operation restricts to a map:*

$$- \oplus - : (\mathbf{S}_b\mathbb{R}^d) \times (\mathbf{S}_b\mathbb{R}^d) \to \mathbf{S}_b\mathbb{R}^d.$$

Proof This follows immediately from the fact that the Minkowski sum of two compact sets is bounded as well as closed. $\square$

Note that the Minkowski sum of two closed sets is not necessarily closed; for example, in $\mathbb{R}^2$, the set $S = \{(x,y)|y \geq 0\} \oplus \{(x,y)|y \geq \exp x\}$ is not closed as the sequence $(n,0) + (-n, \exp(-n)) = (0, \exp(-n))$ converges to $(0,0) \notin S$. That is why we need to restrict the second argument of the Minkowski operator to $\mathbf{S}_b\mathbb{R}^d$.

Theorem 3.5 *The map* $- \oplus - : (\mathbf{S}_b\mathbb{R}^d) \times (\mathbf{S}\mathbb{R}^d) \to \mathbf{S}\mathbb{R}^d$ *is continuous.*

Proof Clearly $\oplus$ is monotonic in the first argument and also, because of two complementation operations, in the second argument. We check the continuity in the first argument when the second is fixed as (C,D). Let $(A_i, B_i)_{i\in\omega}$ be an increasing chain of partial solids with lub (A,B). We have to show the following two relations: $\bigcup_{i\in\omega}(A_i \oplus C) \supseteq (\bigcup_{i\in\omega} A_i) \oplus C$ and $\bigcup_{i\in\omega}(B_i^c \oplus D^c)^c \supseteq ((\bigcup_{i\in\omega} B_i)^c \oplus D^c)^c$. The first is trivial; as for the second we need to show that: $\bigcap_{i\in\omega}(B_i^c \oplus D^c) \subseteq (\bigcap_{i\in\omega} B_i^c) \oplus D^c$. Let $z \in \bigcap_{i\in\omega}(B_i^c \oplus D^c)$. Then, for each natural number i, there exists $x_i \in B_i^c$ and $y_i \in D^c$ such that $z = x_i + y_i$. Since B_0^c is compact, there exists a subsequence $(x_{i_n})_{n\in\omega}$ which converges to $x \in B^c$. Hence $(y_{i_n})_{n\in\omega}$ converges to $z - x$ which must belong to D^c. Therefore, $z = x + (z-x) \in (\bigcap_i B_i^c) \oplus D^c$. The continuity of $- \oplus -$ when the first argument is fixed is proved in a similar way. $\square$

Unlike the two Boolean operations, the Minkowski operation does not preserve classical solid objects. For example, in $\mathbf{S}[0,4]$

$$([0,1),(1,4]) \oplus (\emptyset, [0,4] \setminus \{2\}) = (\emptyset, [0,2) \cup (3,4]),$$

which is not a classical solid. However, we have the following:

Proposition 3.6 *The map* $- \oplus - : (\mathbf{S}_b\mathbb{R}^d) \times (\mathbf{S}\mathbb{R}^d) \to \mathbf{S}\mathbb{R}^d$ *takes any two maximal elements to a classical solid.*

Proof Let $(A,B) \in \mathbf{S}_b\mathbb{R}^d$ and $(C,D) \in \mathbf{S}\mathbb{R}^d$ be maximal elements. Then, $B^c = \overline{A}$ and $D^c = \overline{C}$. We show that $\overline{A \oplus C} = \overline{A} \oplus \overline{C}$. Since $B^c \oplus D^c$ is closed, we have $\overline{A \oplus C} \subseteq \overline{A} \oplus \overline{C}$. On the other hand, let $a + c \in \overline{A} \oplus \overline{C}$. Then, there are sequences $(a_n)_{n\in\omega}$ and $(c_n)_{n\in\omega}$, with $a_n \in A$ and $c_n \in C$, for all $n \in \omega$, such that $a = \lim a_n$ and $c = \lim c_n$. Therefore, $a + c = \lim a_n + \lim c_n = \lim(a_n + c_n) \in \overline{A \oplus C}$. It follows that $B^c \oplus D^c = \overline{A \oplus C}$, and we conclude that $(A,B) \oplus (C,D) = (A \oplus C, (\overline{A \oplus C})^c)$ is a classical solid. $\square$

4 Computability on the Solid Domain

Let X be a second countable locally compact Hausdorff space. Then $\mathbf{U}X$ and $\mathbf{S}X$ are both ω-continuous bounded complete dcpo's. We will now define effective structures for $\mathbf{U}X$ and $\mathbf{S}X$. Let $\mathcal{O}$ be a countable basis of regular open sets with compact closure for X, which contains the empty set, is closed under regularized binary unions and under binary intersections. Consider an effective enumeration, i.e. a surjection, $O : \mathbb{N} \to \mathcal{O}$, such that there is an effective procedure to obtain $O(i)$ for any $i \in \mathbf{N}$. For convenience, we write O_i for $O(i)$ and often denote the enumeration O by $(O_i)_{i\in\omega}$. Here, we assume that $O_0 = \emptyset$ and stipulate that, for any $n \geq 1$, the relation $\overline{O_i} \subseteq \bigcup_{1\leq m\leq n} O_{j_m}$ is decidable. This, by the way, means that, in the continuous lattice of open subsets of X, the way-below relation on the basis $\mathcal{O}$ is decidable. Since $O_i = \emptyset$ iff $\overline{O_i} \subseteq O_0$, it follows that the equality relation $O_i = \emptyset$ is decidable and we can assume, by redefining the enumeration O, that $O_i = \emptyset$ iff $i = 0$. Furthermore, we assume that the binary intersection and the regularized binary union of basis elements are computable, i.e. there exist two total recursive functions $\phi, \psi : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that $(\overline{O_i \cup O_j})^\circ = O_{\phi(i,j)}$ and $O_i \cap O_j = O_{\psi(i,j)}$. In particular, this implies that the relation $O_i \cap O_j = \emptyset$ is decidable.

Definition 4.1 *Let $(O_i)_{i\in\omega}$ be an effective enumeration of a basis of a second countable locally compact Hausdorff space X, consisting of regular open sets with compact closure. Assume further that the basis is closed under binary intersection and regularized binary union. We say that the effective enumeration $(O_i)_{i\in\omega}$ is an* effective structure *for X, if the following conditions hold:*

- *$O_i = \emptyset$ iff $i = 0$.*
- *For any $n \geq 1$, the relation $\overline{O_i} \subseteq \bigcup_{1\leq m\leq n} O_{j_m}$ is decidable.*
- *There exist total recursive functions $\phi, \psi : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that $(\overline{O_i \cup O_j})^\circ = O_{\phi(i,j)}$ and $O_i \cap O_j = O_{\psi(i,j)}$.*

If X is compact, we will assume the further condition that, for each positive integer n, the relation $\bigcup_{1\leq m\leq n} O_{i_m} = X$ is decidable.

Note that the closure of the basis under binary intersection and regularized union implies its closure under finite intersections and regularized finite unions. For example, if A, B and C are open sets then it is easy to check that $(\overline{A \cup B \cup C})^\circ = (\overline{(\overline{A \cup B})^\circ \cup C})^\circ$. From the effective enumeration O of the basis $\mathcal{O}$, we can obtain an effective enumeration of the basis $\mathcal{S}$ of $\mathbf{S}X$, consisting of pairs of disjoint elements of $\mathcal{O}$. In fact, there are total recursive functions $\alpha, \beta : \mathbb{N} \to \mathbb{N}$ such that $S : \mathbb{N} \to \mathcal{S}$, with $S_n = (O_{\alpha(n)}, O_{\beta(n)})$, gives an effective enumeration of $\mathcal{S}$, with the relation $S_i \ll S_j$ decidable.

The collection $\mathcal{C} = \{\overline{O} | O \in \mathcal{O}\} \cup \{\perp\}$ is a countable basis for the ω-continuous domain $\mathbf{UX}$, with an effective enumeration $C : \mathbb{N} \to \mathcal{C}$ defined by $C_0 = \perp$ and

$C_i = \overline{O_i}$ for $i \geq 1$. Notice that, we have $C_j \ll C_i$ in $\mathbf{U}X$ iff $\overline{O_i} \subseteq O_j$; therefore, the way-below relation, $C_i \ll C_j$, on $\mathcal{C}$ is decidable.

Having equipped $\mathbf{S}X$ and $\mathbf{U}X$ with the above effective structure, we can now deduce the computability of the basic predicates and operations.

Theorem 4.2 *The following functions are computable with respect to the effective structures on $\mathbf{U}X$ and $\mathbf{S}X$.*

(i) The membership predicate, $- \in - : \mathbf{U}X \times \mathbf{S}X \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$.

(ii) The binary union $-\cup- : \mathbf{S}X \times \mathbf{S}X \to \mathbf{S}X$ *and more generally the n-ary union* $\cup : (\mathbf{S}X)^n \to \mathbf{S}X$.

(iii) The binary intersection, $-\cap- : \mathbf{S}X \times \mathbf{S}X \to \mathbf{S}X$ *and more generally the n-ary intersection* $\cap : (\mathbf{S}X)^n \to \mathbf{S}X$.

(iv) $- \subseteq - : \mathbf{S}X \times \mathbf{S}X \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$, *where X is assumed to be compact.*

Proof (i) We have to show that the relations $(C_i \in S_n) = \mathsf{tt}$ and $(C_i \in S_n) = \mathsf{ff}$ are both r.e. The first reduces to $C_i \subseteq O_{\alpha(n)}$, in other words, $\overline{O_i} \subseteq O_{\alpha(n)}$, which is in fact decidable by assumption. The second is similarly decidable.

(ii) We have to show, in the binary case, that the relation $S_n \ll S_i \cup S_j$ is r.e. Writing this relation in detail, it reduces to $(O_{\alpha(n)}, O_{\beta(n)}) \ll (O_{\alpha(i)} \cup O_{\alpha(j)}, O_{\psi(\beta(i),\beta(j))})$, i.e. $\overline{O_{\alpha(n)}} \subseteq O_{\alpha(i)} \cup O_{\alpha(j)}$ and $\overline{O_{\beta(n)}} \subseteq O_{\psi(\beta(i),\beta(j))}$, which are both decidable. The n-ary case is similar.

(iii) Dual to (ii).

(iv) The relations $(S_i \subseteq S_j) = \mathsf{tt}$ and $(S_i \subseteq S_j) = \mathsf{ff}$ reduce to $O_{\beta(i)} \cup O_{\alpha(j)} = X$ and $O_{\psi(\alpha(i),\beta(j))} \neq \emptyset$, which are both decidable. $\square$

4.1 Effective structure over $\mathbf{S}\mathbb{R}^d$

In order to endow $\mathbf{S}\mathbb{R}^d$ with an effective structure, we introduce two different countable bases that are recursively equivalent, but correspond to different types of algorithms in use. The first basis, made of partial dyadic voxel sets, corresponds to the discrete geometry approach, while the second one, made of partial rational polyhedra, is more consistent with the computational geometry point of view and will be the basis for efficient algorithms. The computability of Boolean and Minkowski operators is easier to prove using the partial dyadic voxel sets representation.

4.1.1 Partial dyadic voxel sets

A *dyadic number* is a rational number whose denominator is a power of 2. Given a natural number n, we divide the cube $[-2^n, 2^n]^d$ into $2^{(2n+1)d}$ small cubes each of length 2^{-n}, the coordinates of the 2^d vertices of each small cube will then be integer multiples of 2^{-n}, that is, dyadic numbers. We consider these small cubes as closed cubes: two adjacent cubes overlap along their common face (or k-edge, $0 \leq k \leq d-1$).

A *dyadic voxel set of order* n is the interior of a finite union of these small cubes. We have then $2^{2^{(2n+1)d}}$ distinct dyadic voxel sets of order n, including the trivial ones, that is the empty set and the whole cube $[-2^n, 2^n]^d$ itself. Notice that dyadic voxel sets of order n are regular open sets with compact closure(Fig. 4.

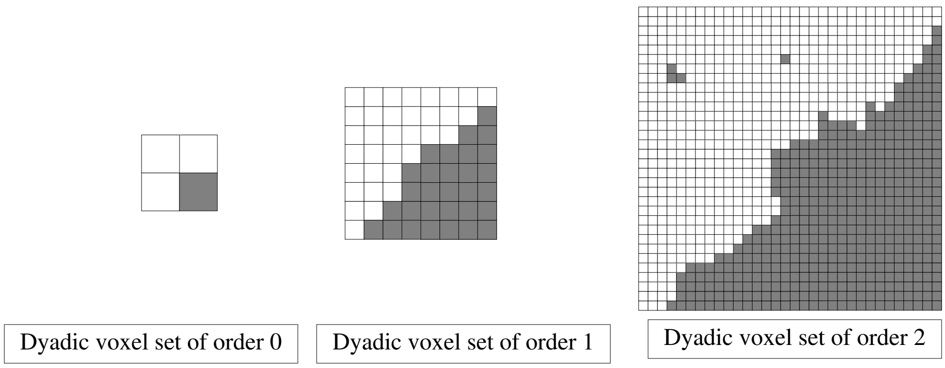

Figure 4: Voxel sets of different orders.

Of course, if $n < m$, the dyadic voxel sets of order n are dyadic voxel sets of order m. We say that a voxel set V has *strict order* n if V is of order n but is not of order k for $k < n$.

The set $\mathcal{V}$ of all the dyadic voxel sets of any order $n = 0, 1, 2, ..$ can be effectively enumerated by $V : \mathbb{N} \to \mathcal{V}$ as follows. We put $V_0 = \emptyset$ and then start by first enumerating, in a given prescribed way, the dyadic voxel sets of strict order 0, then those of strict order 1, and so on. Then, there exists a total recursive function $r : \mathbb{N} \to \mathbb{N}$ such that, for each $i \in \mathbb{N}$, V_i will be a voxel set of strict order $r(i)$, which would be explicitly given as the interior of the union of its small cubes.

Clearly, $V_i = \emptyset$ iff $i = 0$ and the relations $\overline{V_i} \subseteq \bigcup_{1 \leq m \leq n} V_{j_m}$ and $V_i \subseteq \bigcup_{1 \leq m \leq n} V_{j_m}$ are decidable for each $n \geq 1$.

The intersection and the regularized union of dyadic voxel sets of order n are dyadic voxel sets of order n and computing the index of the binary intersection and the regularized binary union of dyadic voxel sets from their indices is a finite procedure. This therefore gives an effective structure for $\mathbb{R}^n$ in the sense of Definition 4.1. From the effective enumeration $(V_i)_{i \in \omega}$ one can construct an effective enumeration $(\mathbb{V}_i)_{i \in \omega}$ of the *partial dyadic voxel sets*, that is the pairs $\mathbb{V}_i = (V_{\alpha(i)}, V_{\beta(i)})$, with α and β total recursive functions, such that $V_{\alpha(i)} \cap V_{\beta(i)} = \emptyset$, with $\mathbb{V}_0 = (\emptyset, \emptyset)$. Then, $(\mathbb{V}_i)_{i \in \omega}$ provides us with a basis of $\mathbf{S}\mathbb{R}^d$ and a partial solid $(A, B) \in \mathbf{S}\mathbb{R}^d$ is computable if and only if the set $\{i \in \mathbb{N} | \mathbb{V}_i \ll (A, B)\}$ is r.e. We can endow $\mathbf{S}[-a, a]^d$, where $a > 0$ is a computable real number, with an effective structure by using the intersection of voxel sets with the cube $\mathbf{S}[-a,a]^d$.

One can then apply the results of the previous section to deduce:

Corollary 4.3 *The following functions are computable with respect to the effective structures on $\mathbf{I}\mathbb{R}^d$, $\mathbf{S}\mathbb{R}^d$ and $\mathbf{S}[-a,a]^d$.*

(i) $- \in - : \mathbf{I}\mathbb{R}^d \times \mathbf{S}\mathbb{R}^d \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$

(ii) $- \cup - : \mathbf{S}\mathbb{R}^d \times \mathbf{S}\mathbb{R}^d \to \mathbf{S}\mathbb{R}^d$.

(iii) $- \cap - : \mathbf{S}\mathbb{R}^d \times \mathbf{S}\mathbb{R}^d \to \mathbf{S}\mathbb{R}^d$.

(iv) $- \subseteq - : \mathbf{S}[-a,a]^d \times \mathbf{S}[-a,a]^d \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$.

In order to study the computability of the Minkowski sum, we need a basis for the domain $\mathbf{S}_b\mathbb{R}^d$ of bounded partial solids. Recall that the non-bottom elements of $\mathbf{S}_b\mathbb{R}^d$ are of the form (A, B), with A and B open and B^c bounded, and therefore compact. The second component, B, will be approximated by the interiors of complements of dyadic voxel sets.

From the effective enumeration $(V_i)_{i\in\omega}$ one can obtain an effective enumeration $(\mathbb{W}_i)_{i\in\omega}$ of the *partial bounded dyadic voxel sets.* There are total recursive functions γ and δ such that $\mathbb{W}_0 = (\emptyset, \emptyset)$ and, for $i > 0$, $\mathbb{W}_i = (V_{\gamma(i)}, {V^c_{\delta(i)}}^\circ)$ where $V_{\gamma(i)} \subseteq V_{\delta(i)}$, which is decidable. This provides us with a basis for $\mathbf{S}_b\mathbb{R}^d$.

Proposition 4.4 *Given basis elements $\mathbb{W}_i$ and $\mathbb{V}_j$ of $\mathbf{S}_b\mathbb{R}^d$ and $\mathbf{S}\mathbb{R}^d$ respectively, there is a total recursive function $\phi : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that $\mathbb{V}_{\phi(i,j)} = \mathbb{W}_i \oplus \mathbb{V}_j$. Given basis elements $\mathbb{W}_i$ and $\mathbb{W}_j$ of $\mathbf{S}_b\mathbb{R}^d$, there is a total recursive function ψ such that $\mathbb{W}_{\psi(i,j)} = \mathbb{W}_i \oplus \mathbb{W}_j$.*

Proof The computation reduces to computing either the Minkowski sum of two dyadic voxel sets, or the Minkowski sum of a dyadic voxel set and the complement of a dyadic voxel set. This is clearly a finite procedure. $\Box$

Corollary 4.5 *The following maps are computable.*

- $- \oplus - : (\mathbf{S}_b\mathbb{R}^d) \times (\mathbf{S}\mathbb{R}^d) \to \mathbf{S}\mathbb{R}^d$
- $- \oplus - : (\mathbf{S}_b\mathbb{R}^d) \times (\mathbf{S}_b\mathbb{R}^d) \to \mathbf{S}_b\mathbb{R}^d$
- $- \subseteq - : (\mathbf{S}_b\mathbb{R}^d) \times (\mathbf{S}\mathbb{R}^d) \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$.

4.1.2 Partial rational polyhedra

A *rational d-simplex* in $\mathbb{R}^d$ is the convex hull of $d+1$ points with rational coordinates that do not lie on the same hyper-plane. An *open rational polyhedron* is the interior of a finite union of rational d-simplexes. Starting with an effective enumeration of the rational d-simplexes, one can obtain an effective enumeration $(P_i)_{i\in\omega}$ of the set of open rational polyhedra with $P_i = \emptyset$ iff $i = 0$. The relations $\overline{P_i} \subseteq \bigcup_{1\leq m\leq n} P_{j_m}$ and $P_i \subseteq \bigcup_{1\leq m\leq n} P_{j_m}$ are decidable for each $n \geq 1$. Rational polyhedra are closed under the binary intersection and the regularized binary union. These operations are computable as they rely only on rational arithmetic and comparison of rational numbers.

A *partial open rational polyhedron* is a pair of disjoint open rational polyhedra. From the effective enumeration $(P_i)_{i\in\omega}$ of open rational polyhedra, one can obtain an effective enumeration $(\mathbb{P}_i)_{i\in\omega}$ of the partial open rational polyhedra.

Partial dyadic voxel sets are trivially partial open rational polyhedra. Moreover, they define the same notion of computability, in other words:

Proposition 4.6 *$\mathbb{P}_i \ll \mathbb{V}_j$ and $\mathbb{V}_i \ll \mathbb{P}_j$ are decidable in i and j.*

From this equivalence, it follows that a partial solid object, or a map, is computable with respect to the effective structure by partial open rational polyhedra if and only if it is computable with respect to the effective structure by partial dyadic voxel sets.

One can define a basis of $\mathbf{S}_b\mathbb{R}^d$ exactly in the same way as with rational partial dyadic voxel sets.

Our domain-theoretic notion of computability so far has the essential weakness of lacking a quantitative measure for the rate of convergence of basis elements to a computable element. This shortcoming can be redressed by enriching the domain-theoretic notion of computability with an additional requirement which allows a quantitative degree of approximation. We will see in the next two sections that this can be done in at least two different ways.

5 Hausdorff computability

In this section we will enrich the notion of computability with convergence with respect to the Hausdorff metric. Let X be a compact metric space, with its solid domain $\mathbf{S}X$ effectively given by a basis $(S_i)_{i\in\omega}$, with $S_i = (O_{\alpha(i)}, O_{\beta(i)})$. Let d_H denote the Hausdorff distance between compact sets with the convention that $d_H(\emptyset,\emptyset) = 0$ and for $Y \neq \emptyset$, $d_H(\emptyset, Y) = \infty$. We assume that the three double sequences $(d_H(\overline{O_i}, \overline{O_j}))_{i,j\in\omega}$, $(d_H(\overline{O_i}, O_j^c))_{i,j\in\omega}$ and $(d_H(O_i^c, O_j^c))_{i,j\in\omega}$ of real numbers are computable.

Definition 5.1 *A partial solid (A, B) is* Hausdorff computable *if there is a total recursive function f such that:*

- $A = \bigcup_{i \in \omega} O_{\alpha(f(i))}$ *with* $d_H(\overline{A}, \overline{O_{\alpha(f(i))}}) < 2^{-i}$ *and* $d_H(A^c, O^c_{\alpha(f(i))}) < 2^{-i}$.

- $B = \bigcup_{i \in \omega} O_{\beta(f(i))}$ *with* $d_H(\overline{B}, \overline{O_{\beta(f(i))}}) < 2^{-i}$ *and* $d_H(B^c, O^c_{\beta(f(i))}) < 2^{-i}$.

Lemma 5.2 *Let* $(A_i)_{i \in \omega}$ *be a decreasing sequence of compact subsets of a compact metric space* X *and* $\bigcap_{i \in \omega} A_i = A$. *Then* $d_H(A_i, A) \to 0$ *and* $d_H(\overline{A^c_i}, \overline{A^c}) \to 0$.

Proof Let B_ϵ be the open ball of radius ϵ centred at the origin. Consider

$$A \oplus B_\epsilon = \{x \in X | \exists a \in A.\ d(x, a) < \epsilon\}.$$

Then, there exists $i \in \omega$ such that $A_i \subseteq A \oplus B_\epsilon$ [8, page 226]. It follows that $d_H(A_i, A) \leq \epsilon$. Furthermore, we have $\overline{A^c} \subseteq \bigcup_{i \in \omega} A^c_i \oplus B_\epsilon$. It follows that there exists $i \in \omega$ such that $\overline{A^c} \subseteq A^c_i \oplus B_\epsilon$, and hence, $d_H(\overline{A^c_i}, \overline{A^c}) \leq \epsilon$. $\Box$

Proposition 5.3 *A computable maximal element of* $\mathbf{S}X$ *is Hausdorff computable.*

Proof Let (A, B) be a computable maximal element of $\mathbf{S}X$. From the regularity of A and B, we get $\overline{A} = B^c$ and $\overline{B} = A^c$. From the computability of (A, B) in $\mathbf{S}X$, it follows that there exists a total recursive function ρ such that $(A, B) = (\bigcup_{i \in \omega} O_{\alpha \circ \rho(i)}, \bigcup_{i \in \omega} O_{\beta \circ \rho(i)})$, where the sequences of basis elements are both increasing with i. For convenience, put $A_i = O_{\alpha \circ \rho(i)}$ and $B_i = O_{\beta \circ \rho(i)}$. We have, $A^c = \bigcap_{i \in \omega} A^c_i$ and $B^c = \bigcap_{i \in \omega} B^c_i$. Since X is compact, A^c, A^c_i, B^c and B^c_i are also compact.

Applying Lemma 5.2, we get: $d_H(A^c_i, A^c) \to 0$, $d_H(\overline{A_i}, \overline{A}) \to 0$, $d_H(B^c_i, B^c) \to 0$, and $d_H(\overline{B_i}, \overline{B}) \to 0$. Using $\overline{A} = B^c$ and the triangular inequality we deduce: $d_H(\overline{A_i}, B^c_i) \to 0$ and similarly $d_H(\overline{B_i}, A^c_i) \to 0$.

Since α, β and ρ are total recursive functions, $(d_H(\overline{A_i}, B^c_i))_{i \in \omega}$, i.e. $(d_H(O_{\alpha \circ \rho(i)}, O^c_{\beta \circ \rho(i)}))_{i \in \omega}$ and $(d_H(\overline{B_i}, A^c_i))_{i \in \omega}$, i.e. $(d_H(\overline{O_{\beta \circ \rho(i)}}, O^c_{\alpha \circ \rho(i)}))_{i \in \omega}$, are computable sequences of real numbers. Therefore, we can effectively find the first integer $k(i) \in \mathbb{N}$ such that: $d_H(\overline{A_{k(i)}}, B^c_{k(i)}) < 2^{-i}$ and $d_H(\overline{B_{k(i)}}, A^c_{k(i)}) < 2^{-i}$. Now, given three subsets E, F and G, with $E \subseteq F \subseteq G$, we can check easily that: $d_H(E, F) \leq d_H(E, G)$ and $d_H(F, G) \leq d_H(E, G)$. Applying this to $\overline{A_{k(i)}} \subseteq \overline{A} = B^c \subseteq B^c_{k(i)}$ and $\overline{B_{k(i)}} \subseteq \overline{B} = A^c \subseteq A^c_{k(i)}$, it follows that:

$$d_H(\overline{B_{k(i)}}, \overline{B}) < 2^{-i}, \quad d_H(B^c, B^c_{k(i)}) < 2^{-i}, \quad d_H(\overline{A_{k(i)}}, \overline{A}) < 2^{-i}, \quad d_H(A^c, A^c_{k(i)}) < 2^{-i},$$

which completes the proof. $\Box$

From the definition, it is clear that the complement (B, A) of a partial solid (A, B) is Hausdorff computable if and only if (A, B) is Hausdorff computable. However, Boolean operators do not preserve Hausdorff computability in general, as we will show in the following example(Fig. 5).

Example 5.4 *We will construct Hausdorff computable maximal elements (A, B) and (A', B') of $\mathbf{S}([0,1]\times[-1,1])$ which have a non-Hausdorff computable intersection. Let $(a_n)_{n\in\omega}$ be a computable, increasing sequence of rational numbers, with $a_0 > 0$, whose limit is a non-computable, left-computable real number $l < 1$.*

Let $g_n : [0,1] \to [0,1]$, for $n \in \mathbb{N}$, be defined by

$$g_n(t) = \begin{cases} 2^{-n}(1 - \frac{t}{a_n}) & \text{if } t < a_n \\ 0 & \text{if } t \geq a_n \end{cases}$$

Then, put $f_n = \max\{g_i | 0 \leq i \leq n\}$, $A_n = \{(x,y) \in [0,1] \times [-1,1] | f_n(x) + 2^{-n} < y\}$ and $B_n = \{(x,y) \in [0,1] \times [-1,1] | y < f_n(x)\}$.

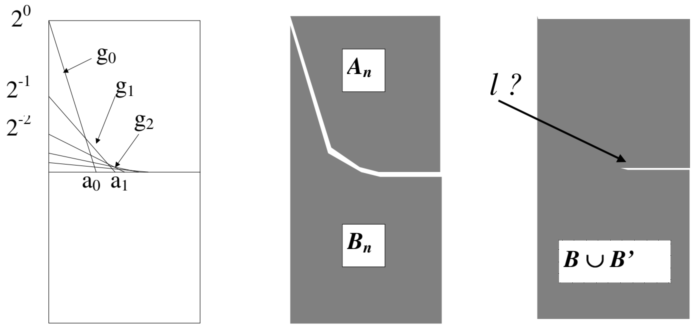

Figure 5: Intersection does not preserve Hausdorff computability.

The sets $A = \bigcup_{i\in\omega} A_i$ and $B = \bigcup_{i\in\omega} B_i$ are regular and (A, B) is a Hausdorff computable, maximal element of $\mathbf{S}([0,1] \times [-1,1])$. The partial solid (A', B') with $A' = [0,1] \times [-1,0)$ and $B' = \overline{A'}^c = [0,1] \times (0,1]$ is Hausdorff computable. Consider the intersection $(A, B) \cap (A', B') = (A \cap A', B \cup B')$. We have $A \cap A' = \emptyset$ and $B \cup B' = ([l,1] \times \{0\})^c$.

If the last component were Hausdorff computable, there would be a computable sequence of basis elements $(X_i)_{i\in\omega}$ such that $d_H(X_i^c, [l,1] \times \{0\}) < 2^{-i}$. But this is in contradiction with the non-computability of l.

6 Lebesgue Computability

We now consider the notion of measure-theoretic computability which is closed under Boolean operations and can be expressed for solids on locally compact spaces as well. Suppose we have the effective structure, introduced in Section 4, on the solid domain $\mathbf{S}X$ of a second countable locally compact space X, given in terms of the countable basis $\mathcal{O}$. Let μ be a finite Borel measure on X, such that $(\mu(O_i))_{i\in\omega}$

is a computable sequence of real numbers. If $(A, B) \in \mathbf{S}X$ is computable then $(A, B) = \bigsqcup_{i\in\omega} S_{\delta(i)} = \bigsqcup_{i\in\omega}(O_{\alpha(\delta(i))}, O_{\beta(\delta(i))})$ for a total recursive function $\delta : \mathbb{N} \to \mathbb{N}$ such that $(S_{\delta(i)})_{i\in\omega}$ is an increasing chain. It follows that $(\mu(O_{\alpha(\delta(i))}))_{i\in\omega}$ and $(\mu(O_{\beta(\delta(i))}))_{i\in\omega}$ are computable increasing sequences of real numbers which converge to $\mu(A)$ and $\mu(B)$, respectively. Hence, $\mu(A)$ and $\mu(B)$ are left-computable real numbers. We say that the computable partial solid (A, B) is *μ-computable* if $\mu(A)$ and $\mu(B)$ are both computable real numbers. It follows that, (A, B) is μ-computable iff there exists a total recursive function δ such that $\mu(A) - \mu(O_{\alpha(\delta(i))}) < \frac{1}{2^i}$ and $\mu(B) - \mu(O_{\beta(\delta(i))}) < \frac{1}{2^i}$, for all $i \in \mathbb{N}$. The definition extends naturally to computable elements of $(\mathbf{S}X)^m$ for any positive integer m.

Proposition 6.1 *If $\mu(X)$ is a computable real number and $(A, B) \in \mathbf{S}X$ is computable with $\mu(X\setminus(A\cup B))$ a left-computable real number, then (A, B) is μ-computable.*

Proof We have the disjoint union $X = A \cup B \cup (X \setminus (A \cup B))$. Since $\mu(B)$ and $\mu(X\setminus(A\cup B))$ are left-computable, it follows that $\mu(A) = \mu(X)-\mu(B)-\mu(X\setminus(A\cup B))$ is also right-computable, and, hence, computable. Similarly $\mu(B)$ is computable. □

Corollary 6.2 *If $\mu(X)$ is a computable real number and $(A, B) \in \mathbf{S}X$ is a computable maximal element with $\mu(\partial A) = 0$, then (A, B) is μ-computable.*

Proof By Lemma 2.5, $\partial A = \partial B$. Hence $\mu(X \setminus (A \cup B)) = \mu(\partial A) = 0$. □

We say that a computable sequence of partial solids $((A_n, B_n))_{n\in\omega}$ is *μ-computable* if $(\mu(A_n))_{n\in\omega}$ and $(\mu(B_n))_{n\in\omega}$ are computable sequences of real numbers. As for computable elements, the definition extends naturally to computable sequences of $(\mathbf{S}X)^m$ for any positive integer m. If $((A_n, B_n))_{n\in\omega}$ is a computable sequence of partial solid objects, then there exist total recursive functions a and b with $(A_n, B_n) = \bigsqcup_{i\in\omega}(O_{a(n,i)}, O_{b(n,i)})$ where the sequences of open sets are increasing with i.

Lemma 6.3 *Suppose $((A_n, B_n))_{n\in\omega}$ is a computable sequence of partial solids, with $(A_n, B_n) = \bigsqcup_{i\in\omega}(O_{a(n,i)}, O_{b(n,i)})$ for total recursive functions a and b where the sequences of open sets are increasing with i. Then, $((A_n, B_n))_{n\in\omega}$ is μ-computable iff there exist total recursive functions $r, s : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that*

$$\mu(A_n) - \mu(O_{a(n,r(n,i))}) \leq 2^{-i}, \qquad \mu(B_n) - \mu(O_{b(n,s(n,i))}) \leq 2^{-i}.$$

Proof Since $(\mu(O_j))_{j\in\omega}$ is a computable sequence of real numbers, it follows that $(\mu(O_{a(n,i)}))_{n,j\in\omega}$ is a computable double sequence of real numbers. Since $(\mu(O_{a(n,i)})))_{n,j\in\omega}$ converges monotonically upwards to the sequence $(\mu(A_n))_{n\in\omega}$ as $j \to \infty$, it follows by [26, Proposition 2, Page 20], that the convergence is effective in both n and j, i.e. the recursive function r, with the required property exists, iff $(\mu(A_n))_{n\in\omega}$ is a computable sequence of real numbers. Similarly, the recursive function s with the required property exists iff $(\mu(B_n))_{n\in\omega}$ is a computable sequence of real numbers. □

A computable function $f : (\mathbf{S}X)^m \to \mathbf{S}X$ is said to be *μ-computable* if it takes any μ-computable sequence of m-tuples of partial solids to a μ-computable sequence of partial solids.

Theorem 6.4 *The binary operations $-\cup-$ and $-\cap-$ are μ-computable. More generally, the n-ary operations of $\cup$ and $\cap$ are μ-computable.*

Proof Let $((A_n, B_n))_{n\in\omega}$ and $((C_n, D_n))_{n\in\omega}$ be μ-computable sequences of partial solids with $(A_n, B_n) = \bigsqcup_{i\in\omega}(O_{a(n,i)}, O_{b(n,i)})$ and $(C_n, D_n) = \bigsqcup_{i\in\omega}(O_{c(n,i)}, O_{d(n,i)})$, where the sequences of open sets are increasing and a, b, c and d are total recursive functions.

Since, by Theorem 4.2, $-\cup-$ is computable, it sends computable sequences to computable sequences. Hence, $((A_n \cup C_n, B_n \cap D_n))_{n\in\omega}$ is a computable sequence of partial solids. We show that $(\mu(A_n \cup C_n))_{n\in\omega}$ and $(\mu(B_n \cap D_n))_{n\in\omega}$ are computable sequences of real numbers.

Let r and s be total recursive functions, given by Lemma 6.3, such that

$$\mu(A_n) - \mu(O_{a(n,r(n,i))}) \leq 2^{-i}, \qquad \mu(C_n) - \mu(O_{c(n,s(n,i))}) \leq 2^{-i}.$$

Then, for the total recursive function u defined by $u(n,i) = \max(r(n,i), s(n,i))$, we have:

$$\begin{array}{rcl}
\mu(A_n \cup C_n) - \mu(O_{a(n,u(n,i))} \cup O_{c(n,u(n,i))}) & = & \mu((A_n \cup C_n) \setminus (O_{a(n,u(n,i))} \cup O_{c(n,u(n,i))})) \\
& \leq & \mu((A_n \setminus O_{a(n,u(n,i))}) \cup (C_n \setminus O_{c(n,u(n,i))})) \\
& \leq & \mu(A_n \setminus O_{a(n,u(n,i))}) + \mu(C_n \setminus O_{c(n,u(n,i))}) \\
& \leq & 2^{-i+1}
\end{array}$$

We have:

$$\mu(O_{a(n,u(n,i))} \cup O_{c(n,u(n,i))}) = \mu(O_{a(n,u(n,i))}) + \mu(O_{c(n,u(n,i))}) - \mu(O_{a(n,u(n,i))} \cap O_{c(n,u(n,i))}).$$

Since $O_{a(n,u(n,i))} \cap O_{c(n,u(n,i))} = O_{\psi(a(n,u(n,i)),c(n,u(n,i)))}$, it follows that

$$\mu(O_{a(n,u(n,i))} \cap O_{c(n,u(n,i))})_{n,i\in\omega}$$

is a computable double sequence of real numbers. Therefore,

$$(\mu(O_{a(n,u(n,i))} \cup O_{c(n,u(n,i))}))_{n,i\in\omega}$$

is the linear sum of three computable double sequences of real numbers. Hence, $(\mu(O_{a(n,u(n,i))} \cup O_{c(n,u(n,i))}))_{n,i\in\omega}$ is itself a computable double sequence of real numbers, which converges, as $i \to \infty$, to $(\mu(A_n \cup C_n))_{n\in\omega}$ effectively in i and n, as the above calculation shows. Therefore, by Lemma 6.3, $(\mu(A_n \cup C_n))_{n\in\omega}$ is a computable sequence of real numbers. Similarly, $(\mu(B_n \cap D_n))_{n\in\omega}$ is a computable sequence of real numbers. This establishes the μ-computability of $-\cup-$. The case of $-\cap-$ follows by duality. The case of the n-ary operations of $\cup$ and $\cap$ is similar. $\Box$

Now suppose μ is a locally finite Borel measure, i.e. one which is finite on any compact subset of X, such that $(\mu(O_i))_{i\in\omega}$ is a computable sequence of real numbers.

We say that a computable partial solid object $(A,B) \in \mathbf{S}X$ is *μ-computable* if $\mu(A\cap O_n)_{n\in\omega}$ and $\mu(B\cap O_n)_{n\in\omega}$ are computable sequences of real numbers. The computable sequence of partial solids $((A_n,B_n))_{n\in\omega}$ is *μ-computable* if $(\mu(A_n\cap O_m))_{n,m\in\omega}$ and $(\mu(B_n\cap O_m))_{n,m\in\omega}$ are computable sequences of real numbers. These definitions extend naturally to computable elements and to sequences of elements of $(\mathbf{S}X)^k$ for any positive integer k. We say that a computable map $P : \mathbf{S}X \to \mathbf{S}Y$ is *μ-computable* if it takes any μ-computable sequence of partial solids of to a μ-computable sequence of partial solids. We say that a map $P : \mathbf{S}X^m \to \mathbf{S}X$ is *μ-computable* if it takes any μ-computable sequence of m-tuples of partial solids to a μ-computable sequence of partial solids.

Lemma 6.5 *Suppose $((A_n,B_n))_{n\in\omega}$ is computable, with $(A_n,B_n) = \bigsqcup_{i\in\omega}(O_{a(n,i)},O_{b(n,i)})$ for total recursive functions a and b where the sequences of open sets are increasing with i. Then $((A_n,B_n))_{n\in\omega}$ is μ-computable if and only if there exists total recursive functions r and s such that*

$$\mu(A_n\cap O_m)-\mu(O_{a(n,r(n,m,i))}\cap O_m) \le 2^{-i}, \quad \mu(B_n\cap O_m)-\mu(O_{b(n,s(n,m,i))}\cap O_m) \le 2^{-i}.$$

Proof Since $\mu(O_{a(n,i)}\cap O_m) = \mu(O_{\psi(a(n,i),m)})$, it follows that $(\mu(O_{a(n,i)}\cap O_m))_{n,i,m\in\omega}$ is a computable triple sequence of real numbers. This sequence converges monotonically upwards to the sequence $(\mu(A_n \cap O_m))_{n,m\in\omega}$ as $i \to \infty$. Hence, by [26, Proposition 2, Page 20], the recursive function r with the required property exists if and only if the sequence $(\mu(A_n \cap O_m))_{n,m\in\omega}$ is computable. Similarly, the recursive function s exists if and only if the sequence $(\mu(B_n \cap O_m))_{n,m\in\omega}$ is computable. $\square$

As in the case of finite measures, the binary operations $-\cup-$ and $-\cap-$, and, more generally, the n-ary operations $\cup$ and $\cap$ are μ-computable. The proof is similar to that of Theorem 6.4, this time using Lemma 6.5.

Next, we consider the most important case, namely, when μ is the Lebesgue measure λ on $\mathbb{R}^d$. We show that there are computable partial solids which are not Lebesgue computable. In fact, we will provide an example of a computable maximal element of $\mathbf{S}[-1,1]$ which is not Lebesgue computable.

Example 6.6 *This example uses a modification of a construction, due to Reinhold Heckmann, of a regular open set of the real line which has a boundary with non-zero Lebesgue measure. The construction is similar to that of the standard Cantor set except that at each stage two open intervals, rather than just one, are removed. Let $(a_n)_{n\in\omega}$ be a strictly increasing computable sequence of rational numbers $a_n > 0$ converging to the non-computable real number $a < 1$. Put $b_0 = a_0$ and $b_{n+1} = a_{n+1} - a_n$ for $n \ge 0$. Start with the closed interval $[-1,1]$ and remove two open intervals each of length b_0 such that three closed intervals of equal length are left. In each of these three closed intervals, remove two open intervals, each of length $\frac{b_1}{3}$, and so on. At the nth stage, there are 3^n closed intervals, in each we remove two* *open intervals each of length $\frac{b_n}{3^n}$, resulting in a total of 2×3^n open intervals(Fig. 6). For $1 \le m \le 3^n$, we denote by B_{nm} and C_{nm}, respectively, the left and the right open intervals removed in the mth closed interval. Let $B_n = \bigcup_{1\le m\le 3^n} B_{nm}$, $C_n = \bigcup_{1\le m\le 3^n} C_{nm}$. Finally, put $B = \bigcup_{n\in\omega} B_n$ and $C = \bigcup_{n\in\omega} C_n$. It is straightforward to check that $\lambda(B) = \lambda(C) = a$ and that B and C are regular open sets, with $B = C^{c\circ}$ and $C = B^{c\circ}$. By construction $(B, C) \in \mathbf{S}[-1, 1]$ is a computable, maximal solid object, which is not Lebesgue computable. This example can be lifted to $\mathbb{R}^d$ by taking the product of (B, C) with $[-1, 1]^{d-1}$.*

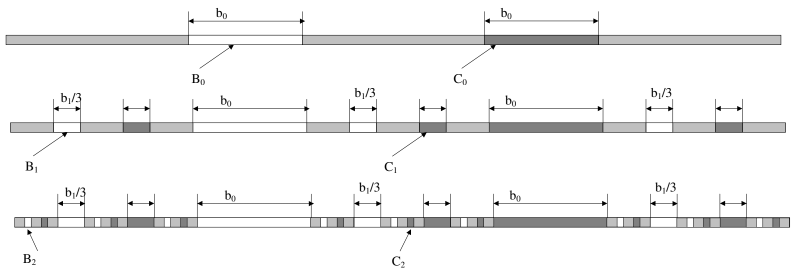

Figure 6: A non-Lebesgue computable regular solid.

One can also use a construction of a fractal Jordan curve by Ker-I-Ko and Weihrauch [20] in $\mathbb{R}^2$ to show that there is even a computable but non-Lebesgue computable maximal solid object $(B, C) \in \mathbf{S}\mathbb{R}^2$ such that the common boundary $\partial B = \partial C$ is a Jordan curve.

We conjecture that the Minkowski operation preserves Lebesgue computable maximal elements, i.e. if $(A, B), (C, D) \in \mathbf{S}[-a, a]^d$ are Lebesgue computable maximal elements then $(A, B) \oplus (C, D)$ is Lebesgue computable. However, the Minkowski operation does not preserve Lebesgue computable elements in general as the following example shows.

Example 6.7 *Let $0 < l < 1$ be a right computable, non-computable real number and consider the non-maximal element $(\emptyset, ([0, l] \times \{0\})^c) \in \mathbf{S}[-2, 2]^2$, which is Lebesgue computable. Let B_1 be the open ball of radius 1 around the origin. Then, the Minkowski sum*

$$(\emptyset, ([0, l] \times \{0\})^c) \oplus (B_1, {B_1}^{c^\circ}) = (\emptyset, (\overline{B_1} \oplus ([0, l] \times \{0\}))^c)$$

is not Lebesgue computable, since the second component has measure $16 - (\pi + 2l)$ which is not a computable real number.

A computable partial solid (A, B), with $\mu(X \setminus (A \cup B)) = 0$, can be manufactured with an error that can be made as small as we want in volume, assuming an idealized manufacturing device.

Despite their differences, the notions of Hausdorff computability, which measures the visual proximity of objects effectively, and that of Lebesgue computability, which measures the area or volume of objects effectively, both correspond to observable properties of solids and are therefore both useful in practice.

7 Conclusion

As stated in the Introduction, the solid domain described here is based on a realistic notion of computability which corresponds to the observable properties of solids; it is also closed under Boolean operations which are computable in the model as are the basic predicates. Furthermore, the model admits non-regular as well as regular sets.

In order to design reasonably efficient algorithms based on our model, one should carefully choose the representation for partial solids. Representations used in industrial applications are generally polyhedra or B.Rep. (Boundary representation), that is a set of faces (surfaces), edges (curves) and vertices, connected by an adjacency graph featuring the boundary of the solid.

The dyadic voxel set representation can be made reasonably efficient using recursive binary space subdivision, i.e. octree-like structure. For solids which have, as it is often the case in applications, a boundary with a bounded curvature almost everywhere, partial rational polyhedra will provide a more efficient representation. The performance of these representations can be formally compared by the growth rate, as a function of n, of the volume of data (the number of bits) needed to represent a partial solid up to the Hausdorff or Lebesgue accuracy of 2^{-n}.

However, using partial rational polyhedra in a chain of successive Boolean operations would entail a prohibitive growth of the number of digits necessary to represent the rational coordinates of the vertices. An effective way to overcome this problem would be to use "dyadic polyhedra" together with a rounding process. The idea is to use polyhedra whose vertex coordinates are dyadic numbers. Then, since these polyhedra are not closed under Boolean operators, one can round the exact result to some best approximation in terms of dyadic polyhedra. This process is similar to rounding in fixed or floating point arithmetic. It is also related to some recent works dealing with robustness in computational geometry.

Our future work will focus on realistic implementations based on these ideas as well as theoretical definitions of complexity allowing a formal comparison between algorithms and representations. Also, in order to apply this work to actual CAGD, one needs to capture more information on solids and geometric objects. In particular, we have to deal more generally with the boundary representation and the differential properties of curves and surfaces.

Acknowledgements

The first author has been supported by EPSRC and would like to thank Reinhold Heckmann for constructing a regular open set, on the real line, with a boundary of non-zero Lebesgue measure.

Appendix

We give here the formal definitions of a number of notions in domain theory used in the paper; see [1, 2, 26] for more detail. We think of a partially ordered set (poset) $(P, \sqsubseteq)$ as the set of output of some computation such that the partial order is an order of information: in other words, $a \sqsubseteq b$ indicates that a has less information than b. For example, the set $\{0,1\}^\infty$ of all finite and infinite sequences of bits 0 and 1 with $a \sqsubseteq b$ if the sequence a is an initial segment of the sequence b is a poset and $a \sqsubseteq b$ simply means that b has more bits of information than a. Any T_0 topological space has an inherent information ordering, called the *specialisation ordering*, defined by $a \sqsubseteq b$ iff $a \in O \Rightarrow b \in O$, for all open subsets O. A non-empty subset $A \subseteq P$ is *directed* if for any pair of elements $a, b \in A$ there exists $c \in A$ such that $a \sqsubseteq c$ and $b \sqsubseteq c$. A directed set is therefore a consistent set of output elements of a computation: for every pair of output a and b, there is some output c with more information than a and b. A *directed complete partial order (dcpo)* or a *domain* is a partial order in which every directed subset has a least upper bound (lub). We say that a dcpo is *pointed* if it has a least element which is denoted by $\perp$ and is called *bottom*.

For two elements a and b of a dcpo we say a is *way-below* or *approximates* b, denoted by $a \ll b$, if for every directed subset A with $b \sqsubseteq \bigsqcup A$ there exists $c \in A$ with $a \sqsubseteq c$. The idea is that a is a finitary approximation to b: whenever the lub of a consistent set of output elements has more information than b, then already one of the input elements in the consistent set has more information than a. In $\{0,1\}^\infty$, we have $a \ll b$ iff $a \sqsubseteq b$ and a is a finite sequence. The closed subsets of the *Scott topology* of a domain are those subsets C which are downward closed (i.e. $x \in C \;\&\; y \sqsubseteq x \;\Rightarrow\; y \in C$) and closed under taking lub's of directed subsets (i.e. for every directed subset $A \subseteq C$ we have $\bigsqcup A \in C$).

A basis of a domain D is a subset $B \subseteq D$ such that for every element $x \in D$ of the domain the set $B_x = \{y \in B | y \ll x\}$ of elements in the basis way-below x is directed with $x = \bigsqcup B_x$. An *(ω)-continuous* domain is a dcpo with a (countable) basis. In other words, every element of a continuous domain can be expressed as the lub of the directed set of basis elements which approximate it. In a continuous dcpo D, subsets of the form $\twoheaduparrow a = \{x \in D | a \ll x\}$, for $a \in D$, forms a basis for the Scott topology. A domain is *bounded complete* if every bounded subset has a lub; in such a domain every non-empty subset has an infimum or greatest lower bound.

It can be shown that a function $f : D \to E$ between dcpo's is continuous with respect to the Scott topology if and only if it is *monotone* (i.e. $a \sqsubseteq b \;\Rightarrow\; f(a) \sqsubseteq f(b)$)

and preserves lub's of directed sets i.e. for any directed $A \subseteq D$, we have $f(\bigsqcup_{a \in A} a) = \bigsqcup_{a \in A} f(a)$. Moreover, if D is an ω-continuous dcpo, then f is continuous iff it is monotone and preserves lub's of increasing sequences (i.e. $f(\bigsqcup_{i \in \omega} x_i) = \bigsqcup_{i \in \omega} f(x_i)$, for any increasing $(x_i)_{i \in \omega}$).

The *interval domain* $\mathbf{I}[0,1]^n$ of the unit box $[0,1]^n \subseteq \mathbb{R}^n$ is the set of all non-empty n-dimensional sub-rectangles in $[0,1]^n$ ordered by reverse inclusion. A basic Scott open set is given, for every open subset O of $\mathbb{R}^n$, by the collection of all rectangles contained in O. The map $x \mapsto \{x\} : [0,1]^n \to \mathbf{I}[0,1]^n$ is an embedding onto the set of maximal elements of $\mathbf{I}[0,1]^n$. Every maximal element $\{x\}$ can be obtained as the least upper bound (lub) of an increasing chain of elements, i.e. a shrinking, nested sequence of sub-rectangles, each containing $\{x\}$ in its interior and thereby giving an approximation to $\{x\}$ or equivalently to x. The set of sub-rectangles with rational coordinates provides a countable basis. One can similarly define, for example, the interval domain $\mathbf{I}\mathbb{R}^n$ of $\mathbb{R}^n$.

An important feature of domains, in the context of this paper, is that they can be used to obtain computable approximations to operations which are classically non-computable. For example, comparison of a real number with 0 is not computable. However, the function $N : \mathbf{I}[-1,1] \to \{\mathsf{tt}, \mathsf{ff}\}_\perp$ with

$$N([a,b]) = \begin{cases} \mathsf{tt} & \text{if } b < 0 \\ \mathsf{ff} & \text{if } 0 < a \\ \perp & \text{otherwise} \end{cases}$$

is the computable approximation to the comparison predicate. Here, $\{\mathsf{tt}, \mathsf{ff}\}_\perp$ is the three element pointed domain with two incomparable maximal elements tt and ff.

The *upper space* $\mathbf{U}X$ of a compact metric space X is the set of all non-empty compact subsets of X ordered by reverse inclusion. In fact, $\mathbf{U}X$ is a generalization of the interval domain and has similar properties; for example a basic Scott open set is given, for every open subset $O \subseteq X$, by the collection of all non-empty compact subsets contained in O. As with the interval domain, the map $x \mapsto \{x\} : X \to \mathbf{U}X$ is an embedding onto the set of maximal elements of $\mathbf{U}X$. The upper space gives rise to a computational model for fractals and for measure and integration theory [10]. The idea of the solid domain of $[0,1]^n$ (see Section 2), represented by pairs of closed subsets, is closely linked with $\mathbf{U}[0,1]^n$.

An ω-continuous domain D with a least element $\perp$ is *effectively given* wrt an effective enumeration $b : \mathbb{N} \to B$ of a countable basis B if the set $\{< m,n > | b_m \ll b_n\}$ is recursive, where $< .,. >: \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ is the standard pairing function i.e. the isomorphism $(x,y) \mapsto \frac{(x+y)(x+y+1)}{2} + x$. This means that for each pair of basis elements (b_m, b_n), it is possible to decide in finite time whether or not $b_m \ll b_n$. We say $x \in D$ is *computable* if the set $\{n | b_n \ll x\}$ is r.e. This is equivalent to say that there is a master programme which outputs exactly this set. It is also equivalent to the existence of a recursive function g such that $(b_{g(n)})_{n \in \omega}$ is an increasing chain in D with $x = \bigsqcup_{n \in \omega} b_{g(n)}$. If D is also effectively given wrt to another basis $B' = \{b'_0, b'_1, b'_2, \cdots\}$ such that the sets $\{< m,n > | b_m \ll b'_n\}$ and

$\{< m, n > | b'_m \ll b_n\}$ are both decidable, then x will be computable wrt B iff it is computable wrt B'. We say that B and B' are *recursively equivalent*.

We can define an effective enumeration ξ of the set D_c of all computable elements of D. Let θ_n, $n \in \omega$, be the nth partial recursive function. It can be shown [13] that there exists a total recursive function σ such that $\xi : \mathbb{N} \to D_c$ with $\xi_n := \bigsqcup_{i\in\omega} b_{\theta_{\sigma(n)}(i)}$, with $(b_{\theta_{\sigma(n)}(i)})_{i\in\omega}$ an increasing chain for each $n \in \omega$, is an effective enumeration of D_c. A sequence $(x_i)_{i\in\omega}$ is *computable* if there exists a total recursive function h such that $x_i = \xi_{h(i)}$ for all $i \in \omega$.

We say that a continuous map $f : D \to E$ of effectively given ω-continuous domains D (with basis $\{a_0, a_1 \cdots\}$) and E (with basis $\{b_0, b_1 \cdots\}$) is *computable* if the set $\{< m, n > | b_m \ll f(a_n)\}$ is r.e. This is equivalent to say that f maps computable sequences to computable sequences. Computable functions are stable under change to a recursively equivalent basis. Every computable function can be shown to be a continuous function [35, Theorem 3.6.16]. It can be shown [13] that these notions of computability for the domain $\mathbf{I}\mathbb{R}$ of intervals of $\mathbb{R}$ induce the same class of computable real numbers and computable real functions as in the classical theory [26].

We also need the following classical definitions of sequences of real numbers. A sequence $(r_i)_{i\in\omega}$ of rational numbers is *computable* if there exist three total recursive functions a, b, and s such that $b(i) \neq 0$ for all $i \in \omega$ and

$$r_i = (-1)^{s(i)} \frac{a(i)}{b(i)}.$$

A computable double sequence of rational numbers is defined in a similar way. A sequence $(x_i)_{i\in\omega}$ of real numbers is *computable* if there exists a computable double sequence $(r_{ij})_{i,j\in\omega}$ of rational numbers such that

$$|r_{ij} - x_i| \le 2^{-j} \qquad \text{for all } i \text{ and } j$$

A computable double sequence of real numbers is defined analogously. If $(x_{nk})_{n,k\in\omega}$ is a computable double sequence of real numbers which converges to a sequence $(x_n)_{n\in\omega}$ effectively in k and n (i.e. there exists a total recursive function $e : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that $|x_{nk} - x_n| \le 2^{-N}$ for all $k \ge e(n, N)$), then the sequence $(x_n)_{n\in\omega}$ is computable [26, Page 20].

References

[1] S. Abramsky and A. Jung. Domain theory. In S. Abramsky, D. M. Gabbay, and T. S. E. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 3. Clarendon Press, 1994.

[2] R. M. Amadio and P.-L. Curien. *Domains and Lambda-Calculi.* Cambridge Tracts in Theoretical Computer Science, 1998.

[3] S. M. Barker. Toward a topology for computational geometry. *Computer Aided Design*, 27(4):311–318, 1995.

[4] A. Bauer, L. Birkedal, and D. S. Scott. Equilogical spaces. URL: `http://www.cs.cmu.edu/Groups/LTC/`. Preprint submitted to Elsevier, 1998.

[5] L. Birkedal, A. Carboni, G. Rosolini, and D. S. Scott. Type theory via exact categories (extended abstract). In *Proceedings of the 13th Annual IEEE Symposium on Logic in Computer Science LICS '98*, pages 188–198. IEEE Computer Society Press, 1998.

[6] V. Brattka and K. Weihrauch. Computability on subsets of Euclidean space I: Closed and compact subsets. *Theoretical Computer Science*, 219:65–93, 1999.

[7] H. Desaulniers and N. Stewart. Robustness of numerical methods in geometric computation when problem data is uncertain. *Computer Aided Design, special issue on uncertainties in geometrical design*, 1993.

[8] J. Dugundji. *Topology.* Allyn and Bacon, 1966.

[9] A. Edalat. Dynamical systems, measures and fractals via domain theory. *Information and Computation*, 120(1):32–48, 1995.

[10] A. Edalat. Domains for computation in mathematics, physics and exact real arithmetic. *Bulletin of Symbolic Logic*, 3(4):401–452, 1997.

[11] A. Edalat and A. Lieutier. Foundation of a computable solid modeling. In *Proceedings of the fifth symposium on Solid modeling and applications*, ACM Symposium on Solid Modeling and Applications, pages 278 – 284, 1999.

[12] A. Edalat and P Sünderhauf. Computable Banach spaces via domain theory. *Theoretical Computer Science*, 219:169–184, 1999.

[13] A. Edalat and P. Sünderhauf. A domain theoretic approach to computability on the real line. *Theoretical Computer Science*, 210:73–98, 1999.

[14] S. Fang, B. Bruderlin and X. Zhu. Robustness in solid modelling: a tolerance-based intuitionistic approach. *Computer-Aided Design, special issue on uncertainties in geometrical design*, 1993.

[15] G. Gierz, K. H. Hofmann, K. Keimel, J. D. Lawson, M. Mislove, and D. S. Scott. *A Compendium of Continuous Lattices.* Springer Verlag, Berlin, 1980.

[16] L.J. Guibas, D. Salesin, and J. Stolfi. Epsilon Geometry : Building robust algorithms from imprecise computations. *ACM Symposium on Computational Geometry*, pages 208–217, 1989.

[17] C. Hu. *Towards Robust Interval Solid Modeling of Curved Objects.* PhD thesis, MIT, May 1995.

[18] C. Hu, T. Maekawa, N. Patrikalakis, and E. Sherbrooke. Robust interval algorithm for curve intersections. *Computer Aided Design*, 28(6-7):495–506, 1996.

[19] D.J. Jackson. Boundary Representation Modelling with local Tolerances. *ACM Conference on Solid Modelling*, 1995.

[20] K. Ko and K. Weihrauch. On the measure of two-dimensional regions with polynomial-time computable boundaries. In S Homer, , and J Cai, editors, *Eleventh Annual IEEE Conference on Computational Complexity*, pages 150–159,. IEEE Computer Society Press, 1996.

[21] C. Kreitz and K. Weihrauch. Theory of representations. *Theoretical Computer Science*, 38:35–53, 1985.

[22] A. Lieutier. Représentation b.rep et calculabilité. LMC, IMAG, Université Joseph Fourrier, BP 53X 38041 Grenoble Cedex, France, 1997. Journées modeleurs géométriques.

[23] D. Michelucci. Arithmetic Issues in Geometric Computations. *Second Real Numbers and Computer Conference*, 1996.

[24] V. Milenkovic. Robust polygon modelling. *Computer-Aided Design, special issue on uncertainties in geometrical design*, 1993.

[25] T.J. Peters, D.R. Ferguson, N.F. Stewart, and P.S. Fussell. Algorithmic tolerances and semantics in data exchange. *ACM Annual Symposium on Computational Geometry*, 1997.

[26] M. B. Pour-El and J. I. Richards. *Computability in Analysis and Physics*. Springer-Verlag, 1988.

[27] F. Preparata and M. Shamos. *Computational Geometry: an introduction*. Springer-Verlag, 1985.

[28] A. Requicha and R. Tilove. Mathematical Foundations of Constructive Solid Geometry : General Topology of Closed Regular Sets. In *Automata, Languages and Programming*, Production Automation Project, March 78, University of Rochester, Rochester, New York, 1978.

[29] A.A.G. Requicha. Representation for Rigid Solids : Theory, Methods, and Systems. *Computer Surveys*, 12(4), 1980.

[30] A.A.G. Requicha and H.B. Voelker. Boolean Operations in Solid Modeling: Boundary Evaluation and Merging Algorithms. *Proceedings of the IEEE*, 73(1), 1985.

[31] D. S. Scott. A new category? Domains, spaces, and equivalence relations. URL: `http://www.cs.cmu.edu/Groups/LTC/`, 1996.

[32] M. Segal. Using tolerances to guarantee valid polyhedral modeling results. *Computer Graphics*, 24, 1990.

[33] V. Stoltenberg-Hansen and J. V. Tucker. Complete local rings as domains. *Journal of Symbolic Logic 53*, 53:603–624, 1988.

[34] V. Stoltenberg-Hansen and J. V. Tucker. Effective algebras. In D. Gabbay S. Abramsky and T.S.E. Maibaum eds., editors, *Handbook of Logic in Computer Science*, volume 4, pages 357–526. Oxford University Press, 1995.

[35] K. Weihrauch. *Computability*, volume 9 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, 1987.

[36] K. Weihrauch. A foundation for computable analysis. In D.S. Bridges, C.S. Calude, J. Gibbons, S. Reeves, and I.H. Witten, editors, *Combinatorics, Complexity, and Logic*, Discrete Mathematics and Theoretical Computer Science, pages 66–89, Singapore, 1997. Springer-Verlag. Proceedings of DMTCS'96.