EFFECTIVE SOLUTIONS
OF
RECURSIVE DOMAIN EQUATIONS

by

Akira Kanda

Submitted for the Degree of
Doctor of Philosophy
at the
University of Warwick

A Dedication To

Wolfgang Amadeus Mozart (1756-1791)

E.E.6031

CONTENTS

ACKNOWLEDGEMENTS

To my supervisor David Park, whose discovery of the theorem 2.2.1 was the first real step to clear up many mysteries about effectively given domains, and to make the notion sound. He encouraged me to work on the theory of effectively given SFP objects, which had been predicted by many people, to be a routine extension of the theory of effectively given domains; but which turned out to give many deeper insights into the effectiveness of domains and to be a nontrivial extension. Indeed the author and Park were forced to question seriously whether computability of elements of an effectively given domain is dependent on the indexing of its basis or not, through an attempt to develop a theory of effectively given SFP objects. Also talks with him helped me to clarify the notion of effective O-category.

To Michael Smyth who patiently explained his work on effectively given continuous domains and on categorical solutions of recursive domain equations. His works essentially motivated me. Also his criticism and suggestions played very important roles in the development of this dissertation.

To Dana Scott whose excellent lectures were a very helpful introduction for me to the study of domains. The beauty and insight inherent in his lattice theoretic theory of computation were the first things which attract me to this field.

To Michael Paterson who taught me much about computability by answering many elementary questions on computability problems.

Gordon Plotkin and Meurig Beynon are very much appreciated for their intrinsic comments on the last chapter and encouragement to take the last chapter more seriously, even though it was considered to be rather obvious argument. The inclusion of the power domain functor was suggested by them.

I also was benefitted from discussions with William Wadge and Klause Weihrauch. Especially comments from Weihrauch on the category of recursively enumerable sets were very helpful.

The research and writing took place at Warwick University and the University of Leeds. The author was supported in part by SRC grant GR/A66772.

Finally, this research was carried out under two successive crises of the author, a financial crisis which covered the initial half of the course of this research, and a domestic crisis which covered the last half of it. The author thanks to all of those who helped and encouraged him in any form. This research would never be finished without the warm support of these friends.

ABSTRACT

Solving recursive domain equations is one of the main concerns in the denotational semantics of programming languages, and in the algebraic specification of data types. Because we are to solve them for the specification of computable objects, effective solutions of them should be needed. Though general methods for obtaining solutions are well known, effectiveness of the solutions has not been explicitly investigated.* The main objective of this dissertation is to provide a categorical method for obtaining effective solutions of recursive domain equations. Thence we will provide effective models of denotational semantics and algebraic data types. The importance of considering the effectiveness of solutions is two-fold. First we can guarantee that for every denotational specification of a programming language and algebraic data type specification, implementation exists. Second, we have an instance of a computability theory where higher type computability and even infinite type computability can be discussed very smoothly.

*While this dissertation has been written, Plotkin and Smyth obtained an alternative to our method which worked only for effectively given categories with universal objects.

INTRODUCTION

I.1 Historical Remarks

Recursive domain equations play a crucial role in the denotational semantics of programming languages as developed by Scott and Strachey [21] and their followers (Tennent [27] and Stoy [25]). For example in denotational semantics of a language which allows commands to be stored, the following domain equations should be solved as a recursive specification of the domain of stores:

$$S = [L \to ([S \to S] + V')] \qquad \text{--------} \quad (1)$$

where L is the domain of locations, V' is the domain of other store values and [S→S] is the domain of commands which is the space of functions from S to S.

If we interpret + and → set theoretically, then by a straightforward cardinal argument, we can observe that there is no solution to (1). Indeed the right side becomes much bigger than the left side. Scott [16] indicated that if we restrict functions allowed in the function spaces so that we can make the cardinal of [S→S] the same as that of S, we could solve equations like (1). This calls for structuring domains rather than regarding them as plain sets.

In fact Scott [16] regarded domains as complete lattices where the partial ordering $a \sqsubseteq b$ means that the computaion a approximates the computation b. Thence he allowed only continuous, in the sense of directed limit preserving, functions in function spaces. This idea of Scott is based on a principle that every computable object should be a limit of a chain of finite approximations to itself. Indeed every partial recursive function is such an object (see Kleene [6]).

Later several authors (Milner [7], Markowsky and Rosen [15], Plotkin [10], Smyth [22]) pointed out that complete lattices are too rich and a sufficient structure is the structure used by Scott in his earlier work (1969 private comunication). Roughly the structure is a poset with the least element which admits a lub for each directed subset. This structure is the so called cpo (complete partial ordering).

In order to discuss computability, the partial ordering structure is not sufficient. We need some kind of effectiveness structure.

Scott [16] proposed a notion of effectively given domains to tackle this problem. The essential idea of this approach is to handle only those cpo's (countably continuous cpo's) each of which has a countable substructure called a basis s.t. the whole structure can be regained from the basis by means of completion, i.e. by means of taking lub's of directed subsets of the basis. Then assuming computability of the finite join operation on the basis, we define the computable elements to be the lub's of r.e. directed subsets of the basis. Roughly speaking, we regard an element to be computable iff it can be approximated effectively. In case the basis is the set of all compact (finite) elements of the cpo, we call such an effectively given cpo as effectively given algebraic cpo.

This idea of Scott was further studied, for the algebraic case by Egli and Constable [1], and Markowsky and Rosen [15], and for the continuous case by Tang [26] and Smyth [22].

The firs' solution by Scott [18] and Tang [26], for solving recursive domain equations like (1) used the idea of universal domains. They showed that a solution of the equation:

$$D = [D \to D]$$

obtained as the limit T_∞ of:

$$T_0 = \text{the two point lattice.}$$

$$T_{n+1} = [T_n \to T_n].$$

is a universal domain for the class of countably continuous lattices, i.e. every element of this class is a retract of T_∞. They developed a retract calculus which yields a continuous lattice solution to each recursive domain equation as the range of an idempotent which is a least fixed point of a functional associated with the domain equation. Scott [15] also showed that $P\omega$ is a universal domain for the class of countably continuous lattices and developed a retract calculus which provides countably continuous lattice solutions to recursive domain equations.

Furthermore Plotkin [12] and Scott [20] independently obtained a universal domain for the class of countably continuous bounded complete cpo's.

Scott [17] suggested that there should be a categorical method for obtaining lattice solutions to recursive domain equations. This idea was made explicit by Reynolds [13] and Wand [29]. Smyth [24] proposed a theory of ω-categories which guarantees the initiality of the solutions of the domain equations. This theory relates to the leastness of the solutions via retract calculus. Smyth and Plotkin [24] studied several interesting relations between the categorical approach and the retract calculus approach.

Even though not made precise, effective methods for obtaining solutions of recursive domain equations were considered for the retract calculus approach as in Scott [18], Scott [20], and Plotkin [12]. On the contrary there has been no known categorical method for effectively solving domain equations.

I.2 Recent Developments

All methods, obtained so far, for yielding solutions to recursive domain equations like (1) used a mathematical device to reduce the cardinality of the right-hand side to that of the left-hand side by restricting functions to be continuous. Recently the author [3,4] showed that we can play this game at the cardinality ω. He proposed to handle only those partially ordered sets each of which can be recovered from its basis by means of "effective completion". This guarantees each structure to be countable. He then showed that these structures can be characterized as the sets of computable elements of effectively given domains. Then by allowing only effective functions from D to D' in [D→D'], he showed that the resulting function space is also such a structure. Notice that we have reduced both sides of (1) to countable sets. We call such structures effective domains.

Furthermore the author [3,4] proposed a theory of effective ω-categories (he called it "effective categories") which is an effective version of Smyth's ω-categories and showed this categorical theory provides effectively initial effective domain solutions to recursive domain equations. Also the author and Park [5] showed that this theory can be applied to obtain effectively initial solutions, which are effectively given domains.

In parallel to these effectiveness results Smyth and Plotkin [24] developed an effective retract calculus in categorical setting.

Apart from a denotational semantics, recent development of Lehmann and Smyth [23] showed that the recursive domain specifications play an essential role in the algebraic specification of data types. In connection with this, the relation between Wand's O-category and Smyth's ω-category appeared to be important. Smyth and Plotkin [24] studied this problem in great detail .

But they doubted the possibility of introducing effectiveness to O-categories in general. Instead they observed that every O-category with a universal object has a unique ordered monoid representation. Thence by introducing effectiveness to the ordered monoid representation they managed to consider effectively given O-categories with a universal object. Thus they developed an effective retract calculus. However following an idea of Park on effectiveness of O-categories, the author recently obtained a very natural notion of effective O-categories in general. He then studied relations between effective ω-categories and effective O-categories.

I.3 The Objective of This Dissertation

In this thesis, we will be concerned mainly with the recent results of effective O-categories and effective ω-categories. As a concrete instance of these effective categories, we will review the class of effectively given bounded complete algebraic cpo's (called effectively given (algebraic) domains) and emphasize a crucial point which has been overlooked in the previous works. Namely the dependence of the computablity of elements on the indexing of the basis, as discovered by Park and the notion of effective embeddings and effective isomorphisms as developed by the author (see Kanda and Park [5]). Furthermore we will study the class of effective domains as an effective category.

So far the effectiveness of domains is studied only for the bounded complete case. Plotkin [10] showed that this condition, imposed to make function spaces of algebraic domains algebraic, can not be preserved under his power domain construction, and proposed the so called SFP condition to compensate for this deficiency. This condition is more general than bounded completeness. We observe that the SFP condition is more natural than bounded completeness on account of effectiveness. Thence we will study effectively given SFP objects and observe that they form an effective category.

It has been said that one of the biggest demerits of the categorical approach is lack of suitable effectiveness notion. In this dissertation, we will present an answer to this criticism. Furthermore we argue that the categorical approach, as it is here, has generality in its application. In fact there is no known way of solving recursive domain equations which involve Plotkin power domain constructor, by means of the retract calculus. But the categorical method enables us to do so.

We admit that the categorical approach still is not so developed as the universal domain approach, in the sense that it lacks in natural notation for the computable elememnts. This problem is left open.

Finally notice that the results in this dessertation supply an effective version of Lehmann and Smyth [23] data types. Also they provide a natural way of discussing computability over higher types and infinite types.

CHAPTER 1: NON-EFFECTIVE SOLUTIONS

I have not indicated in which degree the results in this work are original, which is always a difficult problem when a uniform approach to a subject is presented.

J. Engelfriet

In this chapter, before we study effectiveness, we briefly review results already known on non-effective domain theory.

1.1 Complete Partial Ordering

By a partially ordered set (poset), we mean a pair $(D,\sqsubseteq)$ where D is a set and $\sqsubseteq$ is a partial ordering, i.e., reflexive, anti-symmetric, and transitive relation on D. An element x of D is an upper bound of a subset $S\subseteq D$, in symbols $S\sqsubseteq x$ iff $x\sqsupseteq y$ for all $y\in S$. An upper bound z of S is a least upper bound (lub) of S, in symbols $\sqcup S$, iff $x\sqsupseteq S$ implies $x\sqsupseteq z$. A subset $S\subseteq D$ is directed iff every finite subset $F\subseteq S$ has an upper bound in S. D is directed (bounded) complete iff every directed (bounded) subset has a lub.

Definition 1.1.1

Given posets $(D,\sqsubseteq)$ and $(D',\sqsubseteq')$:

(1) A function $f:D\to D'$ is monotone iff $x\sqsubseteq y$ implies $f(x)\sqsubseteq f(y)$.

(2) A function $f:D\to D'$ is continuous iff for every directed subset $S\subseteq D$ with a lub, $f(S)=\{f(x)\,|\,x\in S\}$ is directed and has a lub, and $f(\sqcup S)=\sqcup' f(S)$. □

It is straight-forward to observe that every continuous function is monotone. Given two monotone maps $f,g:D\to D'$, we order them by pointwise ordering, i.e.,

$$f\sqsubseteq g \text{ iff } f(x)\sqsubseteq g(x) \text{ for all } x\in D.$$

Definition 1.1.2

A monotone (continuous) map $h:D \to D$ is a monotone (continuous) idempotent iff $h=h \cdot h$. It is projective (inflative) iff $h \sqsubseteq id_D$ ($h \sqsupseteq id_D$) where id_D is the identity map on D. □

Definition 1.1.3

Let $(e:D \to D', r:D' \to D)$ be a pair of monotone functions:

(1) (e,r) is a monotone retraction pair (from D to D') iff $r \cdot e$ is the identity map on D'. r is called a monotone retraction and e is called a monotone section. We also say that D' is a monotone retract of D (via (e,r)). Notice that r is surjective, thus D'=range(r). Evidently $h=e \cdot r$ is a monotone idempotent and will be called a monotone retraction idempotent of (e,r).

(2) (e,r) is a monotone projection pair iff it is a monotone retraction pair s.t. the monotone retraction idempotent h is projective. In this case r is called a monotone projection and e is called a monotone embedding. h is called a monotone projection idempotent.

(3) (e,r) is a monotone inflation pair iff it is a monotone retraction pair s.t. h is inflative. In this case, we say that r is a monotone inflation and e is a monotone infbedding. Also h is called a monotone inflation idempotent. □

Lemma 1.1.4

(1) A monotone projection determines a corresponding monotone embedding uniquely and vice versa.

(2) A monotone inflation determines a corresponding monotone infbedding uniquely and vice versa.

proof (1) Let (e,r) and (e',r) be monotone projection pairs. Then $e'=e' \cdot r \cdot e \sqsubseteq e$ and $e=e \cdot r \cdot e' \sqsubseteq e'$. Thus e=e'. Let (e,r) and (e,r') be monotone projection pairs. Then $r=r' \cdot e \cdot r \sqsubseteq r'$ and

$r'=r\cdot e\cdot r'\sqsubseteq r$. Thus $r=r'$.

(2) Let (e,r) and (e',r) be monotone inflation pairs. Then $e'=e'\cdot r\cdot e\sqsupseteq e$ and $e=e\cdot r\cdot e'\sqsupseteq e'$. Thus $e=e'$. Let (e,r) and (e,r') be monotone inflation pairs. Then $r=r'\cdot e\cdot r\sqsupseteq r'$ and $r'=r\cdot e\cdot r'\sqsupseteq r$. Therefore $r=r'$. ◻

By virtue of this lemma, we will say that $e(r)$ is the <u>adjoint</u> of $r(e)$ whenever (e,r) is a monotone projection pair or a monotone inflation pair.

<u>Lemma 1.1.5</u>

(1) A monotone embedding preserves and reflects all existing least upper bounds.

(2) A monotone inflation preserves all existing lub's.

<u>proof</u> (1) Let (e,r) be a monotone projection pair from D to D'. Assume $\bigsqcup X\in D$. Then $e(\bigsqcup X)\sqsupseteq e(X)$. Let $v'\sqsupseteq e(X)$. Then $r(v')\sqsupseteq r\cdot e(X)=X$. Thus $r(v')\sqsupseteq\bigsqcup X$. Thus $e(\bigsqcup X)\sqsubseteq e\cdot r(v')\sqsubseteq v'$. Therefore $e(\bigsqcup X)=\bigsqcup e(X)\in D'$. Thus e preserves lub's. Now let $\bigsqcup e(X)\in D'$. Then $r(\bigsqcup e(X))\sqsupseteq r\cdot e(X)=X$. Let $u\sqsupseteq X$. Then $e(u)\sqsupseteq\bigsqcup e(X)$ and $r\cdot e(u)=u\sqsupseteq r(\bigsqcup e(X))$. Therefore $r(\bigsqcup e(X))=\bigsqcup X$. Therefore e reflects lub's.

(2) Let (e,r) be a monotone inflation pair from D to D'. Assume $\bigsqcup X'\in D'$. Then $r(\bigsqcup X')\sqsupseteq r(X')$. Let $v\sqsupseteq r(X')$. Then $e(v)\sqsupseteq e\cdot r(X')\sqsupseteq X'$. Thus $e(v)\sqsupseteq\bigsqcup X'$. Thus $r(\bigsqcup X')\sqsubseteq r\cdot e(v)=v$. Therefore $r(\bigsqcup X')=\bigsqcup r(X')$. Therefore r preserves lub's. ◻

Notice that 1.1.5 is claiming that all monotone embeddings and monotone inflations are continuous. But we can not establish that every monotone projection and every monotone infbedding are continuous. This calls for the following:

<u>Definition 1.1.6</u>

(1) A <u>continuous retraction pair</u> is a monotone retraction pair (e,r) s.t. both r and e are continuous. r is called a <u>continuous retraction</u> and e is called a <u>continuous section</u>. $h=e\cdot r$ is

a continuous retraction idempotent.

(2) A continuous projection pair, a continuous inflation pair are defined similarly. □

Notice that a continuous projection (inflation) is a monotone projection (inflation), which is continuous by 1.1.5, s.t. the corresponding monotone embedding (infbedding) is continuous.

Definition 1.1.7

Given posets D and D', a monotone map $f:D\to D'$ is an isomorphism (from D to D') iff there exists a monotone map $f^R:D'\to D$ s.t. $f\cdot f^R = id_{D'}$ and $f^R\cdot f = id_D$. In this case we say that D and D' are isomorphic (via f) and denote it by $D\cong D'$. □

Evidently an isomorphism $f:D\to D'$ is a monotone embedding s.t. the adjoint f^R is also a monotone embedding. Thus by 1.1.5 both f and f^R are continuous.

The combination of posets and monotone maps is not quite interesting although it is natural. But the combination of continuous functions and the so called cpo's (defined below) is interesting.

Definition 1.1.8

A complete partial order (cpo) is a directed complete poset with a least element (called bottom). We will denote the bottom of a cpo $(D,\sqsubseteq)$ by $\bot_D$ or simply by $\bot$. □

Definition 1.1.9

Given posets with bottoms D and D', define $D\times D'$, $D+D'$, and $[D\to D']$ to be the following posets with bottoms:

(1) $D\times D' = \{(d,d') \mid d\in D,\ d'\in D'\}$ together with the coordinate-wise ordering, i.e., $(d_1,d_1')\sqsubseteq(d_2,d_2')$ iff $d_1\sqsubseteq d_2$ and $d_1'\sqsubseteq d_2'$.

(2) $D+D' = \{(0,d) \mid d\in D\} \cup \{(1,d') \mid d'\in D'\} \cup \{\bot\}$ together with the following ordering:

$\perp \sqsubseteq (i,x)$ for all $(i,x) \in D+D'$

$(i,x) \sqsubseteq (j,y)$ iff $i=j$ and $x \sqsubseteq y$.

(3) $[D \to D']$ is the set of all continuous functions together with the point-wise ordering. □

Fact 1.1.10

If D and D' are cpo's then so are $D \times D'$, $D+D'$, and $[D \to D']$. □

Definition 1.1.11

(1) A family $\langle D_m, (f_m, f_m^R) \rangle$ is an *ω-sequence of continuous projection pairs of cpo's* iff D_m is a cpo and (f_m, f_m^R) is a continuous projection pair from D_m to D_{m+1}.

(2) The *inverse limit* of an ω-sequence $\langle D_m, (f_m, f_m^R) \rangle$ of continuous projection pairs of cpo's is a set $D_\infty = \{\langle x_m \rangle \mid x_m = f_m^R(x_{m+1})\}$ together with the coordinate-wise ordering. We will denote the inverse limit by $\underleftarrow{\lim} \langle D_m, (f_m, f_m^R) \rangle$ or $\underleftarrow{\lim} \langle D_m \rangle$. The *universal cocone* of $\langle D_m, (f_m, f_m^R) \rangle$ is a family $\langle (f_{m\infty}, f_{m\infty}^R) \rangle$ where $f_{m\infty} : D_m \to D_\infty$ is defined by :

$$f_{m\infty}(x) = \langle f_0^R \cdot \ldots \cdot f_{m-1}^R(x), \ldots, f_{m-1}^R(x), x, f_m(x), f_{m+1} \cdot f_m(x), \ldots \rangle$$

and $f_{m\infty}^R : D_\infty \to D_m$ is defined by $f_{m\infty}^R(\langle x_n \rangle) = x_m$. □

Fact 1.1.12

Given an ω-sequence $\langle D_m, (f_m, f_m^R) \rangle$ of continuous projection pairs of cpo's:

(1) The inverse limit D_∞ is a cpo.

(2) $(f_{m\infty}, f_{m\infty}^R)$ is a projection pair from D_m to D_∞. □

Fact 1.1.13

Let D be a cpo and $h : D \to D$ be a continuous idempotent. Then the range of h, which is the set of all fixed points of h is a cpo where the partial ordering is induced from D. □

Fact 1.1.14

Let D and D' be posets with bottoms $\perp$ and $\perp'$ respectively.

(1) Let (e,r) be a monotone projection pair from D to D'. Then both r and e are strict, i.e., they preserve bottoms.

(2) Let (e,r) be a monotone inflation pair from D to D'. Then r is strict.

proof (1) $r(\bot')\sqsupseteq\bot$ and $e(\bot)\sqsupseteq\bot'$. Thus $\bot=r\cdot e(\bot)\sqsupseteq r(\bot')$ and $\bot'\sqsupseteq e\cdot r(\bot')\sqsupseteq e(\bot)\sqsupseteq\bot'$. Thus $r(\bot')=\bot$ and $e(\bot)=\bot'$.

(2) $r(\bot')\sqsupseteq\bot$ and $e(\bot)\sqsupseteq\bot'$. Thus $\bot=r\cdot e(\bot)\sqsupseteq r(\bot')$. Therefore $r(\bot')=\bot$. ☒

Notice that this fact implies that every continuous projection, continuous embedding, and continuous inflation from a cpo to another are strict.

Fact 1.1.15

Let D_1, D_1', D_2, and D_2' be cpo's s.t. $D_1\cong D_1'$ and $D_2\cong D_2'$, then:

(1) $D_1\times D_2\cong D_1'\times D_2'$

(2) $D_1+D_2\cong D_1'+D_2'$

(3) $[D_1\to D_2]\cong[D_1'\to D_2']$ ☒

Definition 1.1.16

Let $<D_m,(f_m,f_m^R)>$ and $<D_m',(f_m',f_m'^R)>$ be ω-sequences of continuous projection pairs of cpo's. We say they are isomorphic, in symbols, $<D_m,(f_m,f_m^R)>\cong<D_m',(f_m',f_m'^R)>$, iff there exists an isomorphism $i_m:D_m\to D_m'$ with the adjoint $i_m^R:D_m'\to D_m$ s.t. $f_m'\cdot i_m=i_{m+1}\cdot f_m$ for each $m\in N$. □

Fact 1.1.17

$<D_m,(f_m,f_m^R)>\cong<D_m',(f_m',f_m'^R)>$ implies $D_\infty\cong D_\infty'$. ☒

1.1.15 and 1.1.17 indicates that the isomorphic relation is a good criterion for identifying two cpo's.

1.2 Bounded Complete Countably Based Cpo's

As indicated in Scott [16], the first step towards effectiveness is to think of those cpo's each of which has a countable subset (called a basis) from which we can recover original structure by means of completion. Thus the notion of countably based cpo's appears to be important.

Definition 1.2.1

Given a cpo $(D,\sqsubseteq)$, define a relation $\prec$ on D by :

$x \prec y$ iff for every directed subset $S \subseteq D$ with a lub, $y \sqsubseteq \sqcup S$ implies $x \sqsubseteq z$ for some $z \in S$.

In case $x \prec x$, we say that x is compact (or finite). The set of all compact elements of D will be denoted by E_D. □

Fact 1.2.2

In a cpo $(D,\sqsubseteq)$ we have:

(1) $\bot \prec x$.

(2) $x \prec y \sqsubseteq z$ implies $x \prec z$.

(3) $x \sqsubseteq y \prec z$ implies $x \prec z$.

(4) $x \prec y$ and $y \prec z$ implies $x \prec z$.

(5) $x \prec y$ and $z \prec y$ implies $x \sqcup z \prec y$ whenever $x \sqcup z$ exists.

(6) $x \prec y$ implies $x \sqsubseteq y$.

(7) $x \in E_D$ implies $x \sqsubseteq y$ iff $x \prec y$. ▨

Notice that $\prec$ is a transitive and anti-symetric relation on D.

Definition 1.2.3

(1) A pair $(B,\prec)$ is an ω-basis of a cpo $(D,\sqsubseteq)$ iff B is a countable subset of D s.t. $\bot \in B$ and for every $x \in D$, $B_x = \{b \in B \mid b \prec x\}$ is directed and $x = \sqcup B_x$. If $(D,\sqsubseteq)$ has an ω-basis we say that $(D,\sqsubseteq)$ is ω-basable.

(2) An ω-based cpo is a pair $((D,\sqsubseteq),(B,\prec))$ where $(B,\prec)$ is an ω-basis of $(D,\sqsubseteq)$. We will abbreviate this pair as (D,B).

(3) An <u>ω-algebraic</u> <u>cpo</u> is a cpo $(D,\sqsubseteq)$ s.t. E_D is a countable set and for every $x \in D$, $J_x=\{e \in E_D \mid e \sqsubseteq x\}$ is directed, and $x=\bigsqcup J_x$. $(E_D,\sqsubseteq)$ is called the extension basis of $(D,\sqsubseteq)$. □

Notice that in the previous development of the theory of domains, the distinction between ω-basable domains and ω-based domains was never made explicit. Indeed it looks as though these two notions were assumed to be identical. We will observe that they are different and the difference is intrinsic for effectiveness arguments.

<u>Fact 1.2.4</u>

(1) If $(D,\sqsubseteq)$ is an ω-basable cpo, then for every $d \in D$, $D_d=\{x \in D \mid x \prec d\}$ is directed and $d=\bigsqcup D_d$.

(2) A cpo $(D,\sqsubseteq)$ is an ω-algebraic cpo iff $(E_D,\sqsubseteq)$ is an ω-basis of it. □

The relation $\prec$ on an ω-basable cpo enjoys more interesting properties than those listed in 1.2.2.

<u>Fact 1.2.5</u>

Let $((D,\sqsubseteq),(B,\prec))$ be an ω-based domain.

(1) For every directed subset $S \subseteq D$, we have:

$$x \prec \bigsqcup S \quad \text{iff} \quad x \prec z \text{ for some } z \in S.$$

(2) $x \prec y$ iff $x \prec b$ and $b \prec y$ for some $b \in B$.

(3) $x \sqsubseteq y$ iff for all $b \in B$, $b \prec x$ implies $b \prec y$.

<u>proof</u> (1) Sufficiency is true for any poset. For necessity we first prove the following lemma.

<u>Lemma</u> For every directed subset $S \subseteq D$, $x \prec y \,\&\, y \sqsubseteq \bigsqcup S$ implies there exists $z \in S$ s.t. $x \prec z$.

<u>proof of the lemma</u> For every $z \in S$, $D_z=\{t \in D \mid t \prec z\}$ is directed and $z=\bigsqcup D_z$ as observed in 1.2.4. Evidently for every $z_1,z_2 \in S$, $z_1 \sqsubseteq z_2$ implies $D_{z_1} \subseteq D_{z_2}$. Let $S^*=\bigcup_{z \in S} D_z$ and $\{a_1,\ldots,a_n\} \subseteq S^*$. Then $a_i \prec z_i$ for

some $z_i \in S$ $(1 \le i \le n)$. Since S is directed there is an upper bound $z \in S$ of $\{z_1,..,z_n\}$. Thus $D_{z_i} \subseteq D_z$ $(1 \le i \le n)$. Thence $\{a_1,..,a_n\} \subseteq D_z$. By directedness of D_z, there is an upper bound $a \in D_z$ of $\{a_1,..,a_n\}$. Evidently $a \in S^*$, thus S^* is directed. Also evidently $\sqcup S = \sqcup S^*$. Thus for some $t \in S^*$, $x \sqsubseteq t$. But $t \in D_z$ for some $z \in S$. Therefore $x \sqsubseteq t \prec z$ and $x \prec z$. ∅

Now we resume the proof of (1). Let $x \prec \sqcup S$, then $x \prec \sqcup S \sqsubseteq \sqcup S$. Thus $x \prec z$ for some $z \in S$ by the above lemma.

(2) Sufficiency is (4)-1.2.2. We prove necessity. Let $x \prec y$. Since $y = \sqcup\{b \in B \mid b \prec y\}$, $x \prec b \prec y$ for some $b \in B$ by (1).

(3) Let $x \sqsubseteq y$, then $b \prec x$ implies $b \prec y$. Now assume that $b \prec x$ implies $b \prec y$. Then $B_x \subseteq B_y$. Thus $x \sqsubseteq y$. ∅

The following fact is the reason why we call $(E_D, \sqsubseteq)$ the extension basis of $(D, \sqsubseteq)$. Also it is the reason why we are mainly interested in the extension bases of ω-algebraic cpo's.

<u>Fact 1.2.6</u>

A poset $(D, \sqsubseteq)$ is an ω-algebraic cpo iff E_D is countable and for any cpo Q and every monotone map $m: E_D \to Q$ there is a unique continuous extension $\overline{m}: D \to Q$ of m. Indeed such extension $\overline{m}$ is given by: $\overline{m}(x) = \sqcup\{m(e) \mid e \in E_D, e \sqsubseteq x\}$. ∅

An <u>ω-transitively ordered set</u> (<u>ω-toset</u>) is a pair $(E,<)$ where E is a countable set and $<$ is a transitive relation on E. For ω-tosets, we define the notion of upper bounds, lub's, bounded subsets, directed subsets, as we did for posets. To make the point that they are for an ω-toset $(E,<)$ explicit, we call them $<$-upper bounds, $<$-lub's, $<$-bounded subsets, and $<$-directed subsets respectively. Also given ω-tosets $(E,<)$ and $(E',<')$, we say that a map $f: E \to E'$ is <u>t-monotone</u> iff $x<y$ implies $f(x) <' f(y)$. Given a pair of t-monotone maps $(i: E \to E', j: E' \to E)$, we say that this pair is an <u>isomorphism pair</u> iff $i \cdot j = id_{E'}$ and $j \cdot i = id_E$. In this case we also say that E is isomorphic to E' (via (i,j)), in symbols $E \cong E'$.

Definition 1.2.7

(1) An R-structure is an ω-toset $(E,<)$ with the least (w.r.t.$<$) element $\perp$ s.t. for any $a\in E$, the set $[a]=\{b\in E \mid b<a\}$ is directed.

(2) Given two R-structures $(E,<)$ and $(E',<')$, we say that E is isomorphic to E' (in symbols $E\cong E'$) iff $E\cong E'$ as ω-tosets.

(3) A cut of an R-structure $(E,<)$ is a down-ward closed directed subset of E. More precisely, it is a subset $X\subseteq E$ s.t. X is directed and $x<y \& y\in X$ implies $x\in X$. The completion of $(E,<)$ is a pair $((\overline{E},\subseteq),([E],\prec))$ where $\overline{E}$ is the set of all cuts of E and $\subseteq$ is the set theoretical inclusion as a partial ordering, and $[E]=\{[e] \mid e\in E\}$ and $\prec$ is taken over $(\overline{E},\subseteq)$. Note $[a]\in\overline{E}$ for all $a\in E$. □

It is evident that an ω-basis of an ω-basable cpo is an R-structure.

Fact 1.2.8

Given two ω-basable cpo's $(D,\sqsubseteq)$ and $(D',\sqsubseteq')$, $(D,\sqsubseteq)\cong(D',\sqsubseteq')$ iff there are ω-bases $(B,\prec)$ and $(B',\prec')$ of $(D,\sqsubseteq)$ and $(D',\sqsubseteq')$ respectively s.t. $(B,\prec)\cong(B',\prec')$.

proof If $(i:B\to B', j:B'\to B)$ is an isomorphism pair s.t. $(B,\prec)$ and $(B',\prec')$ are ω-bases of $(D,\sqsubseteq)$ and $(D',\sqsubseteq')$ respectively, then $(\overline{i}:D\to D', \overline{j}:D'\to D)$ is an isomorphism pair where $\overline{i}$ and $\overline{j}$ are continuous extensions of i and j respectively s.t. $\overline{i}(x)=\bigsqcup\{i(b) \mid b\in B, b\prec x\}$ and $\overline{j}(x)=\bigsqcup\{j(b') \mid b'\in B', b'\prec' x'\}$. Conversely if $(D,\sqsubseteq)$ has an ω-basis $(B,\prec)$ and $(\overline{f}:D\to D', \overline{g}:D'\to D)$ is an isomorphism pair, then $(\overline{f}(B),\prec')$ is an ω-basis of $(D',\sqsubseteq')$ and $(f:B\to\overline{f}(B), g:\overline{f}(B)\to B)$ is an isomorphism pair where $f=\overline{f}\restriction B$ and $g=\overline{g}\restriction\overline{f}(B)$. ☒

By a strong R-structure (SR-structure)', we mean an R-structure $(E,<)$ satisfying:

$$[a]\subseteq[b] \ \& \ b<c \ \text{ implies } a<c \text{ , and } [a]=[b] \text{ implies } a=b.$$

In (1)-1.2.7, if $\perp<\perp$ then $[\perp]=\{b\in E \mid b<\perp\}\neq\emptyset$. If we allow the empty set to be directed, as in Smyth [22], then $\perp$ is not the bottom wrt $\subseteq$ in $(\overline{E},\subseteq)$.

Fact 1.2.9 (Park [9])

Let $(E,<)$ be a strong R-structure. Then $([E],\prec)$ is isomorphic to $(E,<)$, where $((\overline{E},\subseteq),([E],\prec))$ is the completion of $(E,<)$.

Fact 1.2.10 (Smyth [22],Park [9])

(1) If $(E,<)$ is an R-structure, then $((\overline{E},\subseteq),([E],\prec))$ is an ω-continuous cpo. Furthermore for any $x,y\in\overline{E}$,

$$x\prec y \text{ iff } \exists a\in y.\, x\subseteq[a] .$$ ☒

(2) Let $(D,\sqsubseteq)$ be an ω-basable cpo and $(B,\prec)$ be an ω-basis of it. Then $(B,\prec)$ is an SR-structure. Therefore by virtue of 1.2.8 and 1.2.9, $(D,\sqsubseteq)\cong(\overline{B},\subseteq)$.

Notice that given an R-structure $(E,<)$, it is not necessarily isomorphic to $([E],\prec)$. It is so whenever $(E,<)$ is an SR-structure. But it is easy to observe that $(E,<)\cong(E',<')$ iff $([E],\prec)\cong([E'],\prec')$. This leads to the following fact:

Fact 1.2.11

Given ω-basable cpo's $(D,\sqsubseteq)$ and $(D',\sqsubseteq')$. $(D,\sqsubseteq)\cong(D',\sqsubseteq')$ iff there are R-structures $(E,<)$ and $(E',<')$ s.t. $(\overline{E},\subseteq)\cong(D,\sqsubseteq)$ and $(\overline{E}',\subseteq)\cong(D',\sqsubseteq')$ and $(E,<)\cong(E',<')$. ☒

Our arguments about completion seem to suggest that the ω-bases are redundant representations of ω-based cpo's, and R-structures are rich enough. But later in this section, we will observe that R-structures might be too poor to get strong results. To be prepared for this observation, we notice that the step from ω-bases to R-structure is 'throwing away $\sqsubseteq$ ordering'.

For ω-algebraic cpo's the whole argument about completion is much simpler, since we are interested only in extension bases.

Definition 1.2.12

A strict poset is a poset with the least element. The

<u>algebraic</u> <u>completion</u> of a strict poset $(E,\sqsubseteq)$ is a poset $(\overline{E},\subseteq)$ where $\overline{E}$ is the set of all downward closed directed, w.r.t. $\sqsubseteq$, subsets of E and $\subseteq$ is the set theoretical inclusion. We call each element of $\overline{E}$ an <u>ideal</u>. □

<u>Fact 1.2.13</u>

(1) The algebraic completion $(\overline{E},\subseteq)$ of a countable strict poset $(E,\sqsubseteq)$ is an ω-algebraic cpo. There is a canonical map $\tau:E\to\overline{E}$ s.t. $\tau(E)$ is the extension basis of $(\overline{E},\subseteq)$ and $(E,\sqsubseteq)\cong(\tau(E),\subseteq)$ as posets. Indeed $\tau(x)=\{e\in E\,|\,e\sqsubseteq x\}$ for all $x\in E$.

(2) Given an ω-algebraic cpo $(D,\sqsubseteq)$, the extension basis $(E_D,\sqsubseteq)$ is a countable strict poset and $(D,\sqsubseteq)\cong(\overline{E},\subseteq)$

(3) A poset is an ω-algebraic cpo iff it is the algebraic completion of a countable strict poset.

(4) Given two ω-algebraic cpo's $(D,\sqsubseteq)$ and $(D',\sqsubseteq')$, they are isomorphic iff $(E_D,\sqsubseteq)\cong(E_{D'},\sqsubseteq')$. □

It has been well-known that given two ω-basable cpo's D and D', The function space $[D\to D']$ need not be ω-based. But if D and D' are bounded complete, then so is the function space and it has an ω-basis. This leads to the following notion:

<u>Definition 1.2.14</u>

(1) An <u>ω-basable</u> <u>domain</u> is an ω-basable cpo which is bounded complete.

(2) An <u>ω-based</u> <u>domain</u> is an ω-continuous cpo (D,B) s.t. D is bounded complete.

(3) An <u>ω-algebraic</u> <u>domain</u> is an ω-algebraic cpo which is bounded complete. □

Bounded completeness of ω-basable cpo's can be characterized in terms of bases.

Fact 1.2.15

(1) An ω-continuous cpo $((D,\sqsubseteq),(B,\prec))$ is an ω-continuous domain iff every $\prec$-bounded finite subset of $(B,\prec)$ has a lub (w.r.t. $\sqsubseteq$) in D.

(2) Given an R-structure $(E,<)$, $(\overline{E},\subseteq)$ is an ω-basable domain iff for every finite $<$-bounded subset $S\subseteq E$, $\cup\{[x]\mid x\in S\}$ exists in $\overline{E}$. We call such R-structures BR-structures.

(3) An ω-algebraic cpo $(D,\sqsubseteq)$ is an ω-algebraic domain iff the extension basis $(E_D,\sqsubseteq)$ has bounded joins ,i.e., every finite bounded subset of E_D has a lub in E_D. Notice that the lub of a finite subset of E_D, if any, is compact in $(D,\sqsubseteq)$, thus is in E_D.

proof (1) "only if" part is trivial. We prove "if" part. Assume X is a bounded subset of D with an upper bound z. Then $\{e\in B\mid e\prec x$ for some $x\in X\}=\bigcup_{x\in X}B_x\subseteq B_z$. Therefore every finite subset S of $\bigcup_{x\in X}B_x$ is $\prec$-bounded. Thus by assumption $\sqcup S\in D$. Let $Y=\{\sqcup S\mid S$ is a finite subset of $\bigcup_{x\in X}B_x\}$, and $\{\sqcup S_1,\ldots,\sqcup S_n\}\subseteq Y$. Then $S_1\cup\ldots\cup S_n$ is a finite subset of $\bigcup_{x\in X}B_x$. Thus $\sqcup(S_1\cup\ldots\cup S_n)\in Y$. Evidently $\sqcup(S_1\cup\ldots\cup S_n)$ is an upper bound of $\{\sqcup S_1,\ldots,\sqcup S_n\}$. Thus Y is directed. Thus $\sqcup Y\in D$. But evidently $\sqcup X=\sqcup Y$. Thus D is bounded complete.

(2) Similarly to (1).

(3) Notice that for every $e\in E_D$, $e\prec x$ iff $e\sqsubseteq x$. Also remember that for every finite subset $X\subseteq E_D$, if $\sqcup X$ exists then $\sqcup X\in E_D$. ▨

There is a qualitative difference between (1),(2) of 1.2.15 and (3) of 1.2.15. Indeed for non-algebraic case, the argument is not quite purely that of bases, since we have to refer to the completion of the basis in order to talk about the lub's of bounded finite subsets. This gives rise to a question if the notion of bases of ω-based cpo's is quite adequate.

Fact 1.2.16

(1) Given ω-based domains $((D,\sqsubseteq),(B,\prec))$ and $((D',\sqsubseteq'),(B',\prec'))$ the following pairs are also ω-continuous domains.

$$((D\times D',\sqsubseteq_{\times}),(B\times B',\prec_{\times}))$$
$$((D+D',\sqsubseteq_{+}),(B+B',\prec_{+}))$$
$$(([D\to D'],\sqsubseteq_{\to}),([B\to B'],\prec_{\to}))$$

where $\sqsubseteq_{\times},\sqsubseteq_{+},\sqsubseteq_{\to}$ are partial orderings on $D\times D'$, $D+D'$, $[D\to D']$ as in 1.1.9, and $\prec_{\times},\prec_{+},\prec_{\to}$ are taken over $D\times D'$, $D+D'$, $[D\to D']$. $B\times B'$ and $B+B'$ are evident bases of $D\times D'$ and $D+D'$ respectively. $[B\to B']$ is the set of all possible lub's of finite subsets of the set $\{[b,b'] \mid b\in B, b'\in B'\}$ where $[b,b']=\lambda x\in D.$ <u>if</u> $b\prec x$ <u>then</u> b' <u>else</u> $\bot_{D'}$. We call such $[b,b']$ a <u>step function</u> from (D,B) to (D',B').

(2) Given ω-algebraic domains D and D', so are the following cpo's:

$$D\times D', \quad D+D', \quad \text{and } [D\to D'].$$

Indeed $E_{D\times D'}=E_D\times E_{D'}$, $E_{D+D'}=E_D+E_{D'}$, and $E_{[D\to D']}$ is the set of all existing finite joins of step functions from (D,E_D) to $(D',E_{D'})$.

☐

For ω-basable domains, we are interested only in whether they have ω-bases, and we do not care which bases these are. Therefore there should be no problem in defining embeddings and projections of ω-basable domains as we did for cpo's. But for ω-domains, we are interested in particular bases, thus it is more natural to think of those embeddings which 'embed' bases.

Definition 1.2.17

Given ω-based domains (D,B) and (D',B'), a pair of continuous maps $(e:D\to D', r:D'\to D)$ is called a <u>strong projection pair</u> from (D,B) to (D',B') iff $e(B)\subseteq B'$. r is called a <u>strong projection</u> and e is called a <u>strong embedding</u>.

☐

For ω-algebraic domains, strong projection pairs and continuous projection pairs coincide. Indeed we have:

Fact 1.2.18

Let D and D' be cpo's and ($f:D\to D'$, $g:D'\to D$) be a continuous projection pair. Then $f(E_D)\subseteq E_{D'}$.

proof Let $e\in E_D$ and X' be a directed subset of D'. Assume $f(e)\sqsubseteq \bigsqcup X'$, then $e=g\cdot f(e)\sqsubseteq g(\bigsqcup X')=\bigsqcup g(X')$. Therefore $e\sqsubseteq g(x')$ for some $x'\in X'$. Thus $f(e)\sqsubseteq f\cdot g(x')\sqsubseteq x'$. This means $f(e)\in E_{D'}$. Now let $e'\in E_{D'}$ and $X\subseteq D$ be directed s.t. $g(e')\sqsubseteq\bigsqcup X$. Then $f\cdot g(e')\sqsubseteq f(\bigsqcup X)=\bigsqcup f(X)$. Therefore $f\cdot g(e')\sqsubseteq f(x)$ for some $x\in X$. Thus $g(e')=g\cdot f\cdot g(e')\sqsubseteq g\cdot f(x)\sqsubseteq x$. This implies $g(e')\in E_D$. ☒

For ω-algebraic domains, continuous projection pairs (thus strong projection pairs) can be characterized in terms of bases.

Definition 1.2.19

Given ω-algebraic domains D and D', a map $i:E_D\to E_{D'}$ is an imbedding from E_D to $E_{D'}$ iff

(1) i is injective,

(2) for every finite subset $S\subseteq E_D$, $\bigsqcup S$ exists iff $\bigsqcup i(S)$ exists,

(3) $i(\bigsqcup S)=\bigsqcup i(S)$ for every finite subset $S\subseteq E_D$ s.t. $\bigsqcup S$ exists. □

The following theorem establishes that imbeddings and continuous projection pairs of ω-algebraic domains are the same.

Lemma 1.2.20

Let D and D' be ω-algebraic domains.

(1) Let $i:E_D \to E_{D'}$ be an imbedding, then the continuous extension $\bar{i}:D \to D'$ given by $\bar{i}(x)=\bigsqcup\{i(e)\mid e\in E_D, e\sqsubseteq x\}$ is a continuous embedding with adjoint $\bar{j}:D'\to D$ s.t. $\bar{j}(x')=\bigsqcup\{e\in E_D \mid i(e)\sqsubseteq x'\}$.

(2) Let $(i:D\to D', j:D'\to D)$ be a continuous projection pair then the restriction of $\bar{i}$ to E_D is an imbedding from E_D to $E_{D'}$.

proof (1) $\bar{i}$ evidently is continuous, well-defined and is an extension of i. Now let F be a finite subset of $\{e\in E_D \mid i(e)\sqsubseteq i(x')\}$. Then $i(F)$ is bounded by x', thus $\bigsqcup i(F)\in E_{D'}$. Since i is an imbedding, $\bigsqcup F\in E_D$. Evidently $i(\bigsqcup F)=\bigsqcup i(F)\sqsubseteq x'$. Thus $\bigsqcup F\in\{e\in E_D\mid i(e)\sqsubseteq x'\}$. Thus $\{e\in E_D \mid i(e)\sqsubseteq x'\}$ is directed. Thus $\bar{j}$ is well-defined. Let $X'\subseteq D'$ be directed. Then:

$$\bar{j}(\bigsqcup X')=\bigsqcup\{e\in E_D \mid i(e)\sqsubseteq \bigsqcup X'\}$$
$$=\bigsqcup\{e\in E_D\mid i(e)\sqsubseteq x' \text{ for some } x'\in X'\}=\bar{j}(X').$$

Thus $\bar{j}$ is continuous. Now:

$$\bar{j}\cdot\bar{i}(x)=\bigsqcup\{e\in E_D\mid i(e)\sqsubseteq\bigsqcup\{i(e)\mid e\sqsubseteq x\}\}$$
$$=\bigsqcup\{e\in E_D\mid i(e)\sqsubseteq i(e') \text{ for some } e'\sqsubseteq x\}$$
$$=\bigsqcup\{e\in E_D\mid e\sqsubseteq x\}=x.$$

Also
$$\bar{i}\cdot\bar{j}(x')=\bigsqcup\{i(e)\mid e\in E_D, e\sqsubseteq\bigsqcup\{e\in E_D\mid i(e)\sqsubseteq x'\}\}$$
$$=\bigsqcup\{i(e)\mid e\in E_D, i(e)\sqsubseteq x'\}\sqsubseteq x'.$$

(2) Evident. ∎

Note that if i is an imbedding then i preserves all existing (in E_D) lub's of E_D, for $\bar{i}$ is a continuous embedding. This suggests the following prospective notion of imbeddings among bases of ω-based domains. Given ω-continuous domains (D,B) and (D',B'), an 'imbedding' from B to B' is an injection $i:B\to B'$ s.t. for every $S\subseteq B$, $\bigsqcup S\in B$ iff $\bigsqcup i(S)\in B'$, and $i(\bigsqcup S)=\bigsqcup i(S)$ once $\bigsqcup S$ exists in B.

Evidently for every (weaker) strong projection pair $(\bar{i},\bar{j})$ from D to D', the restriction of $\bar{i}$ to B is such an imbedding. Conversely, given an imbedding i from B to B', let $\bar{i}:D\to D'$ be $i(x)=\sqcup\{i(b)\mid b\in B, b\prec x\}$, then $\bar{i}$ is a continuous extension of i, since for $b\in B$,

$$\bar{i}(b)=\sqcup i(B_b)=i(\sqcup B_b)=i(b).$$

Thus $\bar{i}(B)\subseteq B'$. Furthermore it can readily be seen that $\bar{i}$ is a continuous embedding with the adjoint $\bar{j}$ s.t. $\bar{j}(x')=\sqcup\{b\in B\mid i(b)\sqsubseteq x'\}$. Therefore imbeddings of bases characterise strong projection pairs among ω-based domains.

At this point, the idea of bases and that of R-structures seem to differ. Let E and E' be BR-structures. Define $i:E\to E'$ to be an injection s.t. $\cup([S])\in\bar{E}$ iff $\cup[i(S)]\in\bar{E}'$ for every $S\subseteq E$; and $i(\cup[S])=\cup[i(S)]$ once $\cup([S])\in\bar{E}$. Notice that this map i was used as an embedding of BR-structures in Smyth[22]. But we do not know if $\bar{i}([E])\subseteq[E']$ where $\bar{i}:\bar{E}\to\bar{E}'$ is defined by $\bar{i}(x)=\cup\{i([a])\mid a\in x\} = \cup i[x]$ for all $x\in\bar{E}$.

Given ω-based domains (D,B) and (D',B'), let (i,j) be a continuous projection pair w.r.t. $\sqsubseteq$ (not $\prec$) from B to B'. Then it can readily be seen that i is an imbedding and $(\bar{i},\bar{j})$ is a strong projection pair from D to D'. Thus continuous projection pairs among ω-bases characterize strong projection pairs.

So far we have observed that the strong projection pairs can be characterized in terms of bases. But we do not know if this can be done in terms of R-structures yet. All we can say is that it would potentially be problematic to get rid of $\sqsubseteq$ orderings from the ω-bases and obtain R-structures. Precisely speaking a basis seems to be $(B,\prec,\sqsubseteq)$ rather than $(B,\prec)$.

Fact 1.2.21

Let $<D_m,(f_m,f_m^R)>$ be an ω-sequence of continuous projection pairs of ω-basable domains. The inverse limit D_∞ of it is an ω-basable domain. Indeed if B_m is an ω-basis of D_m for each $m\in N$, then $B_\infty=\cup_m f_{m\infty}(B_m)$ is an ω-basis of D_∞. Evidently $f_{m\infty}(B_m)\subseteq B_\infty$. But $f_{m\infty}^R(B_\infty)$ is not necessarily a subset of B_m. ☒

Definition 1.2.22

Let $<(D_m,B_m),(f_m,f_m^R)>$ be an ω-sequence of strong projection pairs of ω-based domains. By the inverse limit of it, we mean (D_∞,B_∞) where $B_\infty=\cup_m f_{m\infty}(B_m)$. □

Fact 1.2.23

If $<(D_m,B_m),(f_m,f_m^R)>$ is an ω-sequence of strong projection pairs of ω-based domains, then (D_∞,B_∞) is an ω-based domain. Furthermore $(f_{m\infty},f_{m\infty}^R)$ is a strong projection pair. ☒

Things are much simpler for algebraic cases since continuous projection pairs and strong projection pairs coincide.

Fact 1.2.24

If $<D_m,(f_m,f_m^R)>$ is an ω-sequence of continuous projection pairs of ω-algebraic domains, the inverse limit D_∞ is an ω-algebraic domain. Indeed $E_{D_\infty}=\cup_m E_{D_m}$. ☒

By almost the same argument as for cpo's, we can easily observe that isomorphism is a good criterion for identifying two ω-basable (ω-algebraic) domains.

Given two ω-based domains (D,B) and (D',B') we say they are isomorphic iff $B\cong B'$. Notice that by virtue of 1.2.8, $(D,B)\cong(D',B')$ implies $D\cong D'$. It is now very easy to observe that this isomorphic relation is a good criterion for identifying two ω-based domains, for it is preserved under domain constructors.

<u>Fact 1.2.25</u> (Scott [20],Plotkin [12],Smyth [22])

A poset D is an ω-basable domain iff it is isomorphic to h(X) for some ω-algebraic domain X and some continuous idempotent $h:X\to X$. Indeed D has an ω-basis $h(E_X)$, and there is a single ω-algebraic domain, say P, s.t. every ω-basable domain is isomorphic to h(P) for some continuous idempotent $h:P\to P$. We call such P a <u>universal domain</u> of the class of ω-basable domains. ☒

At this point, the notion of ω-based domain seems to be less interesting than that of ω-basable domains. In fact it is not known yet if such a universal domain exists for the class of ω-based domains. The point here is that, given an ω-based domain (D,B), even if we have $h:P\to P$ satisfying $h(P)\cong D$, we do not know if we have h such that $h(E_P)\cong B$ or not.

One could argue that if we require bases to be closed under finite joins, we could avoid this difficulty and at the same time we could solve the problem for bounded completeness. But why (intuitively) must this be so? Can we find any convincing motivation (rather than technical reason) for this restriction?

We have observed several problems on ω-based domains. We claim that all of these problems are essentially due to the lack of concrete examples of ω-based domains. So far most domains which appear to be important for computer science are ω-algebraic. Only the interval lattice (See Scott[32]) seems to be the one which indicates the necessity of ω-based domains. Indeed there should always be some problems in obtaining abstract notions from only a few examples. In other words such attempts tend to be too arbitrary.

We hope that these observations will justify our omission of the non-algebraic case in the rest of this dissertation.

1.3 SFP Objects

Remember that in the previous section, bounded completeness was introduced as a sufficient condition for making function spaces basable. But as Plotkin showed this condition is not preserved under Plotkin's power domain construction [10]. Furthermore he showed that there exists a weaker condition which can be preserved under all interesting domain constructions and which makes function spaces basable. We will review this condition in this section.

Definition 1.3.1

(1) Let D be a poset and X be a subset of D. We say $u \in D$ is a minimal upper bound (mub) of X iff u is an upper bound of X and for every upper bound v of X, $v \sqsubseteq u$ implies v=u. We will write $U_D(X)$ to denote the set of all mub's of X. Furthermore $U_D^*(X)$ denote the least subset of D satisfying:

$$U_D^*(X) \supseteq X, \text{ and}$$

$$U_D(Y) \subseteq U_D^*(X) \text{ for all } Y \subseteq U_D^*(X).$$

(2) $U_D(X)$ is said to be complete iff whenever v is an upper bound of $X \subseteq D$, then there exists $u \in U_D(X)$ s.t. $u \sqsubseteq v$.

(3) A poset D is said to be bounded m-complete iff $U_D(X)$ is complete for every subset $X \subseteq D$. Also D is said to have bounded m-joins iff $U_D(X)$ is complete for every finite subset X of D. □

Notice that if $X \subseteq D$ has a lub $\sqcup X$, then $U_D(X) = \{\sqcup X\}$. Therefore bounded m-completeness is a generalization of bounded completeness and bounded m-join is a generalization of bounded join.

Definition 1.3.2

A countably algebraic cpo D is an SFP object iff for every finite subset X of E_D, $U_D(X)$ is complete and $U_D^*(X)$ is finite.

□

According to this definition, every finite cpo is an SFP object.

Lemma 1.3.3

Let D be an algebraic cpo and E_D be the basis of D. Then we have:

(1) For every finite subset X of E_D, $\mathcal{U}_D(X)\subseteq E_D$ and in case D is an SFP object, $\mathcal{U}^*_D(X)\subseteq E_D$.

(2) For every finite subset X of E_D, $\mathcal{U}_D(X)=\mathcal{U}_{E_D}(X)$.

(3) For every finite subset X of E_D, $\mathcal{U}_D(X)$ is complete in D iff it is complete in E_D.

proof (1) Let $u\in\mathcal{U}_D(X)$. By algebraicity, $\{e\in E_D|e\sqsubseteq u\}=J_u$ is directed and $u=\bigsqcup J_u$. Since u is an upper bound of X, X is a finite subset of J_u. Thus there exists an upper bound v of X in J_u. Thus $v\sqsubseteq u$. By minimality of u, u=v. Thus $u\in E_D$. Therefore $\mathcal{U}_D(X)\subseteq E_D$. Now define:

$$\mathcal{U}^0_D(Y)=\phi$$
$$\mathcal{U}^{r+1}_D(Y)=\{\mathcal{U}_D(Y')\,|\,Y'\subseteq\mathcal{U}^r_D(Y)\}\cup Y$$

where $Y\subseteq E_D$. Then by induction on r we have:

$$\mathcal{U}^r_D(Y)\subseteq\mathcal{U}^{r+1}_D(Y)\subseteq\mathcal{U}^*_D(Y).$$

Also if $\mathcal{U}^r_D(Y)=\mathcal{U}^{r+1}_D(Y)$, then $\mathcal{U}^*_D(Y)=\mathcal{U}^r_D(Y)$. If D is SFP then $\mathcal{U}^*_D(X)$ is finite, therefore $\mathcal{U}^*_D(X)=\mathcal{U}^r_D(X)$ for some $r\in N$. Therefore $\mathcal{U}^*_D(X)\subseteq E_D$.

(2) Evident.

(3) Evident. ∎

Now we have the following alternative characterization of SFP objects, as an immediate corollary to 1.3.3.

Corollary 1.3.4

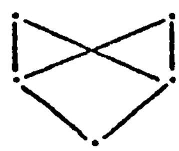
A countably algebraic cpo D is an SFP object iff E_D has bounded m-joins and for every finite subset X of E_D, $\mathcal{U}^*_{E_D}(X)$ is finite.

∎

Remember that an algebraic cpo D is bounded complete iff E_D has bounded joins. Furthermore we have observed that bounded join property implies bounded m-join property. Thence showing "every ω-algebraic domain is an SFP object" amounts to showing that for every finite subset X of E_D, $U^*_{E_D}(X)$ is finite. But bounded join property of E_D yields:

$$U^*_{E_D}(X)=U^2_{E_D}(X)=X\cup\{\sqcup Y \mid Y\subseteq(X\cup\{\bot\})\}$$

for every finite subset X of E_D. Therefore $U^*_{E_D}(X)$ is finite. Also the following finite cpo is not bounded complete:



In summary, we have established:

Fact 1.3.5

The class of SFP objects properly contains the class of ω-algebraic domains. ☐

The following alternative characterization of SFP objects due to Plotkin [10] is more comprehensive (at least intuitively) and easier to handle in many situations.

Fact 1.3.6

A cpo D is an SFP object iff it is the inverse limit of an ω-sequence $<D_m,(f_m,f_m^R)>$ of continuous projection pairs of finite cpo's. ☐

Readers are refered to Plotkin [10] for the details of this proof. Here we will draw readers attention to the canonical sequence of projections which Plotkin introduced to prove 1.3.6. Assume we are given an SFP object D with an indexing $\varepsilon:N\to E_D$ of the basis E_D s.t. $\varepsilon(0)=\bot$. By the canonical sequence of D w.r.t. ε we mean the ω-sequence $\langle D_m,(f_m,f_m^R)\rangle$ of continuous projection pairs of finite cpo's where $D_m=U^*_{E_D}(\{\varepsilon(o),..,\varepsilon(m)\})$ and $f_m:D_m\to D_{m+1}$ is a continuous embedding defined by $f_m(x)=x$ with the corresponding projection $f_m^R:D_{m+1}\to D_m$ s.t. $g_m(y)=\sqcup\{x\in D_m|x\sqsubseteq y\}$. Notice that $D_m\subseteq D_{m+1}\subseteq D$ and $D\cong D_\infty$ via $\Omega:E_{D_\infty}\to E_D$ s.t. $\Omega(\langle x_m\rangle)=\sqcup x_m$. It is obvious that the adjoint (inverse) of Ω is $\Omega^R:E_D\to E_{D_\infty}$ given by $\Omega^R(x)=f_{m\infty}(x)$ where $m=\mu k.[x\in D_k]$.

A countable poset E is called a <u>finitary poset</u> iff it has bottom and bounded m-joins and for every finite subset X of E, $U^*_E(X)$ is finite. By 1.3.4 we have the following completion theorem.

<u>Theorem 1.3.7</u>

(1) If $(E,\sqsubseteq)$ is a finitary poset then the algebraic completion $(\bar{E},\subseteq)$ is an SFP object.

(2) If $(D,\sqsubseteq)$ is an SFP object then $(E_D,\sqsubseteq)$ is a finitary poset and $(\bar{E},\subseteq)=(D,\sqsubseteq)$.

(3) A cpo is an SFP object iff it is isomorphic to the algebraic completion of a finitary poset.

Let $\langle P_m,(r_m,r_m^R)\rangle$ and $\langle Q_m,(s_m,s_m^R)\rangle$ be ω-sequences of continuous projection pairs of cpo's s.t. $P=\varprojlim\langle P_m\rangle$ and $Q=\varprojlim\langle Q_m\rangle$. Assume $\langle(r_{m\infty},r_{m\infty}^R)\rangle$ and $\langle(s_{m\infty},s_{m\infty}^R)\rangle$ are universal cocones of $\langle P_m,(r_m,r_m^R)\rangle$ and $\langle Q_m,(s_m,s_m^R)\rangle$ respectively. Define continuous maps:

$$F_m:[P_m\to Q_m]\to[P_{m+1}\to Q_{m+1}]$$
$$R_m:P_m\times Q_m\to P_{m+1}\times Q_{m+1}$$
$$U_m:P_m+Q_m\to P_{m+1}+Q_{m+1}$$

by:

$$F_m(f)=s_m\cdot f\cdot r_m^R$$
$$R_m=r_m\times s_m$$
$$U_m=r_m+s_m\ .$$

Then they are continuous embeddings with the adjoints:

$$F_m^R(f)=s_m^R\cdot f\cdot r_m,$$
$$R_m^R=r_m^R\times s_m^R,$$
$$U_m^R=r_m^R+s_m^R.$$

Therefore $<[P_m\to Q_m],(F_m,F_m^R)>$, $<P_m\times Q_m,(R_m,R_m^R)>$, and $<P_m+Q_m,(U_m,U_m^R)>$ are ω-sequences of continuous projection pairs of cpo's.

<u>Fact 1.3.8</u>

(1) $\underleftarrow{\lim}<[P_m\to Q_m],(F_m,F_m^R)>\cong[P\to Q]$.

(2) $\underleftarrow{\lim}\ <P_m\times Q_m,(R_m,R_m^R)>\cong P\times Q$.

(3) $\underleftarrow{\lim}\ <P_m+Q_m,(U_m,U_m^R)>\cong P+Q$.

<u>proof</u> (outline) Assume $<(F_{m\infty},F_{m\infty}^R)>$, $<(R_{m\infty},R_{m\infty}^R)>$, and $<(U_{m\infty},U_{m\infty}^R)>$ are universal cocones.

(1) For each $f\in[P\to Q]$ let $f_{(m)}=s_{m\infty}^R\cdot f\cdot r_{m\infty}$. Then $f_{(m)}\in[P_m\to Q_m]$ and $<f_{(m)}>\in\underleftarrow{\lim}<[P_m\to Q_m],(F_m,F_m^R)>$. Define $\Phi:[P\to Q]\to\underleftarrow{\lim}<[P_m\to Q_m]>$ by $\Phi(f)=<f_{(m)}>$. Furthermore define $\Phi^R:\underleftarrow{\lim}<[P_m\to Q_m]>\to[P\to Q]$ by: $\Phi^R(f)(x)=\bigsqcup s_{m\infty}\cdot f_m(x_m)$ where $f_m=F_{m\infty}^R(f)\in[P_m\to Q_m]$ and $x_m=r_{m\infty}^R(x)\in P_m$. It can readily be seen that (Φ,Φ^R) is an isomorphism pair.

(2) Given $(x,y)\in P\times Q$, evidently $(x_m,y_m)\in P_m\times Q_m$. Furthermore $<(x_m,y_m)>$ belongs to $\underleftarrow{\lim}<P_m\times Q_m>$ Define $\P:P\times Q\to\lim<P_m\times Q_m>$ by $\P((x,y))=<(x_m,y_m)>$. Conversely for every $<(a_m,b_m)>$ in $\underleftarrow{\lim}<P_m\times Q_m>$ define $\P^R(<(a_m,b_m)>)=(\bigsqcup_m r_{m\infty}(a_m),\bigsqcup_m s_{m\infty}(b_m))$. Evidently $(\P,\P^R)$ is an isomorphism pair.

(3) Define $\sigma_m:P+Q\to P_m+Q_m$ by:

$$\sigma_m((i,x))=\underline{\text{if}}\ i=0\ \underline{\text{then}}\ (i,r_{m\infty}^R(x))\ \underline{\text{else}}\ (i,s_{m\infty}^R(x)).$$

Define $\Sigma:P+Q\to\underleftarrow{\lim}<P_m+Q_m>$ by $\Sigma((i,x))=<\sigma_m((i,x))>$. Furthemore define $\Sigma^R:\underleftarrow{\lim}<P_m+Q_m>\to P+Q$ by:

$$^R(<(i,c_m)>)=\underline{\text{if}}\ i=0\ \underline{\text{then}}\ (i,\bigsqcup_m r_{m\infty}(c_m))$$
$$\underline{\text{else}}\ (i,\bigsqcup_m s_{m\infty}(c_m)).$$

Then evidently (Σ,Σ^R) is an isomorphism pair.

∎

In the above assume P_m and Q_m are finite cpo's, this means P and Q are SFP objects. Evidently $[P_m \to Q_m]$, $P_m \times Q_m$, $P_m + Q_m$ are finite cpo's. Thus by 1.3.8, $[P \to Q]$, $P \times Q$, $P+Q$ are SFP objects. In summary we have established:

<u>Fact 1.3.9</u>

The class of SFP objects is closed under the domain constructors $\to$, $\times$, and $+$.

Even though the proof of 1.3.9, which used the alternative characterization 1.3.6 of SFP objects is very simple, it does not tell much about how the domain constructors $\to$, $\times$, and $+$ operate on bases. In the next theorem we will study this:

<u>Theorem 1.3.10</u>

Let P and Q be SFP objects. Then we have:

(1) The basis of $[P \to Q]$ is the set of all possible finite joins of step functions

(2) $E_{P \times Q} = E_P \times E_Q$

(3) $E_{P+Q} = E_P + E_Q$.

<u>proof</u> Let $\langle P_m, (r_m, r_m^R) \rangle$ and $\langle Q_m, (s_m, s_m^R) \rangle$ be ω-sequences of continuous projection pairs of finite cpo's s.t. $P \cong \varprojlim \langle P_m, (r_m, r_m^R) \rangle$ and $Q \cong \varprojlim \langle Q_m, (s_m, s_m^R) \rangle$.

(1) Let Φ, Φ^R, F_m, F_m^R be as in 1.3.8. It can readily be seen that $\Phi([r_{m\infty}(a), s_{m\infty}(b)]) = F_{m\infty}([a,b])$ for every $a \in P_m$ and $b \in Q_m$. Notice that every compact element of $\varprojlim \langle [P_m \to Q_m], (F_m, F_m^R) \rangle$ is $F_{m\infty}(f)$ for some $f \in [P_m \to Q_m]$ and some $m \in N$. But evidently for every $f \in [P_m \to Q_m]$ we have:

$$f = \sqcup \{[x_1, f(x_1)], .., [x_n, f(x_n)]\}$$

where $P_m = \{x_1, .., x_n\}$. Therefore we have:

$$e = F_{m\infty}(f) = F_{m\infty}(\sqcup \{[x_1, f(x_1)], .., [x_n, f(x_n)]\})$$
$$= \sqcup \{F_{m\infty}([x_1, f(x_1)]), .., F_{m\infty}([x_n, f(x_n)])\}.$$

Thus $\phi^R(e)=\sqcup\{\phi^R\cdot F_{m\infty}([x_1,f(x_1)]),..,\phi^R\cdot F_{m\infty}([x_n,f(x_n)])\}$

$=\sqcup\{[r_{m\infty}(x_1),s_{m\infty}\cdot f(x_1)],..,[r_{m\infty}(x_n),s_{m\infty}\cdot f(x_n)]\}.$

Therefore every compact element of $[P\to Q]$ is a finite join of step functions.

Conversely every step function of $[P\to Q]$ is compact in $[P\to Q]$.

Therefore every finite join which exists of step functions is compact in $[P\to Q]$.

(2), (3) Evident. ☒

The following lemma is a generalization of (1)-1.1.5.

Lemma 1.3.11

Let D and D' be cpo's. Let $(i:D\to D', j:D'\to D)$ be a monotone projection pair. Then for every $X\subseteq D$:

$$U_{D'}(i(X))=i(U_D(X)).$$

proof Let $u\in U_{D'}(i(X))$. For every $x\in X$, $u\sqsupseteq i(x)$. Thus $j(u)\sqsupseteq j\cdot i(x)=x$. Thus $j(u)\sqsupseteq X$. If $v\sqsupseteq X$ and $v\sqsubseteq j(u)$ then $i\cdot j(u)\sqsupseteq i(v)$. Thus $u\sqsupseteq i(v)$. But $i(v)\sqsupseteq i(X)$. Thus by the minimality of u, $u=i(v)$. Therefore $v=j(u)$. Thus $j(u)\in U_D(X)$. Since $i\cdot j\sqsubseteq id_{D'}$, $i\cdot j(u)\sqsubseteq u$. But $i\cdot j(u)\sqsupseteq i(X)$ for $j(u)\sqsupseteq X$. Thus by the minimality of u, $i\cdot j(u)=u$. Therefore $u\in i(U_D(X))$ for $u=i\cdot j(u)$ and $j(u)\in U_D(X)$. Thence we have established $U_{D'}(i(X))\subseteq i(U_D(X))$. Conversely let $u\in U_D(X)$. Evidently $i(u)\sqsupseteq i(X)$. Let $v\sqsupseteq i(X)$ and $v\sqsubseteq i(u)$. Then $j(v)\sqsubseteq j\cdot i(u)=u$. Evidently $j(v)\sqsupseteq X=i\cdot j(v)$. By the minimality of u, $u=j(v)$. Thus $i(u)=i\cdot j(v)\sqsubseteq v$. Thus $i(u)=v$. Therefore $i(u)$ is a minimal upper bound of $i(X)$. Therefore we have $i(U_D(X))\subseteq U_{D'}(i(X))$. ☒

We will define continuous projection pairs among SFP objects as we did for cpo's, since every SFP object is a cpo. By virtue of 1.2.18, every continuous embedding does embed bases of SFP objects which it embeds.

Fact 1.3.12

Let $<D_m,(f_m,f_m^R)>$ be an ω-sequence of continuous projection pairs of SFP objects. Then the inverse limit $D_\infty = \varprojlim <D_m,(f_m,f_m^R)>$ is again an SFP object. Indeed $E_{D_\infty} = \bigcup_m f_m(E_{D_m})$ where $<(f_{m\infty},f_{m\infty}^R)>$ is the universal cocone of $<D_m,(f_m,f_m^R)>$. ☐

By the same arguments as for cpo's, it can readily be seen that the isomorphic relation is a good criterion for identifying two SFP objects.

It shoud be noted that the continuous projection pairs can be characterized as "imbeddings" of bases.

Definition 1.3.13

Let D and D' be SFP objects. A map $i:E_D \to E_{D'}$ is an imbedding from E_D to $E_{D'}$ iff

(1) i is injective,

(2) For every finite subset $S \subseteq E_D$, $Card(U_{E_D}(S)) = Card(U_{E_{D'}}(i(S)))$,

(3) $e \in U_{E_D}(S)$ iff $i(e) \in U_{E_{D'}}(i(S))$. ☐

Notice that this definition coincides with 1.2.19 if D and D' are ω-algebraic domains.

Lemma 1.3.14

Let D and D' be SFP objects.

(1) Let $i:E_D \to E_{D'}$ be an imbedding, then the continuous extension $\bar{i}:D \to D'$ given by $\bar{i}(x) = \sqcup\{i(e) \mid e \in E_D, e \sqsubseteq x\}$ is a continuous embedding with the adjoint $\bar{j}:D' \to D$ s.t. $\bar{j}(x') = \sqcup\{e \in E_D \mid i(e) \sqsubseteq x'\}$.

(2) $i = \bar{i} \restriction E_D$ is an imbedding if $\bar{i}:D \to D'$ is a continuous embedding.

proof Except the well-definedness of $\bar{j}$ the proof is the same as for 1.2.20. Let F be a finite subset of $R_{x'} = \{e \in E_D \mid i(e) \sqsubseteq x'\}$. Then there is a mub m' of i(F) and $m' \in E_{D'}$. Since i is an imbedding there is a mub m of F s.t. $m \in E_D$ and $i(m) = m'$. Since $m' = i(m) \sqsubseteq x'$, $m \in R_{x'}$. Thus $R_{x'}$ is directed. ☐

Now we review Plotkin power domain construction [10].

Definition 1.3.15

Let D be an ω-algebraic cpo and T be a non-empty (node-)labeled finitary tree satisfying:

(1) For each node t, the label $l(t) \in E_D$;

(2) T has no terminating branches;

(3) If t' is a descendant of t in T then $l(t) \sqsubseteq l(t')$.

Let L be the function which assignes to each (infinite) path π through, the lub of labels occuring along π. We say that T is a generating tree over D, which generates the set $S_T = \{L(\pi) \mid \pi \text{ is a path through } T\} \subseteq D$. A set $S \subseteq D$ is finitely generable (f.g.) if it is generated by some tree T. The class of f.g. subsets of D is denoted by $\mathcal{F}(D)$. □

Notice that every finite subset of D is in $\mathcal{F}(D)$. By M(D), we denote the set of all non-empty finite subsets of E_D.

Definition 1.3.16

Let D be an ω-algebraic cpo, $A \in M(D)$ and $S, S' \in \mathcal{F}(D)$.

Define:

(1) $A \sqsubseteq_M S \Leftrightarrow (\forall x \in S.\exists a \in A.a \sqsubseteq x) \& (\forall a \in A.\exists x \in S.a \sqsubseteq x)$;

(2) $S \precsim_M S' \Leftrightarrow \forall A \in M(D).A \sqsubseteq_M S \Rightarrow A \sqsubseteq_M S'$ □

It can readily be seen that for $A, A' \in M(D)$, $A \precsim_M A'$ iff $A \sqsubseteq_M A'$. Also $(\mathcal{F}(D), \precsim_M)$ is a pre-ordered set. Given a pre-ordered set $(P, \leq)$, $[P]$ denotes the quotient poset $(P/\equiv_P, \leq/\equiv_P)$ where $\equiv_P$ is the canonical equivalence relation over P. [x]denotes the equivalence class of x and for $S \subseteq P$, $[S] = \{[x] \mid x \in S\} \subseteq [P]$. For every monotone $f: P \to P'$, $[f]:[P] \to [P']$ is given by $[f][x] = [f(x)]$.

Theorem 1.3.17

$[(\mathcal{F}(D), \sqsubseteq_M)]$, in short $\mathcal{F}[D]$, is an ω-algebraic cpo with the basis $[(M(D), \sqsubseteq_M)]$, in short $M[D]$. We call $\mathcal{F}[D]$ the (strong) power domain of D.

Smyth [30] presented a weaker power domain construction which has the advantage of preserving the bounded completeness. But this construction identifies too many elements. Plotkin power domain construction does preserve not bounded completeness but SFP condition.

Theorem 1.3.18

If D is the inverse limit of an ω-sequence $\langle D_m, (p_m, p_m^R)\rangle$ of projection pairs of ω-algebraic cpo's, then $\langle \mathcal{F}[D_m], ([p_m], [p_m^R])\rangle$ is an ω-sequence of projection pairs of ω-algebraic cpo's and $\mathcal{F}[D]$ is isomorphic to the inverse limit of it. ∅

Corollary 1.3.19

If D is an SFP object, then so is $\mathcal{F}[D]$.

∅

1.4 Initial Non-effective Solutions

In the previous sections, we studied several classes of domains, namely cpo's, ω-basable domains, ω-algebraic domains, and SFP objects. All of these classes admit canonical solutions to recursive domain equations. Smyth and Plotkin [24] developed a categorical theory which unifies the arguments for all of these classes. We will briefly review their theory.

Definition 1.4.1

Let $\underline{K}$ be a category and $F:\underline{K}\to\underline{K}$ be an endofunctor. A fixed point of F is a pair (A,α) where A is an object of $\underline{K}$ and $\alpha:FA\to A$ is an isomorphism of $\underline{K}$. An F-algebra is a pair (A,α) where $A\in\underline{K}$ and $\alpha:FA\to A$ is a $\underline{K}$-morphism. Given F-algebras (A,α) and (A',α'), an F-homomorphism $f:(A,\alpha)\to(A',\alpha')$ is a $\underline{K}$-morphism $f:A\to A'$ s.t.:

$$f\cdot\alpha=\alpha'\cdot Ff.$$

It can readily be seen that the class of F-algebras and the class of F-homomorphisms form a category. □

Fact 1.4.2

The initial F-algebra, if it exists, is also the initial fixed point of F. ☐

Fact 1.4.3

Suppose $\underline{K}$ has the initial object $\perp$. Let Δ be an ω-codiagram $\langle F^n(\perp),F^n(\perp_{F\perp})\rangle$ where $\perp_{F\perp}$ is the unique $\underline{K}$-morphism from $\perp$ to $F\perp$. Also suppose that $\langle\mu_n:F^n\perp\to A\rangle$ is a colimiting cocone. Suppose too that $\langle F\mu_n:F^{n+1}\perp\to FA\rangle$ is a colimiting cocone of the ω-codiagram $F\Delta=\langle F^{n+1}\perp,F^{n+1}\perp_{F\perp}\rangle$. Then the initial F-algebra exists. ☐

Definition 1.4.4

A category $\underline{K}$ is an ω-category iff it has an initial object denoted by $\perp_{\underline{K}}$, and every ω-codiagram has a colimit. A functor

$F:\underline{K}\to\underline{L}$ is ω-continuous iff it preserves all existing colimits (more precisely, colomiting cocones). □

<u>Corollary 1.4.5</u>

Let $\underline{K}$ be an ω-category and $F:\underline{K}\to\underline{K}$ be an ω-continuous functor then the initial F-algebra exists. ▨

<u>Fact 1.4.6</u>

(1) Let $\underline{K}$ and $\underline{L}$ be ω-categories, then so is $\underline{K}\times\underline{L}$.

(2) A functor $F:\underline{K}\times\underline{L}\to\underline{M}$ is ω-continuous iff it is ω-continuous in both $\underline{K}$ and $\underline{L}$. ▨

By CPO^P, ω-BD^P, ω-AD^P, SFP^P we denote the category of cpo's and continuous projection pairs, the category of ω-basable domains and continuous projection pairs, the category of ω-algebraic domains and continuous projection pairs, and the category of SFP objects and continuous projection pairs respectively.

It can readily be seen that CPO^P, ω-BD^P, ω-AD^P, and SFP^P are all ω-categories. Indeed for each of them, the initial object is the singleton, ω-codiagrams are ω-sequences of continuous projection pairs, ω-colimits are the inverse limits, and the colimiting cocones are the universal cocones. Furthermore the domain constructors $\times$, $+$, and $\to$ are all ω-continuous functors. More precisely, for example, the functor $\to:CPO^P\times CPO^P\to CPO^P$ defined on objects by $\to(D_1,D_2)=[D_1\to D_2]$ and defined on morphisms by

$$\to((p,p^R):D_1\to D_2,(q,q^R):D_1'\to D_2')=(\lambda f\in[D_1\to D_1'].q\cdot f\cdot p^R,\lambda h\in[D_2\to D_2'].q^R\cdot h\cdot p)$$

is an ω-continuous functor. Therefore we can solve recursive domain equations within CPO^P, ω-BD^P, ω-AD^P, and SFP^P.

Let $CPO^{(*)}$, ω-$BD^{(*)}$, ω-$AD^{(*)}$, and $SFP^{(*)}$ denote the category of cpo's and continuous (strict) functions, the category of ω-basable domains and continuous (strict) functions, and the category of ω-algebraic domains and continuous (strict) functions, and SFP objects and continuous (strict) functions. In the previous section we stressed that these combinations are very natural and interesting. Also in the above we showed that CPO^P, ω-BD^P, ω-AD^P, and SFP^P enable us to solve recursive domain equations. Recent development of Smyth and Lehmann [23] showed that initial solutions of recursive domain equations associate operations to solution domains, thence derive data types. In this context, relations between $CPO^{(\ast)}$ and CPO^P, ω-$BD^{(\ast)}$ and ω-BD^P, ω-$AD^{(\ast)}$ and ω-AD^P, and $SFP^{(\ast)}$ and SFP^P appear to be important. Smyth and Plotkin [24] studied this problem in great detail . We will review their work.

Wand [29] noticed that $CPO^{(\ast)}$, ω-$BD^{(\ast)}$, ω-$AD^{(\ast)}$ and $SFP^{(\ast)}$ have richer information in the morphism sets, and presented a notion of O-categories which enable us to make use of these informations.

Definition 1.4.7

A category $\underline{K}$ is an O-category iff every hom-set is a poset in which every ascending non-empty ω-chain has a lub and the composition of morphisms is ω-continuous w.r.t. this partial ordering. □

Note that if $\underline{K}$ is an O-category, so is $\underline{K}^{OP}$ where $f^{OP} \sqsubseteq g^{OP}$ iff $f \sqsubseteq g$ for every $\underline{K}$-morphism f and g. Also if $\underline{L}$ is an O-category then so is $\underline{K} \times \underline{L}$ where $(f,g) \sqsubseteq (f',g')$ iff $f \sqsubseteq f'$ and $g \sqsubseteq g'$.

Definition 1.4.8

Let $\underline{K}$ be an O-category and let $A \xrightarrow{f} B \xrightarrow{g} A$ be arrows s.t. $g \cdot f = id_A$ and $f \cdot g \sqsubseteq id_B$. Then we say that (f,g) is a projection pair from A to B. g is called a projection and f is called an embedding □

Note that this categorical formulation of projection pairs reflects the remarks just before 1.1.6. First we fix up a category $\underline{K}$ and define projection pairs in this category. Therefore we will not obtain a pair (f,g) of monotone maps s.t. $g \cdot f = id_A$, $f \cdot g \sqsubseteq id_B$ and g is not continuous from the category CPO*, as a projection pair. Indeed in CPO* the projection pairs are exactly continuous projection pairs.

Fact 1.4.9

Let (f,g) and (f',g') be projection pairs from A to B in an O-category $\underline{K}$. Then $f \sqsubseteq f'$ iff $g' \sqsubseteq g$. □

Notice that it follows from 1.4.9 that one half of a projection pair uniquely determines the other. Thus if (f,g) is a projection pair in an O-category $\underline{K}$ then we say that g is the (right) adjoint of f and f is the (left) adjoint of g.

Definition 1.4.10

Given an O-category $\underline{K}$, the category of projection pairs of $\underline{K}$, in symbols $\underline{K}^P$ is defined by:

$Ob(\underline{K}^P) = Ob(\underline{K})$

$Hom_{\underline{K}^P}(A,B)$=the set of all projection pairs from A to B,

$id_A^{\underline{K}^P} = (id_A, id_A)$,

$(f',g') \cdot (f,g) = (f' \cdot f, g' \cdot g)$. □

Notice that in the definition of O-categories, the lub of the empty chain is not considered. If we take this into account we have the following notion; An O-category is said to be empty chain complete iff (1) Hom(A,B) has a least element $\perp_{A,B}$ for every $A,B \in \underline{K}$.

(2) $\perp_{B,C} \cdot f = \perp_{A,C}$ for all $f: A \to B$.

Obviously $\perp_{A,B}$ is a lub of the empty chain in Hom(A,B). (2) is concerned with the continuity of composition wrt the empty chain.

Fact 4.1.11 Let $\underline{K}$ be an empty chain complete O-category with a terminal object $\perp$ then $\perp$ is an initial object in $\underline{K}^P$. □

Definition 4.1.12

An O-category $\underline{K}$ has the S-property iff for every ω-codiagram

$\Delta=\langle A_n,(f_n,f_n^R)\rangle$ in $\underline{K}^P$, there is a cocone $\langle\mu,\mu^R\rangle=\langle\mu_n:A_n\to A,\mu_n^R:A\to A_n\rangle$ of Δ s.t. $\mu^R=\langle\mu_n^R:A\to A_n\rangle$ is a limiting cone of the ω-diagram $\Delta^P=\langle A_n,f_n^R\rangle$ in $\underline{K}$. $\langle\mu,\mu^R\rangle$ is called an <u>S-cocone</u> of Δ. □

Note that Plotkin and Smyth [24] emphasized the O-categories which admit limits for every ω-diagram, to gaurantee ω-colimit for the projection pair categories. But this restriction is too strong for ω-BD*, ω-AD*, SFP* are not this type of categories, while they have S-property. As can be seen in the next fact, the notion of S-property reflects the limit-colimit coincidence of Scott [17].

<u>Fact 1.4.13</u>

Let $\underline{K}$ be an O-category with S-property. Every ω-codiagram $\Delta=\langle A_n,(f_n,f_n^R)\rangle$ in $\underline{K}^P$ has a colimiting cocone. Indeed if $\langle\mu,\mu^R\rangle=\langle\mu_n:A_n\to A,\mu_n^R:A\to A_n\rangle$ is a cocone of Δ then the following statements are equivalent:

(1) $\langle\mu,\mu^R\rangle$ is a colimiting cocone of Δ in $\underline{K}^P$

(2) $\mu=\langle\mu_n:A_n\to A\rangle$ is a colimiting cocone of $\Delta^L=\langle A_n,f_n\rangle$ in $\underline{K}$.

(3) $\mu^R=\langle\mu_n^R:A\to A_n\rangle$ is a limiting cone of $\Delta^R=\langle A_n,f_n^R\rangle$ in $\underline{K}$.

(4) $\langle\mu_n\cdot\mu_n^R\rangle$ is an ω-chain in Hom(A,A) s.t. $id_A=\sqcup\mu_n\cdot\mu_n^R$. ▨

<u>Definition 1.4.14</u>

An O-category $\underline{K}$ is said to have <u>locally determined colimits of embeddings</u> iff whenever Δ is an ω-codiagram in $\underline{K}^P$ and $\langle\mu,\mu^R\rangle:\Delta\to A$ is a cocone of Δ in $\underline{K}^P$, then $\langle\mu,\mu^R\rangle$ is a colimiting cocone iff $\langle\mu_n\cdot\mu_n^R\rangle$ is an ω-chain in Hom(A,A) s.t. $id_A=\sqcup\mu_n\cdot\mu_n^R$. □

It immediately follows from 1.4.14 and 1.4.13 that every O-category with S-property has locally determined colimits.

We will consider three O-categories $\underline{K}$, $\underline{L}$, and $\underline{M}$ and a covariant functor $T:\underline{K}^{OP}\times\underline{L}\to\underline{M}$.

<u>Definition 1.4.15</u>

The functor T is <u>locally monotone</u> iff for every $f,f':A\to B$ in $\underline{K}^{OP}$ and $g,g':C\to D$ in $\underline{L}$; $f\sqsubseteq f'$ & $g\sqsubseteq g'$ implies $T(f,g)\sqsubseteq T(f',g')$.

In case T is locally monotone, then we can define a functor $T^P:\underline{K}^P\times\underline{L}^P\to\underline{M}^P$ by:

$$T^P(A,B)=T(A,B),$$

$$T^P((f,f^R),(g,g^R))=(T(f^R,g),T(f,g^R)).$$

Definition 1.4.16

The functor T is locally continuous iff it is continuous on morphism sets, i.e. if $\langle f_n\rangle$ is an ω-chain in $\mathrm{Hom}_{\underline{K}^{OP}}(A,B)$ and $\langle g_n\rangle$ is an ω-chain in $\mathrm{Hom}_{\underline{L}}(C,D)$ then $T(\bigsqcup f_n,\bigsqcup g_n)=\bigsqcup T(f_n,g_n)$ where $T(f_n,g_n)$ is an ω-chain in $\mathrm{Hom}_{\underline{M}}(T(A,C),T(B,D))$. □

Fact 1.4.17

Suppose T is locally continuous and both $\underline{K}$ and $\underline{L}$ have locally determined colimits of embeddings, then $T^P:\underline{K}^P\times\underline{L}^P\to\underline{M}^P$ is an ω-continuous functor. ☒

It immediately follows from 1.4.17 and the remark right after 1.4.14 that if $\underline{K}$ and $\underline{L}$ have S-property then T^P is an ω-continuous functor.

Definition 1.4.18

An empty chain complete O-category which has S-property and a final object is called a Dom-category. □

Notice that in the definition of empty chain completeness the right half of the continuity of composition wrt the lub of the empty chain was omitted, namely,

$$f\cdot\bot_{A,B}=\bot_{A,C}\quad\text{for all } f:B\to C.$$

This condition has the effect of restricting morphisms to strict maps. Given a Dom-category $\underline{K}$ let $\underline{K}^*$ be the O-category obtained from $\underline{K}$ by restricting morphisms to those satisfying the above condition.

Lemma 1.4.19

Let $\underline{K}$ be a Dom-category. Then the terminal object $\bot$ in $\underline{K}$ is the initial object in $\underline{K}^*$.

proof $\perp_{\perp,A}:\perp\to A$. Let $f:\perp\to A$ in $\underline{K}^*$. Then $f\cdot\perp_{\perp,\perp}=\perp_{\perp,A}$. But since $\perp$ is terminal in $\underline{K}$ and hence in $\underline{K}^*$, $\perp_{\perp,\perp}=id_{\perp}$. Thus $f=\perp_{\perp,A}$. This implies that $\perp$ is initial in $\underline{K}^*$. ☐

Lemma 1.4.20

If $\underline{K}=\underline{L}^*$ for some Dom-category $\underline{L}$ then $\underline{K}=\underline{K}^*$. ☐

Dom-categories are the categories which possess a lot of interesting properties of the concrete categories of domains like CPO, ω-BD, ω-AD, and SFP. In fact all of these are Dom-categories and →,+, × are locally continuous functors. For example $\to:CPO^{OP}\times CPO\to CPO$ is defined by:

$$\to(A,B)=[A\to B],$$

$$\to(f:A'\to A,g:B\to B')=\lambda h\in[A\to B].g\cdot h\cdot f.$$

Also it should be noted that one of the simplest example of O-category is the category of sets and partial functions with the set inclusion as the ordering on hom sets.

Notice that if $\underline{K}$ is a Dom-category then $\underline{K}^P=(\underline{K}^*)^P$ is an ω-category. In summary, we have observed how an O-category yields an ω-category and a locally continuous functor yields an ω-continuous functor. The following fact states this:

Facts 1.4.21

(1) If $\underline{K}$ is a Dom-category then $\underline{K}^P=(\underline{K}^*)^P$ is an ω-category.

(2) If $\underline{K},\underline{L}$ and $\underline{M}$ are Dom-categories then $T:\underline{K}^{OP}\times\underline{L}\to\underline{M}$ is ω-continuous whenever T is locally continuous. ☐

CHAPTER 2: EFFECTIVELY GIVEN DOMAINS

> *"In order to make a definition precise, sharp boundaries must be imposed on something. This forces us to become aware of those areas in which our intuition itself is uncertain. This is why finding appropriate definitions is so often the major effort involved in creative scientific work. If a new definition helps classify objects whose status was formerly uncertain, then some new notion must be involved. While on the surface a definition is just a convention, intellectually its acceptance may have a much more active role"*
>
> *Marvin L. Minsky, 1967*
> *in Computation:Finite and Infinite Machines.*

In this chapter, we will present further developments to the theory of effectively given domains. This involves the observation that the notion of computability in an effectively given domain is dependant on the indexing of its basis, as discovered by Park. This indicates that we cannot identify two effectively given domains just because they are order isomorphic. We propose a suitable notion of effective embedding and effective isomorphism to compensate for this deficiency. A less detailed version of this chapter appeared in Kanda and Park [5].

2.1 Effectively Given Domains

The fundamental idea of effectively given domains is to assume effectiveness of finite join operations on a basis of each ω-algebraic domain and to define computable elements as the lub's of r.e. chains of basis elements. For details of the results, obtained so far, based on this idea, see Scott [16,20], Tang [26], Egli and Constable [1], Markowsky and Rosen [15], and Smyth [22].

In this theory it is tempting to avoid questions of indexing. In fact, it has not been clear whether an effectively given domain is to be a domain which can be effectively given in some unspecified way or is to be a domain where this is specified. One could ask if it makes any difference. We will show it does in this chapter. This calls for a rather "tedious" definition of effectively given domains.

Throughout, we assume a fixed acceptable indexing $<\phi_i>$ and $<W_j>$ of partial recursive functions and r.e.sets s.t. W_i=range(ϕ_i).

Definition 2.1.1

(1) Let D be an ω-algebraic domain. A total indexing $\varepsilon:N\to E_D$ is effective iff there is a pair of recursive predicates (b,l) called the characteristic pair of ε, s.t.:

$$b(x) \quad \text{iff} \quad \varepsilon(f_s(x)) \text{ is bounded in } E_D, \text{ and}$$

$$l(k,x) \quad \text{iff} \quad \varepsilon(k)=\sqcup\varepsilon(f_s(x))$$

where f_s is the standard enumeration of finite subsets of N.

(2) An indexed domain is an ordered pair (D,ε) where D is an ω-algebraic domain and $\varepsilon:N\to E_D$ is a total indexing. An indexed domain (D,ε) is an effectively given domain iff ε is effective. We will write D^ε for (D,ε). In case D^ε is an effectively given domain, the characteristic pair of D^ε is that of ε. (E_D,ε) is called the effective basis of the effectively given domain D^ε.

(3) Given an effectively given domain D^ε, $x\in D$ is computable w.r.t. ε (or computable in D^ε) iff for some r.e. set W, $\varepsilon(W)$ is directed and $x=\sqcup\varepsilon(W)$. We say that an r.e. set W is ε-directed iff $\varepsilon(W)$ is directed. The set of all computable elements of D^ε with the induced partial ordering is denoted by Comp(D^ε).

(4) Given effectively given domains D^ε and $D'^{\varepsilon'}$, a function $f:D\to D'$ is computable w.r.t. $(\varepsilon,\varepsilon')$ iff the graph of f which is $\Gamma(f)=\{<n,m>|\varepsilon'(m)\sqsubseteq f\cdot\varepsilon(n)\}$ is r.e. □

It is obvious that there are only counyably many effecively given ω-algebraic domains but there are continumly many ω-algebraic domains. This means most of the ω-algebraic domains are impossible to have effective bases at all. Paterson and Plotkin independently obtained such examples.

In case D^{ε} and $D'^{\varepsilon'}$ have the same characteristic pair, D^{ε} is merely a "renaming" of $D'^{\varepsilon'}$. More formally, there is an order isomorphism $f:D\to D'$ s.t. $f\cdot\varepsilon=\varepsilon'$. We denote this relation by $D^{\varepsilon}\overset{r}{\cong}D'^{\varepsilon'}$. To within $\overset{r}{\cong}$, we can introduce the following partial indexing $\overline{\xi}$ called the <u>acceptable indexing</u> of the class of effectively given domains s.t. $\overline{\xi}(\langle i,j\rangle)$ is the effectively given domain whose characteristic pair is (ϕ_i,ϕ_j). Note that if τ is a partial function then we write $\tau(x)$ iff τ is defined on x.

Notice that there is a well-known recursive isomorphism between N and N×N. One way is the pairing function $\langle n,m\rangle=\frac{1}{2}(n+m)(n+m+1)+m$ and the inverse is the standard enumeration $P_r(n)=\langle\pi_1(n),\pi_2(n)\rangle$ where π_1 and π_2 are the associated projections. We also review how acceptable indexing provides finite representations for the partial recursive functions. For details of this argument readers are refered to Rogers [14].

We assume a suitable symbolism (i.e. syntax of programs) s.t. there is a constructive evaluation process Eval s.t. given a k-ary program R and a tuple $(n_1,..,n_k)$ of natural numbers, Eval may yield a natural number within finite steps. Thence defines a partial recursive function (via Eval). Let $\langle R_i\rangle$ be a constructive enumeration of all programs and ϕ_i the partial recursive function defined by R_i. Then:

(1) i represents ϕ_i in such a sense that R_i can constructively be transformed into $R_{dv1(i)}$ which constructively enumerates the

graph of ϕ_i. Indeed the so called "dove tailing" is the construction needed to do this transformation.

(2) i represents W_i=range(ϕ_i) in such a sense that we can uniformly transform R_i into $R_{dv2(i)}$ which constructively enumerates W_i. Essentially the same dove tailing technique is needed to do this transformation.

In the following we use the usual convention to identify ϕ_i to R_i. Since our indexing ϕ_i is constructive (acceptable) we can regard *dv1* and *dv2* as recursive functions.

Now $\langle i,j \rangle$ represents $\langle \phi_i, \phi_j \rangle$ which characterizes the effectively given domain $\overline{\xi}(\langle i,j \rangle)$. In this sense $\langle i,j \rangle$ is a finite representation of $\overline{\xi}(\langle i,j \rangle)$.

Given an effectively given domain D^{ε} and a computable element x in D^{ε} s.t. W is ε-directed and $x=\sqcup\varepsilon(W)$, it is quite natural to regard a program of a recursive function ρ s.t. W=range(ρ) as a representation of x. We will pursue this idea and will introduce a total indexing to the set of all computable elements of D^{ε}.

<u>Lemma 2.1.2</u>

For every effectively given domain D^{ε}, there is a recursive function $d_{\varepsilon}:N\to N$ s.t. for every $j \in N$, $W_{d_{\varepsilon}(j)}$ is directed and in case W_j is already ε-directed $\sqcup\varepsilon(W_j)=\sqcup\varepsilon(W_{d_{\varepsilon}(j)})$.

<u>proof</u> Remember that $\phi_{dv2(j)}$ recursively enumerates W_j. Think of a recursive function $\rho:N\to N$ defined by:

$$\rho(0)=\phi_{dv2(j)}(0)$$

$$\rho(n+1)=\underline{\text{if}}\ \{\varepsilon(\rho(0)),\ldots,\varepsilon(\rho(n)),\varepsilon(\phi_{dv2(j)}(n+1))\}\text{ is not bounded in }E_D$$

$$\underline{\text{then}}\ \rho(n)$$

$$\underline{\text{else}}\ \mu k.\varepsilon(k)=\sqcup\{\varepsilon(\rho(0)),\ldots,\varepsilon(\phi_{dv2(j)}(n+1))\}.$$

Evidently ρ enumerates a ε-directed set W'. Since this construction is uniform in *dv2*(j), there is a recursive function $d_{\varepsilon}:N\to N$ s.t.

$\rho=\phi_{d_\varepsilon(j)}$. In case W_j is ε-directed evidently $\sqcup\varepsilon(W_j)=\sqcup\varepsilon(W_{d_\varepsilon(j)})$.

∅

We will call d_ε a <u>ε-directing function</u> (or simply a <u>directing function</u>). The above lemma gives us the following total indexing ζ_ε of Comp(D^ε). If $x=\sqcup\varepsilon(W_{d_\varepsilon(j)})$ we say that x has a <u>directed index</u> j and denote it by $\zeta_\varepsilon(j)=x$. If $x=\zeta_\varepsilon(j)$ then $P_{d_\varepsilon(j)}$ is a program which recursively enumerates $W_{d_\varepsilon(j)}$, and $x=\sqcup(W_{d_\varepsilon(j)})$. Therefore j can naturally be regarded as a finite representation of x.

Egli and Constable introduced an alternative to the directed indexing. They tried to transform every r.e. set W to another W' which admits lub's in the sense that $\sqcup\varepsilon(W')$ exists, by making every finite subset of $\varepsilon(W')$ have an upper bound (not necessarily in $\varepsilon(W')$). But the existence of $\sqcup\varepsilon(W')$ can be guaranteed only if D is bounded complete. On the other hand our directed indexing method is based on the directed completeness of D , hence should work out even for the SFP case.

Since we took the view that an effectively given domain is a domain with a specified effective basis, domain constructors must relate not only po-structure but also effective structure. Thus we have to be explicit about constructed bases.

<u>Definition 2.1.3</u>

Given indexed domains D^ε and $D'^{\varepsilon'}$, define $D^\varepsilon \times D'^{\varepsilon'}$, $D^\varepsilon + D'^{\varepsilon'}$, and $D^\varepsilon \times D'^{\varepsilon'}$ to be the following indexed domains:

(1) $D^\varepsilon \times D'^{\varepsilon'}=(D\times D', \varepsilon\times\varepsilon')$ where $\varepsilon\times\varepsilon'(n)=\langle\varepsilon\cdot\pi_1(n),\varepsilon\cdot\pi_2(n)\rangle$.

(2) $D^\varepsilon + D'^{\varepsilon'}=(D+D', \varepsilon+\varepsilon')$

where $\varepsilon+\varepsilon'(n)=$ <u>if</u> n=0 <u>then</u> $\perp$ <u>else</u>

<u>if</u> n=2m+1 <u>then</u> $\langle 0,\varepsilon(m)\rangle$ <u>else</u>

<u>if</u> n=2m <u>then</u> $\langle 1,\varepsilon'(m)\rangle$.

(3) $[D^{\varepsilon}\to D'^{\varepsilon'}]=([D\to D'],[\varepsilon\to\varepsilon'])$

where $[\varepsilon\to\varepsilon'](n)=$ <u>if</u> $\sigma(n)$ has a lub <u>then</u> $\sqcup\sigma(n)$ <u>else</u> $\perp$,

and $\sigma(n)=\{[\varepsilon(i),\varepsilon'(j)]\mid\langle i,j\rangle\in P_r(n)\}$,

where P_r is the standard enumeration of finite subsets of $N\times N$. □

It is well known that if D^{ε} and $D'^{\varepsilon'}$ are effectively given domains then so are $D^{\varepsilon}\times D'^{\varepsilon'}$, $D^{\varepsilon}+D'^{\varepsilon'}$, and $[D^{\varepsilon}\to D'^{\varepsilon'}]$. Indeed the proof of this closure property involves uniform construction of the characteristic pairs of $\varepsilon\times\varepsilon'$, $\varepsilon+\varepsilon'$, and $[\varepsilon\to\varepsilon']$ from those of ε and ε'. Therefore we have:

<u>Theorem 2.1.4</u>

There are recursive functions *Prod*, *Sum*, and *Func* s.t.:

(1) $\overline{\xi}(i)\times\overline{\xi}(j)=\overline{\xi}(Prod(i,j))$.

(2) $\overline{\xi}(i)+\overline{\xi}(j)=\overline{\xi}(Sum(i,j))$.

(3) $[\overline{\xi}(i)\to\overline{\xi}(j)]=\overline{\xi}(Func(i,j))$.

<u>proof</u> (1) Let (b,l) and (b',l') be characteristic pairs of $\overline{\xi}(i)$ and $\overline{\xi}(j)$ respectively. The characteristic pair $(b\times b',l\times l')$ of $\overline{\xi}(i)\times\overline{\xi}(j)$ can be defined by:

$$b\times b'(x)=b(\pi_1(x))\&b'(\pi_2(x))$$
$$=b(\pi_1(x))\times b'(\pi_2(x)),$$
$$l\times l'(k,x)=l(\pi_1(k),\pi_1(x))\&l'(\pi_2(k),\pi_2(x))$$
$$=l(\pi_1(k),\pi_1(x))\times l'(\pi_2(k),\pi_2(x))$$

where the right side $\times$ is the arthmetic multiplication and & is the logical and. (2), (3) can be proved similarly. ☑

Notice that essentially the above proof involves a program transformation which yields a program for $b\times b'$ from the programs for b and b'. In case input to this transformation is not a pair of pairs of programs for a pair of characteristic pairs, some program system is still formally obtained.

Smyth [22] showed that a function $f:D\to D'$ is computable w.r.t. $(\varepsilon,\varepsilon')$ iff $f\in Comp([D^{\varepsilon}\to D'^{\varepsilon'}])$. This proof involves a construction of $[\varepsilon\to\varepsilon']$-directed r.e. set W s.t. $f=\bigsqcup[\varepsilon\to\varepsilon'](W)$ from the graph of f and vice versa. In order to observe that this uniformity induces recursive functions between directed indexing and graph indexing, which is a partial indexing of the class of computable functions by means of the acceptable indices of the graphs of them, we will examine this proof in detail.

Fact 2.1.5

A function $f:D\to D'$ is computable w.r.t $(\varepsilon,\varepsilon')$ iff f is a computable element of $[D^{\varepsilon}\to D'^{\varepsilon'}]$.

proof Let $\Gamma(f)=\{\langle m,n\rangle \mid \varepsilon'(n)\sqsubseteq f\cdot\varepsilon(m)\}$ be r.e. Notice that there are recursive functions *Step* and *Epr* s.t. $P_r(Step(\langle m,n\rangle)=\{\langle m,n\rangle\}$ and $f_s(Epr(m,n))=\{m,n\}$. Then we have: $[\varepsilon\to\varepsilon'](Step(\langle m,n\rangle))=[\varepsilon(m),\varepsilon'(n)]$. Remember that $\varepsilon'(n)\sqsubseteq f\cdot\varepsilon(m)$ iff $[\varepsilon\to\varepsilon'](Step(\langle m,n\rangle))\sqsubseteq f$. Therefore $[\varepsilon\to\varepsilon'](Step(\Gamma(f)))\subseteq E_{[D\to D']}$ and $f=\bigsqcup[\varepsilon\to\varepsilon'](Step(\Gamma(f)))$. Since $\Gamma(f)$ is r.e. $Step(\Gamma(f))$ has a recursive enumeration $r_0,r_1,\ldots$. Define a chain $\langle y_n\rangle$ in $E_{[D\to D']}$ by:

$$y_0=[\varepsilon\to\varepsilon'](r_0)$$

$$y_{n+1}=y_n\sqcup[\varepsilon\to\varepsilon'](r_{n+1}).$$

This is well-defined because every finite subset of $[\varepsilon\to\varepsilon'](Step(\Gamma(f)))$ has a lub in $E_{[D\to D']}$ for $[D\to D']$ is bounded complete. Furthermore there is a recursive function ρ s.t. $y_n=[\varepsilon\to\varepsilon'](\rho(n))$. Indeed $\rho(0)=r_0$ and $\rho(n+1)=\mu k.[l\to l'](k,h(n))$ where h is a recursive function satisfying $P_r(h(n))=\{\langle\rho(n),r_{n+1}\rangle\}$ .Therefore $\langle y_n\rangle$ is an $[\varepsilon\to\varepsilon']$-effective chain. Thus $f=\bigsqcup y_n$ is computable in $[D\to D']$. Conversely let W be $[\varepsilon\to\varepsilon']$-directed s.t. $f=\bigsqcup[\varepsilon\to\varepsilon'](W)$. Evidently $[\varepsilon(m),\varepsilon'(n)]\sqsubseteq f$ iff $[l\to l'](x,Epr(x,Step(\langle m,n\rangle)))=0$ for some $x\in W$

Thus $\{<m,n> \mid [\varepsilon(m),\varepsilon'(n)] \subseteq f\}$ is r.e. Therefore f is computable w.r.t. $(\varepsilon,\varepsilon')$. Note that $[l\to l']$ is the second coordinate of the characteristic pair of $[\varepsilon\to\varepsilon']$. ☒

Theorem 2.1.6

There are recursive functions Dg, Gd:N→N s.t.:

(1) If k is a graph index of f which is computable w.r.t. $(\xi(i),\xi(j))$ then $f=\zeta_{[\xi(i)\to\xi(j)]}(Dg(k,i,j))$.

(2) If $f=\zeta_{[\xi(i)\to\xi(j)]}(k)$ then f has a graph index $Gd(k,i,j)$, where $\xi(i)$ denotes the effective basis of $\overline{\xi}(i)$.

proof (1) Given k∈N, $r=Step\cdot\phi_{dv2(k)}$ recursively enumerate $Step(W_k)$. Given i,j∈N, define ρ by:

$$\rho(0)=r(0),$$

$$\rho(n+1)=\mu k.\phi_{\pi_2\cdot Func(i,j)}(k,h(n))$$

where h is as before, i.e. as in the proof of 2.1.5. Then ρ is a partial recursive function. Since the construction of ρ is uniform in i,j,k, there is a recursive function $\tau(k,i,j)$ s.t. $\phi_{\tau(k,i,j)}=\rho$. This τ is the desired Dg. Notice that if k is a graph index of a computable function then the above ρ is recursive and behaves as the ρ in the previous proof.

(2) Similarly. ☒

Notice that in the above proof again a program transformation was involved. In fact we transformed any given program P_k into $P_{\tau(k,i,j)}$. Indeed we can claim that for most cases of constructing an r.e. set from some other r.e. set in a certain class of r.e. sets, unless we do somehow very sophisticated things, involved is a simple always terminating program transformation. So in the rest of this dissertation, whenever we come across this sort of situation, we will indicate constructive transformation of r.e. sets into r.e. sets and omit details of proofs of results on indices like 2.1.4 and 2.1.6.

Given effectively given domains $D^{\varepsilon}, D'^{\varepsilon'}$, and $D''^{\varepsilon''}$, let $f:D\to D'$ and $g:D'\to D''$ be computable w.r.t. $(\varepsilon,\varepsilon')$ and $(\varepsilon',\varepsilon'')$ respectively. By uniformly constructing the graph of $g\cdot f$, we can show that $g\cdot f$ again is computable w.r.t. $(\varepsilon',\varepsilon'')$. The uniformity of the construction and (1)-2.1.6 establish:

Theorem 2.1.7

There exists a recursive function *Compose* s.t. :

$$\zeta_{[\xi(k)\to\xi(m)]}(i)\cdot\zeta_{[\xi(m)\to\xi(n)]}(j) = \zeta_{[\xi(k)\to\xi(n)]}(Compose(i,j,k,m,n))$$

☐

Theorem 2.1.8

A continuous function from an effectively given domain to another is computable w.r.t. their effective bases iff it maps computable elements to computable elements recursively in directed indices.

proof (only if part): Notice that if $f:D\to D'$ is computable w.r.t. $(\varepsilon,\varepsilon')$, then we can uniformly construct an r.e. set $W_{\Psi_f(k)} = \{n \mid \varepsilon'(n) \sqsubseteq f\cdot\varepsilon(m)$ for some $m\in W_k\}$ for every $k\in N$. Thus Ψ_f is a recursive function. It is evident that if W_k is ε-directed then $W_{\Psi_f(k)}$ is ε'-directed.

(if part): Assume that Ψ_f is a recursive function s.t. $\zeta_{\varepsilon'}(\Psi_f(k)) = f(\zeta_\varepsilon(k))$. Notice that there exists a recursive function *Bd* s.t. $\varepsilon(m)=\zeta_\varepsilon(Bd(m))$. Now $f\cdot\varepsilon(m)=\zeta_{\varepsilon'}\cdot\Psi_f\cdot Bd(m)$. Therefore $\varepsilon'(n)\sqsubseteq f\cdot\varepsilon(m)$ is r.e. Therefore f is computable w.r.t. $(\varepsilon,\varepsilon')$. ☐

The proof of 2.1.8 has a further implication.

Theorem 2.1.9

There exists a recursive function *Apply* s.t. if k is a graph index of $f\in Comp([\overline{\xi}(i)\to\overline{\xi}(j)])$ then:

$$\zeta_{\xi(j)}(Apply(i,j,k,m)=f(\zeta_{\xi(i)}(m)).$$ ☐

It is evident that we have a recursive function *Apply'* which takes directed indices of functions rather than graph indices, for we have 2.1.6.

Let COMPOSE: $[\overline{\xi}(k)\to\overline{\xi}(m)]\times[\overline{\xi}(m)\to\overline{\xi}(n)]\to[\overline{\xi}(k)\to\overline{\xi}(n)]$ be defined by COMPOSE$(f,g)=g\cdot f$. Then it can readily be seen that COMPOSE is computable. Thence we can obtain 2.1.7 as a corollary to 2.1.9.

Before ending this section, we will observe one important effectiveness result which will play an essential role in the categorical argument in the last chapter.

Definition 2.1.10

Given an effectively given domain D^{ε}, an effectively directed (ef-directed) subset of D^{ε} is a directed subset $Z\subseteq \mathrm{Comp}(D^{\varepsilon})$ s.t. $Z=\zeta_{\varepsilon}(W)$ for some r.e. set W. We say this W is ζ_{ε}-directed. If W_j is ζ_{ε}-directed then we say that $\zeta_{\varepsilon}(W)$ has a ζ_{ε}-directed index j. ☐

Theorem 2.1.11

(1) Let $\zeta_{\varepsilon}(W)$ be an ef-directed subset of D^{ε}. Then $\bigsqcup\zeta_{\varepsilon}(W)$ is a computable element of D^{ε}.

(2) There is a recursive function *Lub* s.t. if $\zeta_{\varepsilon}(W_j)$ has a ζ_{ε}-directed index j then $\zeta_{\varepsilon}(Lub(j))=\bigsqcup\zeta_{\varepsilon}(W_j)$. Intuitively speaking taking ef-directed limit is an effective operation.

proof Given an r.e. set W_j, we can construct an r.e. set W' s.t. $Y=\bigcup_{x}\varepsilon(W^{x})=\varepsilon(W')$, with $x\in\zeta_{\varepsilon}(W_j)$ and $\varepsilon(W^{x})=\{e\in\varepsilon\mid e\sqsubseteq x\}$. If W_j is ζ_{ε}-directed then W' is ζ_{ε}-directed and $\bigsqcup Y=\bigsqcup\zeta_{\varepsilon}(W_j)$. Thus we have proved both (1) and (2). ☐

Intuitively, (2)-2.1.11 means that given effective enumeration of programs each of which effectively approximates an element of $\zeta_{\varepsilon}(W)$, we can construct a program which approximates $\bigsqcup\zeta_{\varepsilon}(W)$.

2.2 Effective Embeddings

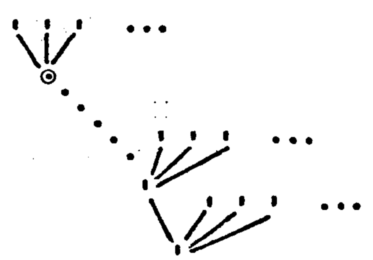
In this section, we will observe that po-structure of a domain can't uniquely determine effective structure, even if it can be effectively given.

Theorem 2.2.1 (Park)

(1) There is a countably algebraic domain D with two different effective bases ε and ε' s.t. $\mathrm{Comp}(D^{\varepsilon})=\mathrm{Comp}(D^{\varepsilon'})$ but s.t. $\mathrm{Comp}([D^{\varepsilon}\to O^{\pi}]) \supsetneqq \mathrm{Comp}([D^{\varepsilon'}\to O^{\pi}])$ where O is the two point lattice and π is an arbitary effective basis of O.

(2) There is a countably algebraic domain D with two different effective bases ε and ε' s.t. $\mathrm{Comp}(D^{\varepsilon}) \not\supseteq \mathrm{Comp}(D^{\varepsilon'})$.

proof (1) Let $(D,\sqsubseteq)$ be the following countably algebraic domain:



Note that D has only one limit point ⊙. Thus the basis of D is the poset obtained from D by removing the limit point. Think of the following poset $(N\cup(N\times N),\sqsubseteq)$ where $i\sqsubseteq j$ iff $i\le j$, $i\sqsubseteq\langle m,n\rangle$ iff $\phi_m(n)$ takes at least i steps, and $\langle m,n\rangle\sqsubseteq\langle m',n'\rangle$ iff $m=m'$ and $n=n'$. Then the partial ordering $\sqsubseteq$ is decidable in terms of the Gödel numbering of $N\cup(N\times N)$. Thus this Gödel numbering provides an effective indexing ε' of E_D. Now think of the following poset $(N\cup(N\times N)\cup(\{\omega\}\ N),\sqsubseteq)$ s.t. $i\sqsubseteq j$ iff $i\le j$, $i\sqsubseteq\langle m,n\rangle$ iff $i\le m$ and $\langle m,n\rangle\sqsubseteq\langle m',n'\rangle$ iff $m=m'$ and $n=n'$. Evidently the Gödel numbering of $N\cup(N\times N)\cup(\{\omega\}\times N)$ provides an effective indexing ε of E_D. Obviously $\mathrm{Comp}(D^{\varepsilon})=\mathrm{Comp}(D^{\varepsilon'})=D$. Now let $f:D\to O$ be a continuous map s.t. $f(x)=$ if $x\sqsupseteq$⊙ then $\top$ else $\bot$. Then f is computable w.r.t.

$\langle\varepsilon,\pi\rangle$ but not so w.r.t. $\langle\varepsilon',\pi\rangle$. Now let $M=\{h\in[D\to O]\mid\{g\in[D\to O]\mid g\sqsubseteq h\}$ is finite$\}$. Then $M=\{h_X\mid X$ is a finite set of leaves above compact elements of D$\}$ where $h_X=$ <u>if</u> $x\sqsubseteq y\in X$ <u>then</u> $\perp$ <u>else</u> $\top$. It can readily be seen that $M\subseteq Comp([D^{\varepsilon}\to O^{\pi}])$ and $M\subseteq Comp([D^{\varepsilon'}\to O^{\pi}])$. Let $\Xi:Comp([D^{\varepsilon}\to O^{\pi}])\to Comp([D^{\varepsilon'}\to O^{\pi}])$ be an isomorphism. Then $\Xi(M)=M$. Notice that $f=\sqcap M$. Therefore $\sqcap M\in Comp([D^{\varepsilon}\to O^{\pi}])$. Since Ξ is an isomorphism, $\Xi(\sqcap M)=\sqcap' M\in Comp([D^{\varepsilon'}\to O^{\pi}])$. But M should have no greatest lower bound in $Comp([D^{\varepsilon'}\to O^{\pi}])$. ☐

Notice that 2.2.1 is more than a counter example to a careless definition of effectively given domains. In fact (1)-2.2.1 indicates that $Comp(D^{\varepsilon})=Comp(D^{\varepsilon'})$ is not sufficient to identify D^{ε} and $D^{\varepsilon'}$. Remember that in domain theory, domain constructors must preserve equivalence of domains, more technically, they must be functors. But if we assume that D^{ε} and $D^{\varepsilon'}$ are equivalent iff $Comp(D^{\varepsilon})=Comp(D^{\varepsilon'})$ then "$\to$" does not preserve this equivalence as shown in (1)-2.2.1. We claim that the following equivalence of effectively given domains is appropriate.

<u>Definition 2.2.2</u>

Let D^{ε} and $D^{\varepsilon'}$ be indexed domains. We say that ε and ε' a are <u>effectively equivalent</u> (in symbol $\varepsilon\overset{e}{=}\varepsilon'$) iff there are recursive functions $r,s:N\to N$ s.t. $\varepsilon'=\varepsilon\cdot s$ and $\varepsilon=\varepsilon'\cdot r$. ☐

It can readily be seen that if either ε or ε' is effective then $\varepsilon\overset{e}{=}\varepsilon'$ implies both ε and ε' are effective and $Comp(D^{\varepsilon})=Comp(D^{\varepsilon'})$.

Notice that D^{ε} and $D^{\varepsilon'}$ of the proof of 2.2.1 are not effectively equivalent. In fact if ε and ε' were effectively equivalent then there could exist a recursive function $c:N\to N$ s.t. $\phi_m(n)$ terminates iff $c(\langle m,n\rangle)=\langle m',n'\rangle$ with $m'\neq\omega$ and we could solve the halting problem.

We can easily extend the notion of effective equivalence to isomorphisms.

Definition 2.2.3

(1) Let D^{ε} and $D'^{\varepsilon'}$ be indexed domains. A function $f:E_D \to E_{D'}$ is an effective imbedding from ε to ε' (in symbols $f:\varepsilon \to \varepsilon'$) iff

1. f is an imbedding from E_D to $E_{D'}$.

2. there is a recursive function r_f s.t. $f \cdot \varepsilon = \varepsilon' \cdot r_f$.

Remember "imbedding" was defined in 1.2.19.

(2) We say that ε and ε' are effectively isomorphic (in symbols $\varepsilon \overset{e}{\cong} \varepsilon'$ or $D^{\varepsilon} \overset{e}{\cong} D'^{\varepsilon'}$) iff there exists an effective imbedding $f:\varepsilon \to \varepsilon'$ s.t. $f^{-1}:E_{D'} \to E_D$ is also an effective imbedding from ε' to ε.

□

Lemma 2.2.4

(1) Let D^{ε} and $D'^{\varepsilon'}$ be indexed domains and f be an effective imbedding from ε to ε'. Then the unique continuous extension $\bar{f}:D \to D'$ of f is a continuous embedding with the adjoint $\bar{g}:D' \to D$ given by $\bar{g}(y) = \bigsqcup\{e \in E_D \mid f(e) \sqsubseteq y\}$. Furthermore $\bar{g} \upharpoonright \bar{f}(E_D) = f^{-1}$.

(2) In case D^{ε} and $D'^{\varepsilon'}$ are effectively given domains, $\bar{f}$ is computable w.r.t. $(\varepsilon, \varepsilon')$ and $\bar{g}$ is computable w.r.t. $(\varepsilon', \varepsilon)$. Furthermore $\bar{f}(\mathrm{Comp}(D^{\varepsilon})) \subseteq \mathrm{Comp}(D'^{\varepsilon'})$, in words, $\bar{f}$ embedds computable elements.

(3) If $D^{\varepsilon} \overset{e}{\cong} D'^{\varepsilon'}$ and either of them is effectively given, then both of them are effectively given and $\mathrm{Comp}(D^{\varepsilon}) \cong \mathrm{Comp}(D'^{\varepsilon'})$.

proof (1) $f = \bar{f} \upharpoonright E_D$ is obviously an imbedding. Thus by 1.2.20.

(2) $\varepsilon'(m) \sqsubseteq \bar{f} \cdot \varepsilon(n)$ iff $\varepsilon'(m) \sqsubseteq \varepsilon' \cdot r_f(n)$. Therefore $\bar{f}$ is computable w.r.t. $(\varepsilon, \varepsilon')$. Since $g \cdot \varepsilon'(n) = \bigsqcup\{\varepsilon(j) \mid f \cdot \varepsilon(j) \sqsubseteq \varepsilon'(n)\}$ $= \bigsqcup\{\varepsilon(j) \mid \varepsilon'(r_f(j)) \sqsubseteq \varepsilon'(n)\}$ and $\varepsilon(m)$ is compact, we have:

$\varepsilon(m) \sqsubseteq g \cdot \varepsilon'(n)$ iff $\varepsilon(m) \sqsubseteq \varepsilon(j)$ for some j s.t. $\varepsilon' \cdot r_f(j) \sqsubseteq \varepsilon'(n)$.

Thus $\varepsilon(m) \sqsubseteq g \cdot \varepsilon'(n)$ is r.e. in m and n. Therefore g is computable w.r.t. $(\varepsilon',\varepsilon)$. By the computability of $\overline{f}$ and 2.1.9, we obviously have $\overline{f}(\mathrm{Comp}(D^{\varepsilon}))\subseteq \mathrm{Comp}(D'^{\varepsilon'})$.

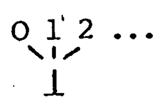
(3) Evident. ☐

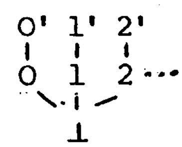
<u>Definition 2.2.5</u>

(1) If $f:E_D \to E_{D'}$ is an effective imbedding from ε to ε', then we say that $\overline{f}:D\to D'$ is an <u>effective embedding</u> from D^{ε} to $D'^{\varepsilon'}$.

(2) Let D^{ε} and $D'^{\varepsilon'}$ be effectively given domains. A continuous embedding $f:D\to D'$ is a <u>computable embedding</u> from D^{ε} to $D'^{\varepsilon'}$ iff f is computable w.r.t. $(\varepsilon,\varepsilon')$ and the adjoint $g:D'\to D$ is computable w.r.t. $(\varepsilon',\varepsilon)$. g is called a <u>computable projection</u> and (f,g) is called a <u>computable projection pair</u>. ☐

Let W be an non-recursive r.e. set and (N,ε), (N',ε') be the following effectively given domains:

N:  $\varepsilon(0)=\bot$, $\varepsilon(n+1)=n$,

N':  $\varepsilon'(0)=\bot$, $\varepsilon'(2(n+1))=(n-1)'$, $\varepsilon'(2n+1)=n$.

Then $f:N\to N'$ defined by $f(x)=$ <u>if</u> $x=\bot$ <u>then</u> $\bot$ <u>else</u> <u>if</u> $x\notin W$ <u>then</u> x <u>else</u> x' is an embedding computable wrt $(\varepsilon,\varepsilon')$. But it is not an effective embedding since W is not a recursive set. This example due to Plotkin indicates that <u>not all embeddings which are computable are effective embeddings</u>.

<u>Theorem 2.2.6</u>

Let D^{ε} and $D'^{\varepsilon'}$ be effectively given domains s.t. $f:D\to D'$ is a computable embedding, then f is an effective embedding.

<u>proof</u> Let $g:D'\to D$ be the adjoint of f. Then both $\varepsilon'(n)\sqsubseteq f\cdot\varepsilon(m)$ and $\varepsilon(n)\sqsubseteq g\cdot\varepsilon'(m)$ are r.e. in indices. We will show the existence of a recursive function r s.t. $f\cdot\varepsilon=\varepsilon'\cdot r$. We claim that the following terminating program computes such $r(m)$ for $m\in N$.

- enumerate n s.t. $\varepsilon'(n)\sqsubseteq f\cdot\varepsilon(m)$.
- for each enumerated n, enumerate k s.t. $\varepsilon(k)\sqsubseteq g\cdot\varepsilon'(n)$.
- continue until we obtain a k s.t. $\varepsilon(k)=\varepsilon(m)$.

The n for which this k is produced is the desired $r(m)$. By a "dove-tailing" technique, we can compute the above process. We now check that such r is actually the one desired. Assume k and n are the values when the above process terminates. Then $\varepsilon(k)\sqsubseteq g\cdot\varepsilon'(n)\sqsubseteq g\cdot f(\varepsilon(m))=\varepsilon(m)$. Since $\varepsilon(k)=\varepsilon(m)$, we have: $g\cdot\varepsilon'(n)=\varepsilon(m)$. But $\varepsilon'(n)\sqsupseteq f\cdot g\cdot\varepsilon'(n)=f\cdot\varepsilon(m)$. Therefore $\varepsilon'(n)=f\cdot\varepsilon(m)$. ☒

In summary, we have observed that the effective embeddings of effectively given domains are exactly the computable embeddings. This coincidence indicates the naturalness of the notion of computable embeddings. It immediately follows from these observations that an isomorphism between two effectively given domains is an effective isomorphism iff both itself and its adjoint are computable. Also this coincidence implies that the composition of two effective embeddings is again an effective embedding.

The coincidence of effective embeddings of effectively given domains and computable embeddings is in fact "effective". Given an effective imbedding $f:\varepsilon\to\varepsilon'$, if $r_f=\phi_j$ then we say that f has a <u>recursive</u> <u>index</u> j. Also we say that the effective embedding $\overline{f}$ has a <u>recursive</u> <u>index</u> j. Now we have:

<u>Theorem 2.2.7</u>

(1) There is a recursive function *Rd* s.t. if i and j are directed indices of a computable embedding $f\in Comp([\overline{\xi}(k)\to\overline{\xi}(m)])$ and the adjoint $g\in Comp([\overline{\xi}(m)\to\overline{\xi}(k)])$ respectively then $Rd(i,j,k,m)$ is a recursive index of f.

(2) There are recursive functions *De* and *Dp* s.t. if i is a recursive index of an effective embedding $f\in Comp([\overline{\xi}(j)\to\overline{\xi}(k)])$

then $De(i,j,k)$ is a directed index of f and $Dp(i,j,k)$ is a directed index of the adjoint $g \in \mathrm{Comp}([\overline{\xi}(k) \to \overline{\xi}(j)])$ of f.

proof By the effectiveness of the proof of 2.2.4 and 2.2.6.

□

If i and j are directed indices of a computable embedding f and its adjoint g respectively, then we say that $\langle i,j \rangle$ is a directed index of the computable projection pair (f,g).

Remember that we have claimed that the notion of effective isomorphism gives an appropriate criterion for identifying two effectively given domains. We can provide quite convincing evidences for this claim. First, evidently $\overset{e}{\cong}$ is an equivalence relation. Furthermore we can show that $\overset{e}{\cong}$ is invariant under the domain constructors $\times$, $+$, and $\to$. More formally we have:

Theorem 2.2.8

Let $A^{\alpha}, B^{\beta}, C^{\gamma}$, and D^{δ} be indexed domains s.t. $A^{\alpha} \overset{e}{\cong} C^{\gamma}$ and $B^{\beta} \overset{e}{\cong} D^{\delta}$. Then we have:

(1) $A^{\alpha} \times B^{\beta} \overset{e}{\cong} C^{\gamma} \times D^{\delta}$

(2) $A^{\alpha} + B^{\beta} \overset{e}{\cong} C^{\gamma} + D^{\delta}$

(3) $[A^{\alpha} \to B^{\beta}] \overset{e}{\cong} [C^{\gamma} \to D^{\delta}]$.

proof (1) and (2) are easy.

We will prove (3). For the sake of simplicity we prove this theorem for $\overset{e}{=}$. Assume r, r', s, s' are recursive functions s.t. $\gamma = \alpha \cdot r'$, $\delta = \beta \cdot s'$, $\alpha = \gamma \cdot r$, and $\beta = \delta \cdot s$. Notice that we have assumed A=C and B=D. Then it can readily be seen that there is a recursive function $i: N \to N$ s.t.:

$$\{[\alpha(i), \beta(j)] \mid \langle i,j \rangle \in P_r(n)\} = \{[\gamma \cdot r(i), \delta \cdot s(j)] \mid \langle r(i), s(j) \rangle \in P_r i(n)\}$$

Thus $[\alpha \to \beta](n) = \underline{\text{if}}\ \sqcup\{[\alpha(i), \beta(j)] \mid \langle i,j \rangle \in P_r(n)\}$ exists $\underline{\text{then}}$ this lub $\underline{\text{else}}\ \bot = [\gamma \to \delta](i(n))$.

Similarly we have a recursive j s.t. $[\gamma \to \delta](n) = [\alpha \to \beta](j(n))$. □

2.3 Algebraic Completion

Smyth [22] characterized effectively given continuous domains as the completion of computable R-structures. The indexing problem was not considered. We will characterize effectively given algebraic domains as the algebraic completion of effective posets, taking care of effective isomorphisms.

Definition 2.3.1

Let $(E,\sqsubseteq)$ be a countable poset with bottom and bounded joins and $\varepsilon:N\to E$ be a total indexing. We call (E,ε) an indexed poset. In case ε is effective, i.e. ε satisfies (1)-2.1.1, we call (E,ε) an effective poset. The (algebraic) completion of an indexed poset (E,ε) is an indexed domain $(\bar{E},\bar{\varepsilon})$ where $\bar{E}$ is the algebraic completion of E and $\bar{\varepsilon}:N\to\tau(E)$ is given by $\bar{\varepsilon}(n)=\tau\cdot\varepsilon(n)$ where τ is the canonical map from E to $\bar{E}$.

Theorem 2.3.2

(1) Let (E,ε) be an effective poset. Then the completion of it is an effectively given domain.

(2) Given an effectively given domain D^{ε}, the effective basis E^{ε}_D is an effective poset and $(\bar{E}_D,\bar{\varepsilon})=(D,\varepsilon)$ (to within $\overset{r}{\cong}$).

(3) An indexed domain is an effectively given domain iff it is the completion of some effective poset (to within $\overset{r}{\cong}$). ∎

The above theorem indicates that the effective bases of effectively given domains are exactly effective posets. This point can be made more explicit. It is obvious that we can introduce the renaming relation on the class of effective posets and to within $\overset{r}{\cong}$ associate acceptable indices to each effective poset. We will use ξ to denote the acceptable indexing of effective posets. Now let $\overline{\xi(\langle i,j\rangle)}$ denote the algebraic completion of the effective poset $\xi(<i,j>)$. Then we have:

Theorem 2.3.3 (The Acceptable Indexing Theorem)

$$\overline{\xi(<i,j>)} = \overline{\xi}(<i,j>) \quad (\text{to within } \overset{r}{\cong}). \qquad \square$$

By virtue of the above theorem, we can say that $\xi(<i,j>)$ is the effective basis of $\overline{\xi}(<i,j>)$.

2.4 Inverse Limits

Let $<D_m^{\varepsilon_m}, (f_m, f_m^R)>$ be an ω-sequence of continuous projection pairs of indexed domains. By the inverse limit of this sequence, in symbols $\varprojlim<D_m^{\varepsilon_m}, (f_m, f_m^R)>$ or $\varprojlim<D_m^{\varepsilon_m}>$, we mean an indexed domain $(D_\infty, \varepsilon_\infty)$ where $D_\infty = \varprojlim<D_m, (f_m, f_m^R)>$ and ε_∞ is given by:

$$\varepsilon_\infty(0) = f_{0\infty}(\varepsilon_0(0)) \qquad \varepsilon_\infty(1) = f_{0\infty}(\varepsilon_0(1))$$

$$\varepsilon_\infty(2) = f_{1\infty}(\varepsilon_1(0)) \qquad \varepsilon_\infty(3) = f_{0\infty}(\varepsilon_0(2))$$

More precisely, $\varepsilon_\infty(<n,m>) = f_{n\infty}(\varepsilon_n(m))$.

In case $D_m^{\varepsilon_m}$ are effectively given domains, even if (f_m, f_m^R) are computable projection pairs, $\lim<D_m^{\varepsilon_m}>$ need not be so. This immediately follows from the observation of effectiveness of SFP objects. For establishing closure under limit, we need the notion of effectiveness of the sequences.

Definition 2.4.1

Let $<D_m^{\varepsilon_m}, (f_m, f_m^R)>$ be an ω-sequence of computable projection pairs of effectively given domains. In case there exists a recursive function $q: N \to N$ s.t. $\pi_1 \cdot q(m)$ is a recursive index of f_m and $\pi_2 \cdot q(m)$ is an acceptable index of $D_m^{\varepsilon_m}$, we say that this sequence is effective. $\square$

By virtue of 2.2.7, we immediately have the following alternative characterization of effective sequences:

Lemma 2.4.2

An ω-sequence $\langle D_m^{\varepsilon_m},(f_m,f_m^R)\rangle$ of computable projection pairs of effectively given domains is effective iff there is a recursive function q s.t. $\pi_1\cdot\pi_1\cdot q(m)$ is a directed index of f_m, $\pi_2\cdot\pi_1\cdot q(m)$ is a directed index of f_m^R, and $\pi_2\cdot q(m)$ is an acceptable index of $D_m^{\varepsilon_m}$. ☒

Theorem 2.4.3 (The Inverse Limit Theorem)

Let $\langle D_m^{\varepsilon_m},(\bar{f}_m,\bar{f}_m^R)\rangle$ be an effective sequence of computable projection pairs of effectively given domains. Then the inverse limit $(D_\infty,\varepsilon_\infty)$ is an effectively given domain. Also $\bar{f}_{m\infty}$ is an effective embedding from ε_m to ε_∞, thence $(\bar{f}_{m\infty},\bar{f}_{m\infty}^R)$ is a computable projection pair. Furthermore there exist recursive functions λ_d and δ_d s.t. $\lambda_d(m)$ and $\delta_d(m)$ are directed indices of $\bar{f}_{m\infty}$ and $\bar{f}_{m\infty}^R$ respectively.

proof Let $f_m=\bar{f}_m\restriction E_{D_m}$, $f_m^R=\bar{f}_m^R\restriction E_{D_{m+1}}$, $f_{m\infty}=\bar{f}_{m\infty}\restriction E_{D_m}$, and $f_{m\infty}^R=\bar{f}_{m\infty}^R\restriction E_{D_\infty}$. Assume $\{\varepsilon_\infty(i_1),\ldots,\varepsilon_\infty(i_n)\}\subseteq E_{D_\infty}$. There are recursive functions a and b s.t. $\varepsilon_\infty(m)=\bar{f}_{a(m)\infty}\cdot\varepsilon_{a(m)}(b(m))$. Let $\text{Deg}(\{\varepsilon_\infty(i_1),\ldots,\varepsilon_\infty(i_n)\})=\max\{a(i_1),\ldots,a(i_n)\}$. Then there exists a recursive function Deg s.t. $Deg(i_1,\ldots,i_n)=\text{Deg}(\{\varepsilon_\infty(i_1),\ldots,\varepsilon_\infty(i_n)\})$. Let $f_{ij}=f_j\cdot\ldots\cdot f_i$ $(i\leq j)$. Since f_m are imbeddings, $\{\varepsilon_\infty(i_1),\ldots,\varepsilon_\infty(i_n)\}$ is bounded iff $\{f_{a(i_1)Deg(i_1,\ldots,i_n)}(\varepsilon_{a(i_1)}(b(i_1)),\ldots,$ $f_{a(i_n)Deg(i_1,\ldots,i_n)}(\varepsilon_{a(i_n)}(b(i_n))\}$ is bounded. Since the given sequence is effective, there is a recursive function Mrg s.t. $Mrg(k,i_1,\ldots,i_n)$ is a recursive index of $f_{a(i_k)Deg(i_1,\ldots,i_n)}$ for every $k\in n$. Thus there is a recursive function Red s.t.

$\varepsilon_{Deg(i_1,..,i_n)}(Red(i_1,..,i_n,k))=f_{a(i_k)Deg(i_1,...,i_n)}(\varepsilon_{a(i_k)}(b(i_k)))$, for every $k\le n$. Therefore "$\{\varepsilon_\infty(i_1),..,\varepsilon_\infty(i_n)\}$ is bounded" is recursive in $i_1,..,i_n$ since ε_m are effective bases. Similarly $\varepsilon_\infty(k)=\sqcup\{\varepsilon_\infty(i_1),..,\varepsilon_\infty(i_n)\}$ is recursive in indicies. Therefore ε_∞ is an effective basis of D_∞.
By the definition of ε_∞ we can effectively find n for each k s.t. $\varepsilon_\infty(n)=\overline{f}_{m\infty}\cdot\varepsilon_m(k)$ for some m. Therefore there is a recursive function r s.t. $\varepsilon_\infty(r(k))=\overline{f}_{m\infty}\cdot\varepsilon_m(k)$. Therefore $\overline{f}_{m\infty}$ is an effective embedding.
The existence of recursive functions λ_d and δ_d immediately follows from the definition of ε_∞. ☐

Notice that in tne above proof, we have constructed a characteristic pair of the inverse limit from a program which enumerates the characteristic pairs of the sequence. This point can be made explicit.

Given an effective sequence $<D_m^{\varepsilon_m},(f_m,f_m^R)>$ of computable projection pairs of effectively given domains, we say it has a <u>sequence</u> <u>index</u> j iff ϕ_j is a recursive function s.t. $\pi_1\cdot\pi_1\cdot\phi_j(m),\pi_2\cdot\pi_1\cdot\phi_j(m)$, are directed indices of f_m and f_m^R respectively, and $\pi_2\cdot\phi_j(m)$ is an acceptable index of $D_m^{\varepsilon_m}$.

<u>Theorem 2.4.4</u>

There is a recursive function *Ivlim* s.t. if j is a sequence index of $<D_m^{\varepsilon_m},(f_m,f_m^R)>$ then *Ivlim*(j) is an acceptable index of the inverse limit $(D_\infty,\varepsilon_\infty)$ ☐

To obtain further affirmative evidence for the notion of effective isomorphisms, let us examine if it is invariant under the inverse limit construction or not. Notice that unlike previously studied domain constructors, the inverse limit construction works not only on domains but also on computable projection pairs among them. Therefore we need the following notion to be preserved under the construction.

Definition 2.4.5

Given two effective sequences $<D_m^{\varepsilon_m},(f_m,f_m^R)>$ and $<D_m'^{\varepsilon_m'},(f_m',f_m'^R)>$ of computable projection pairs, we say that they are effectively isomorphic (in symbols $<D_m^{\varepsilon_m},(f_m,f_m^R)> \overset{e}{\cong} <D_m'^{\varepsilon_m'},(f_m',f_m'^R)>$) iff there are recursive functions u,v s.t. $u(m)$ is a recursive index of an effective isomorphism $i_m:D_m\to D_m'$ and $v(m)$ is a recursive index of the adjoint $j_m:D_m'\to D_m$; and $f_m'\cdot i_m=i_{m+1}\cdot f_m$, $f_m^R\cdot j_{m+1}=j_m\cdot f_m'^R$. □

Theorem 2.4.6

$<D_m^{\varepsilon_m},(f_m,f_m^R)> \overset{e}{\cong} <D_m'^{\varepsilon_m'},(f_m',f_m'^R)>$ implies $\varprojlim<D_m^{\varepsilon_m}> \overset{e}{\cong} \varprojlim<D_m'^{\varepsilon_m'}>$.

proof For the sake of simplicity we will prove this theorem for $\overset{e}{=}$ rather than $\overset{e}{\cong}$. Notice that we have assumed $D_m=D_m'$, $f_m=f_m'$, and $f_m^R=f_m'^R$. Let r_m, r_m' be recursive functions s.t. $\varepsilon_m'=\varepsilon_m\cdot r_m$ and $\varepsilon_m=\varepsilon_m'\cdot r_m'$ for every m. By the effective isomorphism of the sequences, there are recursive functions u,v s.t. $\phi_{u(m)}=r_m$ and $\phi_{v(m)}=r_m'$. Let a and b be as in the proof of 2.4.3. Then:

$$\varepsilon_\infty(n)=f_{a(n)\infty}\cdot\varepsilon_{a(n)}'(r_{a(n)}'\cdot b(n))$$

$$=f_{a(n)\infty}'\cdot\varepsilon_{a(n)}'(r_{a(n)}'\cdot b(n)) \quad (\because f_m=f_m' \text{ and } f_m^R=f_m'^R).$$

But there is a recursive Emb s.t. $\varepsilon_\infty(Emb(n,m))=f_n'\cdot\varepsilon_n'(m)$. Since $\phi_{v(m)}=r_m'$, we have a recursive function r_∞ s.t.:

$$\varepsilon_\infty(n)=\varepsilon_\infty'(r_\infty(n)).$$

Similarly $\varepsilon_\infty'(n)=\varepsilon_\infty(r_\infty'(n))$ for some recursive function r_∞.

◻

Notice finally that the proof of 2.4.3 has a further implication. Indeed we have constructed $\overline{f}_{m\infty}$ and $\overline{f}^R_{m\infty}$ from the effective sequence $\langle D^{\varepsilon}_m m, (f_m, f^R_m)\rangle$. Thus we have:

<u>Theorem 2.4.7</u>

There is a recursive function *Ucone* s.t. if j is a sequence index of $\langle D^{\varepsilon}_m m, (f_m, f^R_m)\rangle$ then $\pi_1 \cdot Ucone(j)$ is an acceptable index of λ_d and $\pi_2 \cdot Ucone(j)$ is an acceptable index of δ_d.

◻

2.5 Addendum

It should be noticed that if two effectively given domains are effectively isomorphic then we can effectively go from an acceptable index of one to that of the other. More formally:

<u>Theorem 2.5.1</u>

There is a recursive function *Trv* s.t. if $D^{\varepsilon} \overset{e}{=} D'^{\varepsilon'}$ via a computable isomorphism pair (h, h^R) with a directed index $\langle i,j\rangle$ and D^{ε} has an acceptable index n, then $D'^{\varepsilon'}$ has an acceptable index *Trv*(n,i,j). ◻

The effective isomorphism also has the following effect:

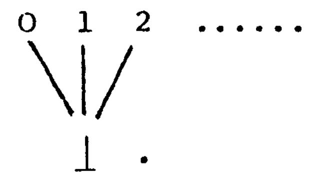
<u>Theorem 2.5.2</u>

There is a recursive function *Trf* s.t. if $D^{\varepsilon} \overset{e}{=} D'^{\varepsilon'}$ via a computable projection pair (h, h^R) with a directed index $\langle i,j\rangle$ and $x \in Comp(D^{\varepsilon})$ has a directed index n then $h(x) \in Comp(D'^{\varepsilon'})$ has a directed index *Trf*(n,i,j). ◻

One could ask if our notion of computability is really adequate. To answer this question, we exhibit how the conventional notion of computability can be embedded into our theory of effectively given domains.

<u>Definition 2.5.3</u>

Let $\mathbb{N}$ be the following ω-algebraic cpo:



All elements of $\mathbb{N}$ are finite and so $E_{\mathbb{N}}=\mathbb{N}$. Let us index $E_{\mathbb{N}}$ by $\varepsilon: N \to E_{\mathbb{N}}$ s.t.:

$$\varepsilon(0)=\bot$$

$$\varepsilon(n+1)=n \qquad\qquad n=0,1,2,\ldots\ .$$ □

It is evident that $(\mathbb{N},\varepsilon)$ is an effectively given domain and all elements of $\mathbb{N}$ are computable wrt ε.

The following theorem ensures that the conventional notion of computability is embedded into our notion of computability.

<u>Theorem 2.5.4</u>

(1) For every partial recursive function $\mathcal{f}: N \to N$, there is a computable (wrt $(\varepsilon,\varepsilon)$) extension $\mathrm{f}: \mathbb{N} \to \mathbb{N}$ of $\mathcal{f}$.

(2) For every function $\mathrm{f}: \mathbb{N} \to \mathbb{N}$ computable wrt $(\varepsilon,\varepsilon)$, there is a partial recursive function $\mathcal{f}: N \to N$ s.t. $\mathcal{f}$ is the restriction (on both domain and codomain) to N of f.

<u>proof</u> (1) Let $\mathcal{f}: N \to N$ be a partial recursive function. Define $\mathrm{f}: \mathbb{N} \to \mathbb{N}$ to be the following function:

$$\mathrm{f}(x) = \underline{\text{if}}\ x=\bot\ \underline{\text{then}}\ \bot$$
$$\underline{\text{else}}\ \underline{\text{if}}\ f(x)\downarrow\ \underline{\text{then}}\ f(x)$$
$$\underline{\text{else}}\ \bot.$$

Then we have:

$$\varepsilon(n) \sqsubseteq \mathrm{f}(\varepsilon(m))$$
$$\varepsilon(n) \sqsubseteq\ \underline{\text{if}}\ m=0\ \underline{\text{then}}\ \varepsilon(m)$$
$$\underline{\text{else}}\ \underline{\text{if}}\ f(m-1)\downarrow\ \underline{\text{then}}\ \varepsilon(f(m-1)+1)$$
$$\underline{\text{else}}\ \varepsilon(0).$$

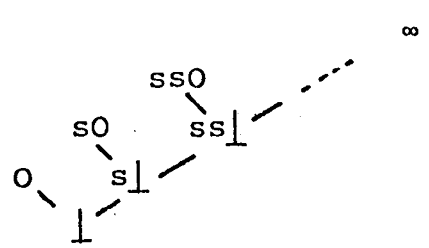
Therefore $\varepsilon(n)\sqsubseteq \mathrm{f}(\varepsilon(m))$ is r.e. in n and m. Thus f is a computable extension of f.

(2) Let $\mathrm{f}:\mathbb{N}\to\mathbb{N}$ be computable wrt $(\varepsilon,\varepsilon)$. Let $f:N\to N$ be the following partial function:

$$f(n) = \mu k.[k\neq 0\ \&\ \varepsilon(k)\sqsubseteq \mathrm{f}(\varepsilon(n+1))].$$

Obviously f is a partial recursive functionand f is a computable extension of f to $\mathbb{N}$.

Lehmann and Smyth [23] proposed the following ω-algebraic domain $\mathbb{N}'$ as a substitute to $\mathbb{N}$.



Here we regard s^nO as a natural number $n\in N$. Therefore we have: $N=\{s^nO \mid n\in N\}\subseteq \mathbb{N}'$. The basis of $\mathbb{N}'$ in symbols $E_{\mathbb{N}'}$ is the poset obtained from $\mathbb{N}'$ by removing its limit point ∞. Let us index $E_{\mathbb{N}'}$ by: $\varepsilon':N\to E_{\mathbb{N}'}$ s.t.:

$$\varepsilon'(2n)=s^n\bot \qquad n=0,1,2,\ldots$$
$$\varepsilon'(2n+1)=s^n0.$$

It is evident that (N',ε') is an effectively given domain s.t. $N'=Comp((N',\varepsilon'))$.

Can we establish a reasonable result similar to 2.5.4 for (N',ε')? Most likely the answer is negative. Indeed we can not show reasonalble continuous extension to N' of the number theoretic subtraction $\dot{-}$ s.t.:

$$n\dot{-}m=\underline{if}\ n\geq m\ \underline{then}\ n-m\ \underline{else}\ 0.$$

This difficulty is essentially due to the fact that, while N' has a structure which reflects the natural total ordering $n\leq n+1$ of natural numbers, most partial recursive functions are not monotone wrt $\leq$.

CHAPTER 3: EFFECTIVE DOMAINS

"In the end the program must still be run on a machine --- a machine which does not possess the benefit of 'abstract' human understanding, a machine that must operate with finite configurations."

Dana Scott, 1970
in Outline of a Mathematical Theory of Computation.

For a theory of computation it is at least desirable to handle only computable objects. In this chapter, we will observe that $(\mathrm{Comp}(D^{\varepsilon}), \varepsilon)$ behaves very well as a domain. A preliminary version of this chapter appeared in Kanda [3,4].

3.1 Effective Domains

An effectively algebraic (ef-algebraic) domain is a pair (X,ε) where X is a poset and ε is a total indexing of E_X s.t.:

(1) E_X has bounded joins.

(2) If $\varepsilon(W_j)$ is directed then $\sqcup\varepsilon(W_j)\in X$. We call such r.e. set W_j ε-directed.

(3) For every $x\in X$, there is a ε-directed r.e. set W^x s.t. $x=\sqcup\varepsilon(W^x)$.

We assume X has $\perp$. We call ε (or E_X^{ε}) an ef-algebraic basis of X.

Definition 3.1.1

An ef-algebraic domain (X,ε) is an effective domain iff the ef-algebraic basis E_X^{ε} is an effective poset. In this case E_X^{ε} is called an effective basis of X. The characteristic pair of E_X^{ε} will be called a characteristic pair of (X,ε). □

Notice that to within the renaming relation $\overset{r}{\cong}$, a characteristic pair uniquely determines an effective domain. Indeed we have:

$$(E_X,\varepsilon)\overset{r}{\cong}(E_{X'},\varepsilon') \quad \text{iff} \quad (X,\varepsilon)\overset{r}{\cong}(X',\varepsilon').$$

In case (ϕ_i,ϕ_j) is a characteristic pair, then we will write $\tilde{\xi}(<i,j>)$ to denote the effective domain determined by (ϕ_i,ϕ_j). We say $<i,j>$ is an <u>acceptable</u> <u>index</u> of $\tilde{\xi}(<i,j>)$.

The "directing function" does exist for every effective domain. Thus we can introduce the "directed indexing" χ_ε to every effective domain as we did for Comp(D^ε) (See 2.1.2). More precisely $\chi_\varepsilon(i)=\sqcup\varepsilon(W_{d_\varepsilon(i)})$.

By an <u>effectively</u> <u>directed</u> (<u>ef-directed</u>) <u>subset</u> of an effective domain X^ε, we mean a directed subset $Z\subseteq X$ s.t. $Z=\chi_\varepsilon(W)$ for some r.e. set W. We say that this W is <u>χ_ε-directed</u>. If W_j is χ_ε-directed then we say that $\chi_\varepsilon(W_j)$ has a <u>χ_ε-directed index</u> j.

By exactly the same argument as 2.1.11, we have:

<u>Theorem 3.1.2</u>

(1) An effective domain is ef-directed complete, i.e. every ef-directed subset has a lub.

(2) There is a recursive function *Lub* s.t. if $\chi_\varepsilon(W_j)$ has a χ_ε-directed index j, then $\chi_\varepsilon(Lub(j))=\sqcup\chi_\varepsilon(W_j)$. ☒

Notice that effective domains are not necessarily directed complete. Indeed RE, the set of all r.e. subsets of N with the set theoretical inclusion as a partial ordering is an effective domain but not directed complete, where the indexing of the basis is P_r.

<u>Definition 3.1.3</u>

Let X^ε and $X'^{\varepsilon'}$ be effective domains. A function $f:X\to X'$ is <u>fully</u> <u>computable</u>(<u>f-computable</u>) w.r.t. $(\varepsilon,\varepsilon')$ iff $\Gamma(f)$ is r.e. and f is <u>ef-continuous</u> w.r.t. $(\varepsilon,\varepsilon')$ i.e. f preserves lub's of ef-directed subsets. □

3.2 Effective Isomorphisms

Notice that the D^{ε} and $D'^{\varepsilon'}$ used in the proof of 2.2.1 are not only effectively given domains but also effective domains. Therefore we need the notion of effective isomorphisms as a criterion for identifying effective domains.

We define imbeddings and effective isomorphisms among the ef-algebraic bases of ef-algebraic domains exactly as we did in 2.2.3 for indexed domains. Given ef-algebraic domains X^{ε} and $X'^{\varepsilon'}$, and an effective imbedding $f:\varepsilon\to\varepsilon'$, let $\tilde{f}:X\to X'$ be the following extension: $\tilde{f}(\sqcup\varepsilon(W))=\sqcup f(\varepsilon(W))$ for every ε-directed W. Notice that $\tilde{f}$ is well-defined since f is an effective imbedding. We call such $\tilde{f}$ an <u>effective embedding</u> from X^{ε} to $X'^{\varepsilon'}$.

For an effective embedding of ef-algebraic domains, we can not expect more than monotonicity. In fact it could not be even an embedding though it is called an "effective embedding". But effective embeddings of effective domains enjoy much more interesting properties.

<u>Definition 3.2.1</u>

Let X^{ε} and $X'^{\varepsilon'}$ be effective domains. We say that a function $f:X\to X'$ is a <u>fully computable</u> (<u>f-computable</u>) <u>embedding</u> from ε to ε' iff f is f-computable w.r.t $(\varepsilon,\varepsilon')$ and there is a unique f-computable w.r.t. $(\varepsilon,\varepsilon')$ map $f^R:X'\to X$ s.t. $f\cdot f^R\sqsubseteq id_{X'}$, and $f^R\cdot f=id_X$. f^R is called a <u>f-computable projection</u> and (f,f^R) is called a f-computable projection pair from X^{ε} to $X'^{\varepsilon'}$. □

<u>Theorem 3.2.2</u>

Let X^{ε} and $X'^{\varepsilon'}$ be effective domains with an effective imbedding $f:\varepsilon\to\varepsilon'$, then the effective embedding $\tilde{f}$ is an f-computable embedding.

<u>proof</u> Obviously $\tilde{f}$ is monotone. We have:

$$\begin{aligned}\tilde{f}(\chi_\varepsilon(i)) &= \tilde{f}(\sqcup\varepsilon(W_{d_\varepsilon(i)}))\\ &= \sqcup f\cdot\varepsilon(W_{d_\varepsilon(i)})\\ &= \sqcup\varepsilon'\cdot r_f(W_{d_\varepsilon(i)})\\ &= \bigsqcup_n \varepsilon'\cdot r_f\cdot\phi_{dv2(d_\varepsilon(i))}(n)\\ &= \bigsqcup_n \varepsilon'\cdot\phi_{t(i)}(n) \quad \text{for some recursive function } t\\ &= \sqcup\varepsilon'(W_{t(i)})\\ &= \sqcup\varepsilon'(W_{d_{\varepsilon'}(t(i))}) \quad (\because W_{t(i)} \text{ is } \varepsilon'\text{-directed})\\ &= \chi_{\varepsilon'}(t(i)).\end{aligned}$$

Therefore given any χ_ε-directed set W, $\tilde{f}(\chi_\varepsilon(W))$ is an effectively directed subset of $X'^{\varepsilon'}$. Furthermore:

$$\begin{aligned}\tilde{f}(\sqcup\chi_\varepsilon(W)) &= \tilde{f}(\sqcup\varepsilon(\textstyle\bigcup_{i\in W} W_{d_\varepsilon(i)}))\\ &= \sqcup f\cdot\varepsilon(\textstyle\bigcup_{i\in W} W_{d_\varepsilon(i)})\\ &= \sqcup\{\sqcup f\cdot\varepsilon(W_{d_\varepsilon(i)}) \mid i\in W\}\\ &= \sqcup\{\tilde{f}(\sqcup\varepsilon(W_{d_\varepsilon(i)})) \mid i\in W\}\\ &= \sqcup\tilde{f}(\chi_\varepsilon(W)).\end{aligned}$$

Therefore f is ef-continuous w.r.t $(\varepsilon,\varepsilon')$. Let F be a finite subset of $\{\varepsilon(i) \mid f\cdot\varepsilon(i)\sqsubseteq\sqcup\varepsilon'(W)\}$ where W is ε'-directed. Thus f(F) is bounded by $\sqcup\varepsilon'(W)$. Therefore $\sqcup f(F)\in E_{X'}$. Since f is an effective imbedding $\sqcup F\in E_X$. Evidently $f(\sqcup F)=\sqcup f(F)\sqsubseteq\sqcup\varepsilon'(W)$. Thus $\sqcup F\in$ $\{\varepsilon(i) \mid f\cdot\varepsilon(i)\sqsubseteq\sqcup\varepsilon'(W)\}$. Thus $\{\varepsilon(i) \mid f\cdot\varepsilon(i)\sqsubseteq\sqcup\varepsilon'(W)\}$ is directed. Furthermore $\{i \mid f\cdot\varepsilon(i)\sqsubseteq\sqcup\varepsilon'(W)\}=\{i \mid \varepsilon'\cdot r_f(i)\sqsubseteq\varepsilon'(n) \text{ for some } n\in N\}$. Thus this set is r.e. Therefore $\{\varepsilon(i) \mid f\cdot\varepsilon(i)\sqsubseteq\sqcup\varepsilon'(W)\}$ is ef-directed subset of E_X^ε. Now define $\tilde{f}^R: X'\to X$ by:

$$\tilde{f}^R(\sqcup\varepsilon'(W))=\sqcup\{\varepsilon(i) \mid f\cdot\varepsilon(i)\sqsubseteq\sqcup\varepsilon'(W)\}$$

where W is ε'-directed. Thence we have:

$$\begin{aligned}\tilde{f}^R\cdot\tilde{f}(\sqcup\varepsilon(W)) &= \tilde{f}^R(\sqcup f(\varepsilon(W)))\\ &= \sqcup\{\varepsilon(i) \mid f\cdot\varepsilon(i)\sqsubseteq\sqcup f\cdot\varepsilon(W)\}\end{aligned}$$

$=\sqcup\{\varepsilon(i)\,|\,i\in W\}$

$=\sqcup\varepsilon(W)$.

Also $\tilde{f}\cdot\tilde{f}^R(\sqcup\varepsilon'(W))=\tilde{f}(\sqcup\{\varepsilon(i)\,|\,f\cdot\varepsilon(i)\sqsubseteq\sqcup\varepsilon'(W)\})$

$=\sqcup\{f\cdot\varepsilon(i)\,|\,f\cdot\varepsilon(i)\sqsubseteq\sqcup\varepsilon'(W)\}\sqsubseteq\sqcup\varepsilon'(W)$.

It can readily be seen that f^R is ef-continuous w.r.t. $(\varepsilon',\varepsilon)$. It is evident that $\varepsilon'(m)\sqsubseteq\tilde{f}(\varepsilon(n))$ is recursive in m and n. Therefore $\tilde{f}$ is f-computable w.r.t. $(\varepsilon,\varepsilon')$. Also we have:

$\varepsilon(m)\sqsubseteq\tilde{f}^R\cdot\varepsilon'(n)$.

$\Leftrightarrow$ $\varepsilon(m)\sqsubseteq\sqcup\{\varepsilon(i)\,|\,f\cdot\varepsilon(i)\sqsubseteq\varepsilon'(n)\}$.

$\Leftrightarrow$ $\varepsilon(m)\sqsubseteq\varepsilon(i)$ and $f\cdot\varepsilon(i)\sqsubseteq\varepsilon'(n)$ for some i.

$\Leftrightarrow$ $\varepsilon(m)\sqsubseteq\varepsilon(i)$ and $\varepsilon'\cdot r_f(i)\sqsubseteq\varepsilon'(n)$ for some i.

Therefore $\varepsilon(m)\sqsubseteq\tilde{f}^R\cdot\varepsilon'(n)$ is r.e. in m and n. Thus f^R is f-computable w.r.t. $(\varepsilon',\varepsilon)$. ☐

By exactly the same argument as in 2.2.6 we have:

Theorem 3.2.3

Let X^ε and $X'^{\varepsilon'}$ be effective domains s.t. $f:X\to X'$ is an f-computable embedding then f is an effective embedding. ☐

Despite discouragingly poor character of effective embeddings of ef-algebraic domains, effective isomorphisms of them are quite interesting.

Lemma 3.2.4

Let X^ε and $X'^{\varepsilon'}$ be ef-algebraic domains s.t. $f:\varepsilon\to\varepsilon'$ is an effective isomorphism. Then $E_X\simeq E_{X'}$ via f and $X\simeq X'$ via $\tilde{f}$.

proof Evidently $f,f^{-1},\tilde{f},\tilde{f}^{-1}$ are monotone. Since both f and f^{-1} are injective, $f\cdot f^{-1}=id_{E_{X'}}$ and $f^{-1}\cdot f=id_{E_X}$. Thus $E_X=E_{X'}$ via f. Now we have $\tilde{f}\cdot\tilde{f}^{-1}(\sqcup\varepsilon'(W))=\tilde{f}(\sqcup f^{-1}\cdot\varepsilon'(W))=\sqcup f\cdot f^{-1}\cdot\varepsilon'(W)=\sqcup\varepsilon'(W)$. Similarly $\tilde{f}^{-1}\cdot\tilde{f}(\sqcup\varepsilon(W))=\sqcup\varepsilon(W)$. Thus $X=X'$ via $\tilde{f}$. ☐

In case either X^{ε} and $X'^{\varepsilon'}$ is an effective domain and $\varepsilon \overset{e}{\cong} \varepsilon'$ via f then both of them are effective domains and $X \overset{e}{\cong} X'$ via f. Notice that an f-computable isomorphism is an isomorphism which is f-computable as well as its adjoint.

3.3 Effective Completion

In this section, we will observe that every effective domain can be characterized as $(\mathrm{Comp}(D^{\varepsilon}),\varepsilon)$ for some effectively given domain D^{ε}.

Definition 3.3.1

The effective completion of an effective poset (B,ε) is a poset $(\tilde{B}^{\varepsilon},\sqsubseteq)$ together with a total indexing $\bar{\varepsilon}:N\to\tau(B)$ where $\bar{\varepsilon}$ is as in 2.1.1, and $\tilde{B}^{\varepsilon}=\{\varepsilon(W)\,|\,W \text{ is } \varepsilon\text{-directed}\}$. We will write $\tilde{B}$ for $\tilde{B}^{\varepsilon}$ if ε is evident from the context. □

Theorem 3.3.2 (The Effective Completion Theorem I)

Given an effective poset (B,ε) we have:

(1) $\tau(B)=E_{\tilde{B}}$ and $(\tau(B),\bar{\varepsilon})=(B,\varepsilon)$.

(2) For every $x\in\tilde{B}$, there is a $\bar{\varepsilon}$-directed r.e. set W^{x} s.t.:

$$x=\sqcup\bar{\varepsilon}(W^{x})=\sqcup\tau\cdot\varepsilon(W^{x}).$$

(3) $(\bar{B},\bar{\varepsilon})=(\mathrm{Comp}(\bar{B},\bar{\varepsilon}),\bar{\varepsilon})$.

(4) $(B,\bar{\varepsilon})$ is an effective domain.

proof (1) $(\tau(B),\bar{\varepsilon})$ is evidently a renaming of (B,ε). For $\tau(B)=E_{\tilde{B}}$ notice that $\tau(B)=E_{\bar{B}}$ and $\tilde{B}^{\varepsilon}\subseteq\bar{B}$.

(2) $x=\{\varepsilon(i)\,|\,i\in W\}$ for some ε-directed r.e. set W. Let $J_x=\tau(x)=\{\tau\cdot\varepsilon(i)\,|\,i\in W\}$. Since $\tau(B)=B$, W is $\bar{\varepsilon}$-directed. Also we have:

$$x=\sqcup J_x=\sqcup\bar{\varepsilon}(W)=\sqcup\tau\cdot\varepsilon(W).$$

(3) By (2) we have $\tilde{B}^{\varepsilon} \subseteq \mathrm{Comp}(\bar{B},\bar{\varepsilon})$. But by the definition of $\tilde{B}^{\varepsilon}$, $\mathrm{Comp}(\bar{B},\bar{\varepsilon}) \subseteq \tilde{B}^{\varepsilon}$.

(4) By (1) and (2). ☐

Theorem 3.3.3 (The Effective Completion Theorem II)

Let (X,ε) be an effective domain. Then (E_X,ε) is an effective poset and $(\tilde{E}_X,\bar{\varepsilon})=(X,\varepsilon)$.

proof Define $\theta:X\to\tilde{E}_X$ by $\theta(x)=\{e\in E_X | e\sqsubseteq x\}$.* Evidently θ is an isomorphism and $\theta\cdot\varepsilon=\bar{\varepsilon}$. Therefore $(\tilde{E}_X,\bar{\varepsilon})=(X,\varepsilon)$. ☐

The following Corollary immediately follows from 3.3.2 and 3.3.3:

Corollary 3.3.4 (The Characterization Theorem)

An ef-algebraic domain is an effective domain iff it is the effective completion of an effective poset iff it is $(\mathrm{Comp}(D^{\varepsilon}),\varepsilon)$ for some effectively given domain D^{ε}.

Notice that we have observed that effective posets are exactly the effective bases of effective(ly given) domains. We can make this more explicit as follows:

Theorem 3.3.5 (The Acceptable Indexing Theorem)

$$\tilde{\xi}(i) = \widetilde{\xi(i)} = \mathrm{Comp}(\bar{\xi}(i))$$

where $\widetilde{\xi(i)}$ is the effective completion of $\xi(i)$. ☐

---

* This θ is well-defined because $\varepsilon(i)\sqsubseteq x=\sqcup\varepsilon(W^x)$ iff $\varepsilon(i)\sqsubseteq\varepsilon(j)$ for some $j\in W^x$, since $\varepsilon(i)$ is compact.

3.4 Domain Constructors

Let X^{ε} and $X'^{\varepsilon'}$ be effective domains. We define $X^{\varepsilon}\times X'^{\varepsilon'}$ and $X^{\varepsilon}+X'^{\varepsilon'}$ by:

$$X^{\varepsilon}\times X'^{\varepsilon'}=(X\times X')^{\varepsilon\times\varepsilon'}.$$

$$X^{\varepsilon}+X'^{\varepsilon'}=(X+X')^{\varepsilon+\varepsilon'}$$

where $\varepsilon\times\varepsilon'$ and $\varepsilon+\varepsilon'$ are as before. By 3.3.5 $X^{\varepsilon}=\mathrm{Comp}(\overline{E}_X,\overline{\varepsilon})$. But evidently $\mathrm{Comp}(\overline{E}_X,\overline{\varepsilon})\times\mathrm{Comp}(\overline{E}_{X'},\overline{\varepsilon}')=\mathrm{Comp}(\overline{E}_X^{\varepsilon}\ \overline{E}_{X'}^{\varepsilon'})$. Thus $X^{\varepsilon}\times X'^{\varepsilon'}$ is an effective domain. Similarly $X^{\varepsilon}+X'^{\varepsilon'}$ is an effective domain.

The problem of function space is not so straightforward because effective domains are not necessarily cpo's.

<u>Definition 3.4.1</u>

Let X^{ε} and $X'^{\varepsilon'}$ be effective domains. Define $(X^{\varepsilon}\to X'^{\varepsilon'})$ to be $(X\to X')^{(\varepsilon\to\varepsilon')}$ where $(X\to X')$ is t he set of all f-computable (w.r.t. $(\varepsilon,\varepsilon')$) functions with the pointwise ordering, and $(\varepsilon\to\varepsilon')$ is the following total indexing of $E_{(X\to X')}$:

$(\varepsilon\to\varepsilon')(n)=$ <u>if</u> $\sigma(n)$ has a lub <u>then</u> $\bigsqcup\sigma(n)$ <u>else</u> $\perp$

where $\sigma(n)=\{(\varepsilon(i),\varepsilon(j))\mid\langle i,j\rangle\in P_r(n)\}$ and

$(e,e')(x)=$ <u>if</u> $x\sqsupseteq e$ <u>then</u> e' <u>else</u> $\perp$. □

<u>Lemma 3.4.2</u>

Let X^{ε} and $X'^{\varepsilon'}$ be effective domains. $h:X\to X'$ is f-computable w.r.t $(\varepsilon,\varepsilon')$ iff it is the restriction to $X=\mathrm{Comp}(\overline{E}_X^{\varepsilon})$ of a function $\overline{E}_X\to\overline{E}_{X'}$ which is computable w.r.t. $(\overline{\varepsilon},\overline{\varepsilon}')$.

<u>proof</u> Necessity is trivial. We prove sufficiency. Assume $h:\mathrm{Comp}(\overline{E}_X^{\varepsilon})\to\mathrm{Comp}(\overline{E}_{X'}^{\varepsilon'})$ is f-computable w.r.t. $(\overline{\varepsilon},\overline{\varepsilon}')$. Evidently $h\upharpoonright\tau(E_X):\tau(E_X)\to\overline{E}_{X'}$ is monotone. Thus $\Psi_h:\overline{E}_X\to\overline{E}_{X'}$ s.t. $\Psi_h(\bigsqcup\overline{\varepsilon}(W))=\bigsqcup h\cdot\overline{\varepsilon}(W)$ for all $\overline{\varepsilon}$-directed W, is the unique continuous extension of $h\upharpoonright\tau(E_X)$. Since h is f-computable w.r.t. $(\overline{\varepsilon},\overline{\varepsilon}')$, Ψ_h is computable w.r.t. $(\overline{\varepsilon},\overline{\varepsilon}')$. Also $\Psi_h(\bigsqcup\overline{\varepsilon}(W))=\bigsqcup h\cdot\overline{\varepsilon}(W)=h(\bigsqcup\overline{\varepsilon}(W))$ for

every ε-directed W. Notice that the second equality is due to the fact that we can effectively go from the effective indexing $\overline{\varepsilon}$ to the directed indexing $\chi_{\overline{\varepsilon}}$. Thus Ψ_h is the computable extension of h. ☐

Theorem 3.4.3

Let X^{ε} and $X'^{\varepsilon'}$ be effective domains. We have:

$$(X^{\varepsilon} \to X'^{\varepsilon'}) = (\mathrm{Comp}([\overline{E}^{\varepsilon}_{X} \to \overline{E}^{\varepsilon'}_{X'}]), [\overline{\varepsilon} \to \overline{\varepsilon}']).$$

Therefore $(X^{\varepsilon} \to X'^{\varepsilon'})$ is an effective domain.

proof Define $\alpha : \mathrm{Comp}([\overline{E}^{\varepsilon}_{X} \to \overline{E}^{\varepsilon'}_{X'}]) \to ((\mathrm{Comp}(\overline{E}^{\varepsilon}_{X}) \to \mathrm{Comp}(\overline{E}^{\varepsilon'}_{X'}))$ by $\alpha(h) = h \upharpoonright \mathrm{Comp}(\overline{E}^{\varepsilon}_{X})$. Then α is an isomorphism with the adjoint β s.t. $\beta(h) = \Psi_h$. Evidently $\alpha[\overline{\varepsilon} \to \overline{\varepsilon}'] = (\overline{\varepsilon} \to \overline{\varepsilon}')$. Therefore up to $\overset{r}{\cong}$ we have established the theorem. ☐

By 2.1.4, 2.3.3, 3.4.3. and 3.3.5, we immediately have:

Theorem 3.4.4

Let *Prod*, *Sum*, and *Func* be as in 2.1.4. Then we have:

(1) $\tilde{\xi}(i) \times \tilde{\xi}(j) = \tilde{\xi}(Prod(i,j))$.

(2) $\tilde{\xi}(i) + \tilde{\xi}(j) = \tilde{\xi}(Sum(i,j))$.

(3) $(\tilde{\xi}(i) \to \tilde{\xi}(j)) = \tilde{\xi}(Func(i,j))$. ☐

Notice that the directed indexing $\zeta_{[\overline{\varepsilon} \to \overline{\varepsilon}']}$ of $(\mathrm{Comp}[\overline{E}^{\varepsilon}_{X} \to \overline{E}^{\varepsilon'}_{X'}], [\overline{\varepsilon} \to \overline{\varepsilon}'])$ is equivalent to the directed indexing $\chi_{(\varepsilon \to \varepsilon')}$ of $(X^{\varepsilon} \to X'^{\varepsilon'})$ in such a sense as $\alpha(\zeta(i)) = \chi(i)$. Therefore from 2.1.7 and 2.1.9 we immediately have:

Lemma 3.4.5

(1) Every f-computable function maps recursively in directed indices. Indeed we have:

$$\chi_{\xi(i)}(Apply'(i,j,k,m)) = \chi_{(\xi(i) \to \xi(j))}(k)(\chi_{\xi(i)}(m)).$$

(2) Given effective domains X^{ε} and $X'^{\varepsilon'}$, an ef-continuous function $X \to X'$ which maps recursively in directed indices is an

f-computable function w.r.t. $(\varepsilon,\varepsilon')$.

(3) The composition of f-computable function is recursive in directed indices uniformly in the ranges and domains of the functions to be composed. ◻

(2)-3.4.5 immediately implies that the composition of two effective embeddings is again an effective embedding for we have 3.2.2 and 3.2.3.

We can introduce the recursive indices of effective embeddings of effective domains as we did for effectively given domains. Remember that in 3.2.2 and 3.2.3, we have estabilished the equivalence of effective embeddings and f-computable embeddings of effective domains. Now with directed indexings for effective domains, by exactly the same arguments as in 2.2.7, we can establish the "effective" equivalence of effective embeddings and f-computable embeddings. Indeed We have:

<u>Lemma 3.4.6</u>

Let *Rd*, *De*, and *Dp* be as in 2.2.7. Then we have:

(1) If i and j are directed indices of an f-computable embedding $f\in(\tilde{\xi}(k)\to\tilde{\xi}(m))$ and the adjoint $g\in(\tilde{\xi}(m)\to\tilde{\xi}(k))$ respectively then *Rd*(i,j,k,m) is a recursive index of f.

(2) If i is a recursive index of an effective embedding $f\in(\tilde{\xi}(j)\to\tilde{\xi}(k))$, then *De*(i,j,k) is a directed index of f and *Dp*(i,j,k) is a directed index of its adjoint.

◻

It is obvious that we have similar results to 2.5.1 and 2.5.2 for effective domains.

3.5 Effective Inverse Limitss

We can characterize the effectiveness of ω-sequences of f-computable projection pairs of effective domains as in 2.4.1 and 2.4.2. Even though an ω-sequence of f-computable projection pairs is effective, the inverse limit construction gives us a poset which is not countable. Therefore we need a notion of "effective" inverse limits which will cut down the cardinal of the limits to $\leq\omega$.

Definition 3.5.1

The effective inverse limit of an effective sequence $<X_m^{\varepsilon_m},(f_m,f_m^R)>$ of f-computable projection pairs of effective domains is a pair $(X_\infty,\varepsilon_\infty)$ where X_∞ is a poset $\{<x_m>|x_m=g_m(x_{m+1})$, there is a recursive function q s.t. $q(m)$ is a directed index of $x_m\}$ with the coordinatewise ordering, and ε_∞ is defined as in section 4 of capter 2. We will write ef-$\varprojlim<X_m^{\varepsilon_m},(f_m,f_m^R)>$ or ef-$\varprojlim<X_m>$ for $(X_\infty,\varepsilon_\infty)$. Evidently ef-$\varprojlim<X_m^{\varepsilon_m},(f_m,f_m^R)>$ is an ef-algebraic domain. □

Theorem 3.5.2

Let $<X_m^{\varepsilon_m},(f_m,f_m^R)>$ be an effective sequence, then the effective inverse limit of it is an effective domain.

proof Evidently ef-$\varprojlim<X_m^{\varepsilon_m},(f_m,f_m^R)>=$ef-$\varprojlim<\mathrm{Comp}(\overline{E}_{X_m}^{\varepsilon_m})^{\overline{\varepsilon}_m},(f_m,f_m^R)>$. Thus it is sufficient to show:

$$\text{ef-}\varprojlim<\mathrm{Comp}(\overline{E}_{X_m}^{\varepsilon_m})^{\overline{\varepsilon}_m},(f_m,f_m^R)>=(\mathrm{Comp}(\varprojlim<\overline{E}_{X_m}^{\varepsilon_m},(\Psi_{f_m},\Psi_{f_m^R})>),\overline{\varepsilon}_\infty)$$

where Ψ_{f_m} is the unique computable extension of f_m. To simplify notations we will write $\overline{f}_m$ for Ψ_{f_m}. Let $d=\langle d_m\rangle\in$ef-$\varprojlim<\mathrm{Comp}(\overline{E}_{X_m}^{\varepsilon_m})^{\overline{\varepsilon}_m},(f_m,f_m^R)>$. There is a recursive function q s.t. $d_m=\chi_{\overline{\varepsilon}_m}(q(m))$. Also $d=\bigsqcup_m f_{m\infty}(d_m)$. Notice that $f_{m\infty}=\overline{f}_{m\infty}\rceil\mathrm{Comp}(\overline{E}_{X_m}^{\varepsilon_m})$.

Also by 2.4.3 there is a recursive function λ_d s.t. $\lambda_d(m)$ is a directed index of $\overline{f}_{m\infty}$. Therefore we have:

$$\begin{aligned} f_{m\infty}(d_m) &= \overline{f}_{m\infty}(d_m) \\ &= (\zeta_{[\overline{\varepsilon}_m \to \overline{\varepsilon}_\infty]}(\lambda_d(m))(\zeta_{\overline{\varepsilon}_m}(q(m)) \\ &= \zeta_{\overline{\varepsilon}_\infty}(t(m)) \quad \text{for some recursive } t. \end{aligned}$$

Thus $d \in \mathrm{Comp}(\underleftarrow{\lim}\langle \overline{E}_X^{\varepsilon}m, (\overline{f}_m, \overline{f}_m^R)\rangle)$.

Conversely let $c = \langle c_m \rangle \in \mathrm{Comp}(\underleftarrow{\lim}\langle \overline{E}_{X_m}^{\overline{\varepsilon}}m, (\overline{f}_m, \overline{f}_m^R)\rangle)$. By the computability of $\overline{f}_{m\infty}^R$, we have: $c_m = \overline{f}_{m\infty}^R(c) \in \mathrm{Comp}(\overline{E}_{X_m}^{\overline{\varepsilon}}m)$. Therefore $c_m = f_m^R(c_{m+1})$ for every m. By 2.4.3. there is a recursive δ_d s.t. $\delta_d(m)$ is a directed index of $\overline{f}_{m\infty}^R$. Therefore $f_{m\infty}^R = \overline{f}_{m\infty}^R \upharpoonright \mathrm{Comp}(\overline{E}_{X_m}^{\varepsilon}m) = \chi_{[\varepsilon_\infty \to \varepsilon_m]}(\lambda_d(m))$. Thus we have:

$$\begin{aligned} c_m = \overline{f}_{m\infty}^R(c) &= f_{m\infty}^R(c) \\ &= \chi_{[\overline{\varepsilon}_\infty \to \overline{\varepsilon}_m]}(\delta_d(m))(c) \\ &= \chi_{[\overline{\varepsilon}_\infty \to \overline{\varepsilon}_m]}(\delta_d(m))(\chi_{\overline{\varepsilon}_\infty}(k)) \\ &= \chi_{\overline{\varepsilon}_\infty}(g(m)) \quad \text{for some recursive function } g. \end{aligned}$$

where $c = \chi_{\overline{\varepsilon}_\infty}(k)$. Therefore we have:

$$c \in \mathrm{ef}\text{-}\underleftarrow{\lim}\langle \mathrm{Comp}(\overline{E}_{X_m}^{\overline{\varepsilon}}m)^{\overline{\varepsilon}_m}, (f_m, f_m^R)\rangle. \qquad \square$$

The invariance of $\overset{e}{\cong}$ under the domain constructors ×,+,→ and the effective inverse limits can be checked by almost the same arguments as in the previous chapter.

Also notice that exactly the same recursive function *Ivlim* as in 2.4.4 establishes the same theorem for effective domains. Furthermore the same recursive function *Ucone* establishes the same theorem as 2.4.7 for effective domains.

CHAPTER 4: EFFECTIVENESS IN SFP OBJECTS

In 1.3, we have observed that the class of SFP objects properly contains that of ω-algebraic domains. In this chapter we will establish a class of effectively given SFP objects which properly contains that of effectively given domains. This generalization has a significance. Our intuitive understanding of computability requires every finite object to be computable in some suitable sense. Therefore every finite cpo should be effectively given by some suitable effective indexing. But this is not the case in the theory of effectively given domains , because most of the finite cpo's are not bounded complete. Remember that every finite cpo is an SFP object. Therefore SFP is a more appropriate class on which effectiveness should be studied. Effectiveness arguments for SFP objects were predicted as routine extensions of those for effectively given domains by several people, but as we will observe later, it turned out to be a far from routine extension and to involve substantial developments.

4.1 Effectively Given SFP Objects

Since SFP objects can be characterized as ω-algebraic cpo's satisfying the SFP condition on bases, the following definition is natural.

Definition 4.1.1

(1) An indexed SFP object is a pair (D,ε) where D is an SFP object and $\varepsilon:N\to E_D$ is a total indexing. ε is said to be effective iff there is a pair (c,d) of recursive predicates (called the characteristic pair) s.t.

$c(x,m)$ iff $\mathrm{Card}(U_{E_D}(\varepsilon(f_s(x))))=m$

$d(k,x)$ iff $\varepsilon(k)\in U_{E_D}(\varepsilon(f_s(x)))$.

(2) An indexed SFP object (D,ε) is <u>effectively given</u> iff ε is effective. E_D^ε will be called the effective basis of D^ε.

(3) Given an effectively given SFP object D^ε, $x\in D$ is <u>computable</u> w.r.t. ε (or computable in D^ε) iff for some r.e. set W, $\varepsilon(W)$ is directed and $x=\bigsqcup\varepsilon(W)$. We say that an r.e. set W is <u>ε-directed</u> if $\varepsilon(W)$ is directed. The set of all computable elements of D^ε will be denoted by $\mathrm{Comp}(D^\varepsilon)$.

(4) Given effectively given SFP objects D^ε and $D'^{\varepsilon'}$, a continuous function $f:D\to D'$ is <u>computable</u> w.r.t. $(\varepsilon,\varepsilon')$ iff $\Gamma(f)=\{\langle n,m\rangle \mid \varepsilon'(m)\sqsubseteq f\cdot\varepsilon(n)\}$ is r.e.

As for effectively given domains, to within the renaming relation $\overset{r}{\cong}$, a characteristic pair uniquely determines an effectively given SFP object. If $\langle\phi_i,\phi_j\rangle$ is a characteristic pair, we will write $\bar\rho(\langle i,j\rangle)$ to denote the effectively given SFP object determined by this characteristic pair. Also we say that $\bar\rho(\langle i,j\rangle)$ has an <u>acceptable index</u> $\langle i,j\rangle$.

Notice that if D^ε is an effectively given SFP object, then the predicate "$\mathrm{Card}(U^*_{E_D}(\varepsilon(f_s(x))))=m$" and "$\varepsilon(k)\in U^*_{E_D}(\varepsilon(f_s(x)))$" are recursive in (x,m) and (k,x) respectively.

<u>Lemma 4.1.2</u> There is a recursive function *Conv* s.t. if n is an acceptable index of an effectively given domain then *Conv*(n) is an acceptable index of it as an effectively given SFP object.

<u>proof</u> Let D^ε be an effectively given domain with the characteristic pair (b,l). $\mathrm{Card}(U_{E_D}(\varepsilon(f_s(x))))=m>0$ iff $m=1$. And indeed $m=1$ iff $\varepsilon(f_s(x))$ is bounded iff $b(x)$ is true.

Thus $c(x,m) \Leftrightarrow$ <u>if</u> m>1 <u>then</u> 0 <u>else</u> m=1 & $b(x)$.

Thus c is a recursive predicate. Also we have:

$$d(k,x) \Leftrightarrow \varepsilon(k) \in U_{E_D}(\varepsilon(f_s(x)))$$

$$\Leftrightarrow \varepsilon(k) = \sqcup \varepsilon(f_s(x))$$

$$\Leftrightarrow l(k,x).$$

Thus d is a recursive predicate. ◻

We have an alternative characterization of the effective bases of SFP objects. In fact $\varepsilon: N \to E_D$ is effective iff there are recursive functions r and s s.t.:

$$r(m,n) \Longleftrightarrow \varepsilon(m) = \varepsilon(n),$$

$$\varepsilon \cdot f_s \cdot s = U_{E_D} \cdot \varepsilon \cdot f_s.$$

<u>Lemma 4.1.3</u>

For every effectively given SFP object D^ε there is a recursive function, called the <u>directing function</u>, d_ε s.t. for every $j \in N$, $W_{d_\varepsilon(j)}$ is ε-directed and if W_j is already ε-directed then $\sqcup\varepsilon(W_j) = \sqcup\varepsilon(W_{d_\varepsilon(j)})$.

<u>proof</u> $\varepsilon(f_s(x))$ bounded iff $Card(U_{E_D}(\varepsilon(f_s(x)))) > 0$. Also $\varepsilon(k) = \sqcup\varepsilon(f_s(x))$ iff $Card(U_{E_D}(\varepsilon(f_s(x)))) = 1$ and $\varepsilon(k) \in U_{E_D}(\varepsilon(f_s(x)))$. Thus the same proof as for 2.1.2 establishes this lemma. In fact this d_ε is the same as in 2.1.2. ◻

By virtue of the above lemma, we can introduce a total indexing called the <u>directed indexing</u> to $Comp(D^\varepsilon)$ for every effectively given SFP object D^ε. If $x = \sqcup\varepsilon(W_{d_\varepsilon(j)})$ we say that x has a <u>directed index</u> j and denote it by $x = \zeta_\varepsilon(j)$. Notice that this indexing coincied with the ζ_ε for effectively given domains in case D^ε is an effectively given domain.

By exactly the same arguments, we have the same results as 2.1.9 and 2.1.11 for effectively given SFP objects. Also the composition of computable functions is recursive in graph indicies.

4.2 Effective Embeddings & Effective Isomorphisms

Since every effectively given domain is an effectively given SFP object, the theorem 2.2.1 call for effective embeddings and effective isomorphisms for effectively given SFP objects.

Definition 4.2.1

(1) Let D^{ε} and $D'^{\varepsilon'}$ be indexed SFP objects. A function $f:E_D \to E_{D'}$ is an effective imbedding from ε to ε'(in symbols $f:\varepsilon \to \varepsilon'$) iff

1. f is an imbedding from E_D to $E_{D'}$,
2. There is a recursive function r_f s.t. $f \cdot \varepsilon = \varepsilon' \cdot r_f$.

Remember "imbeddings" are defined in 1.3.13.

(2) Let D^{ε} and $D'^{\varepsilon'}$ be indexed SFP objects. We say that they are effectively isomorphic (in symbols $D^{\varepsilon} \overset{e}{\cong} D'^{\varepsilon'}$ or $\varepsilon \overset{e}{\cong} \varepsilon'$) iff there exists an effective imbedding $f:\varepsilon \to \varepsilon'$ s.t. f^{-1} is also an effective imbedding from ε' to ε. □

It is quite clear that 4.2.1 coincides with 2.2.3 whenever D^{ε} and $D'^{\varepsilon'}$ are effectively given domains.

As for effectively given domains, we can define effective embeddings to be the unique continuous extensions of effective imbeddings. Also a computable embedding is an embedding which is computable as well as its adjoint, which is called a computable projection. A computable projection pair is a pair of a computable embedding and its adjoint. Evidently all of these notions coincide with 2.2.5 whenever we are considering effectively given domains.

Theorem 4.2.2

(1) Let D^{ε} and $D'^{\varepsilon'}$ be indexed SFP objects, and $f: E_D \to E_{D'}$ be an effective imbedding. Then the continuous extension $\overline{f}$ of f to D is a continuous embedding with the adjoint given by: $g(y) = \bigsqcup\{e \in E_D \mid f(e) \sqsubseteq y\}$. Furthermore $f^{-1} = g \upharpoonright \overline{f}(E_D)$.

(2) In case D^{ε} and $D'^{\varepsilon'}$ are effectively given SFP objects, $\overline{f}$ is computable w.r.t. $(\varepsilon, \varepsilon')$ and g is computable w.r.t. $(\varepsilon', \varepsilon)$.

(3) Let D^{ε} and $D'^{\varepsilon'}$ be effectively given SFP objects s.t. $f: D \to D'$ is a computable embedding, then f is an effective embedding from ε to ε'.

proof Similarly to the proofs of 2.2.4 and 2.2.6. ◻

Note that 4.2.2 immediately implies that the class of effective embeddings is closed under composition for the class of computable maps is closed under it.

It is quite obvious that if $D^{\varepsilon} \overset{e}{\cong} D'^{\varepsilon'}$ and either of them is an effectively given SFP object then both of them are, and $\mathrm{Comp}(D^{\varepsilon}) \cong \mathrm{Comp}(D'^{\varepsilon'})$.

The coincidence of effective embeddings and computable embeddings is "effective".

Theorem 4.2.3

(1) There is a recursive function *Rg* s.t. if i and j are graph indices of a computable embedding $f: \overline{\rho}(k) \to \overline{\rho}(m)$ and its adjoint $f^R: \overline{\rho}(m) \to \overline{\rho}(k)$ respectively, then *Rg*(i,j,k,m) is a recursive index of f.

(2) There are recursive functions *Ge*,*Gp* s.t. if i is a recursive index of an effective embedding $f:\overline{\rho}(j)\to\overline{\rho}(k)$ then *Ge*(i,j,k) is a graph index of f and *Gp*(i,j,k) is a graph index of the adjoint $f^R:\overline{\rho}(k)\to\overline{\rho}(j)$. ☐

Notice that at this moment, we can not make 4.2.3 as in 2.2.7 because we do not know if the function spaces of effectively given SFP objects are effectively given or not yet. But 4.2.3 is a generalization of 2.2.7 since we can effectively go back and forth among directed indexing and graph indexing in effectively given domains.

4.3 Algebraic Completion

Definition 4.3.1

A finitary poset $(E,\sqsubseteq)$ together with a total indexing $\varepsilon:N\to E$ is called an indexed finitary poset. An indexed finitary poset (E,ε) is an effective finitary poset iff ε is effective in such a sense that it has a characteristic pair. By the (algebraic) completion of an indexed poset (E,ε), we mean an indexed SFP object $(\overline{E},\overline{\varepsilon})$ where $\overline{E}$ is the algebraic completion of E and $\overline{\varepsilon}:N\to\tau(E)$ is defined by: $\overline{\varepsilon}(n)=\tau\cdot\varepsilon(n)$ where τ is the canonical map from E to $\overline{E}$.

Theorem 4.3.2

(1) Let (E,ε) be an effective finitary poset. Then the completion of it is an effectively given SFP object.

(2) Given an effectively given SFP object D^{ε}, (E_D,ε) is an effective finitary poset and $(D,\varepsilon)=(\overline{E}_D,\overline{\varepsilon})$.

(3) An indexed SFP object is an effectively given SFP object iff it is the completion of some effective finitary poset. ☐

The above theorem indicates that the effective bases of effectively given SFP objects are exactly effective finitary posets. We will make this point more explicit. To within the renaming relation $\overset{r}{\cong}$, i.e. having the same characteristic pair, we can introduce an acceptable indexing of the class of effective finitary posets. If (ϕ_i, ϕ_j) is the characteristic pair of an effective finitary poset E^ε, then we say that E^ε has an acceptable index $\langle i,j\rangle$ and denote it by $E^\varepsilon = \rho(\langle i,j\rangle)$. Now let $\overline{\rho(\langle i,j\rangle)}$ denote the completion of $\rho(\langle i,j\rangle)$. Then:

<u>Theorem 4.3.3</u> (<u>The Acceptable Indexing Theorem</u>)

$$\overline{\rho(\langle i,j\rangle)} = \bar{\rho}(\langle i,j\rangle).$$

Therefore by virtue of 4.3.3, we can say that $\rho(\langle i,j\rangle)$ is the effective basis of $\bar{\rho}(\langle i,j\rangle)$.

<u>4.4 Effectively Given SFP Objects as Effective Sequences</u>

Remember that SFP objects have a characterization in terms of ω-sequences of continuous projection pairs of finite cpo's. In this section, we will obtain an effective version of this.

Let $\langle D_m^\varepsilon m, (f_m, f_m^R)\rangle$ be an ω-sequence of computable projection pairs of effectively given SFP objects. If there is a recursive function $q: N \to N$ s.t $\pi_1 \cdot q(m)$ is a recursive index of f_m and $\pi_2 \cdot q(m)$ is an acceptable index of $D_m^\varepsilon m$, then we say that this sequence is <u>effective</u>. Obviously this sequence is effective iff there is a recursive function q' s.t. $\pi_1 \cdot \pi_1 \cdot q'(m)$ and $\pi_2 \cdot \pi_1 \cdot q'(m)$ are graph indices of f_m and f_m^R respectively and $\pi_2 \cdot q'(m)$ is an acceptable index of $D_m^\varepsilon m$.

If an effectively given SFP object is a finite cpo, then we call it an <u>effectively given finite cpo</u>.

<u>Theorem 4.4.1</u>

An indexed SFP object D^{ε} is an effectively given SFP object iff there is an effective sequence $\langle D^{\varepsilon}_{m}m, (f_m, f^R_m)\rangle$ of computable projection pairs of finite cpo's s.t.:

$$(D\ ,\varepsilon) \overset{e}{\cong} (D_{\infty}, \varepsilon_{\infty}).$$

<u>proof</u> (Sufficiency) By the same arguments as in 2.4.3, we can establish that ε_{∞} is an effective basis of D_{∞}. Thus $(D_{\infty}, \varepsilon_{\infty})$ is an effectively given SFP object.

(Necessity) Assume D^{ε} is an effectively given SFP object. We will construct an effective sequence of effectively given finite cpo's whose inverse limit is effectively isomorphic to D^{ε}. First notice that for every effective basis $\varepsilon:N\to E_D$, there is an effective basis $\varepsilon':N\to E_D$ s.t. $\varepsilon \overset{e}{\cong} \varepsilon'$ and $\varepsilon'(0)=\bot$. Therefore without loss of generality we can assume $\varepsilon(0)=\bot$. Let $\langle D_m, (f_m, f^R_m)\rangle$ be the Plotkin canonical sequence of D w.r.t. ε. We will introduce an effective indexing ε_m of D_m and show that ε_{∞} is an effective indexing s.t. $\varepsilon \overset{e}{\cong} \varepsilon_{\infty}$. Define ε_m by:

$\varepsilon_m(i)=$ <u>if</u> $i\le m$ <u>then</u> $\varepsilon(i)$ <u>else</u>
<u>if</u> $\{\varepsilon_m(0),\ldots,\varepsilon_m(i-1)\}=I\subset D_m$ <u>then</u>
$\varepsilon(\mu k.[\varepsilon(k)\in D_m \,\&\, \varepsilon(k)\notin I])$ <u>else</u>
<u>if</u> $I=D_m$ <u>then</u> $\varepsilon(0)$.

Notice that there is a recursive function p s.t. $\varepsilon_m(i)=\varepsilon(p(i,m))$. Evidently $D^{\varepsilon_m}_m$ is an effectively given finite cpo. Also $f_m:D_m\to D_{m+1}$ is an effective imbedding (and an effective embedding at the same time). Indeed $f_m\cdot\varepsilon_m=\varepsilon_{m+1}$. Effectiveness of the sequence $\langle D^{\varepsilon}_{m}, (f_m, f^R_m)\rangle$ is obvious. Therefore by the sufficiency, $(D_{\infty}, \varepsilon_{\infty})$ is an effectively given SFP object. Now we prove $D^{\varepsilon_{\infty}}_{\infty} \overset{e}{\cong} D^{\varepsilon}$.

Remember $E_{D_\infty} \cong E_D$ via Ω as in p.1.23. First we will show that there is a recursive function $r:N\to N$ s.t. $\Omega^R\cdot\varepsilon=\varepsilon_\infty\cdot r$. Notice that $\varepsilon(n)\in D_m$ is decidable thus there is a recursive function $h:N\to N$ s.t. $h(n)=\mu k.[\varepsilon(n)\in D_k]$. Also there is a recursive function $i:N\to N$ s.t. $i(n)=\mu k.[\varepsilon_{h(n)}(k)=\varepsilon(n)]$. Thus we have $\varepsilon_\infty(<i(n),h(n)>)=\Omega^R\cdot\varepsilon(n)$.

Thus $\lambda n.c(i(n),h(n))$ is such r.

Conversely we shall show the existence of a recursive function r' s.t. $\Omega\cdot\varepsilon_\infty=\varepsilon\cdot r'$. Remember that there are recursive functions $a,b:N\to N$ s.t. $\varepsilon_\infty(n)=f_{a(n)\infty}\cdot\varepsilon_{a(n)}(b(n))$. Define r' by:

$r'(n)=$ <u>if</u> $b(n)\le a(n)$ <u>then</u> $b(n)$ <u>else</u>

<u>if</u> $J=\{\varepsilon_{a(n)}(0),\ldots,\varepsilon_{a(n)}(n-1)\}\subset D_{a(n)}$ <u>then</u>

$\mu k.[\varepsilon(k)\in D_{a(n)}\ \&\ \varepsilon(k)\notin J]$ <u>else</u>

<u>if</u> $J=D_{a(n)}$ <u>then</u> 0.

Since $D_m=U^*_{E_D}(\{\varepsilon(0),\ldots,\varepsilon(m)\})$, r' is recursive. Evidently r' satisfies $\Omega\cdot\varepsilon_\infty=\varepsilon\cdot r'$. ☑

<u>Definition 4.4.2</u>

Given a finite cpo D, we say that an effective indexing $\varepsilon:N\to D$ is <u>normalized</u> iff ε is under the following constraint:

$\varepsilon(0)=\bot$

$\varepsilon(i)\ne\varepsilon(j)$ if $i\ne j$ & $i,j\le \mathrm{Card}(D)$

$\varepsilon(i)=\varepsilon(0)$ if $i>\mathrm{Card}(D)$.

If ε is normalized, we call D^ε a <u>normalized effectively given finite cpo</u>. □

Normalized effectively given finite cpo's enjoy interesting properties some of which will be listed below.

Lemma 4.4.3

(1) Given a normalized effectively given finte cpo D^{ε}, we can effectively obtain Card(D). More precisely there is a recursive function *Card* s.t. if m is an acceptable index of D^{ε}, then Card(D)=*Card*(m).

(2) Let D^{ε} and $D'^{\varepsilon'}$ be normalized effectively given finite cpo's s.t. $f:D\to D'$ is an effective embedding from ε to ε'. Then for every $x\in D'$, we can decide $x\in f(D)$.

(3) Let f be an effective imbedding from a normalized effectively given finite cpo D^{ε} to another $D'^{\varepsilon'}$, the adjoint $f^R:D'\to D$ maps effectively in directed indices , i.e. $f^R\cdot\varepsilon'(n)=\varepsilon\cdot p_{f^R}(n)$ for some recursive function p_{f^R}.

(4) There exists a recursive function *Apr* s.t. if k is a recursive index of an effective embedding $f:\bar{\rho}(i)\to\bar{\rho}(j)$ of effectively given finite cpo's then $p_{f^R}=\phi_{Apr(i,j,k)}$

proof (1) $\mathrm{Card}(D)=\mu k.[\varepsilon(k)=\varepsilon(k+1)]$.

(2) $\varepsilon'(n)\in f(D)$ iff $\varepsilon'(n)=\varepsilon\cdot r_f(m)$ for some $m\le \mathrm{Card}(D)$.

(3) $f^R(x')=\bigsqcup\{e\in D\,|\,f(e)\sqsubseteq x'\}$. Then:

$$\begin{aligned} f^R\cdot\varepsilon'(n) &= \bigsqcup\{\varepsilon(m)\,|\,\varepsilon'(r_f(m))\sqsubseteq\varepsilon'(n)\} \\ &= \bigsqcup\{\varepsilon(m)\,|\,\varepsilon'(r_f(m))\sqsubseteq\varepsilon'(n)\ \&\ m\le\mathrm{Card}(D)\} \\ &= \varepsilon\cdot p_{f^R}(n) \text{ for some recursive } p_{f^R}. \end{aligned}$$

(4) Immediate from the construction in the proof of (3). ▨

In the proof of 4.4.1, we constructed an effective sequence of computable projection pairs of effectively given finite cpo's, for every effectively given SFP object. We will present a normalized version of this. Given an effectively given SFP object D^{ε}, let $\langle D_m, (f_m, f_m^R)\rangle$ be the Plotkin canonical sequence w.r.t. ε. Define the following indexing $\hat{\varepsilon}_m$ for each D_m:

$$\hat{\varepsilon}_m(0)=\varepsilon(0)=\bot$$

$$\hat{\varepsilon}_m(i)= \underline{\text{if}}\ I=\{\hat{\varepsilon}_m(0),\ldots,\hat{\varepsilon}_m(i-1)\}\subset D_m\ \underline{\text{then}}$$

$$\varepsilon(\mu k.[\varepsilon(k)\in D_m\ \&\ \varepsilon(k)\notin I])\ \underline{\text{else}}$$

$$\underline{\text{if}}\ I=D_m\ \underline{\text{then}}\ \varepsilon(0).$$

Obviously $\hat{\varepsilon}_m$ is normalized and $\hat{\varepsilon}_\infty \overset{e}{=} \varepsilon \overset{e}{=} \varepsilon_\infty$. We will call the effective sequence $\langle D_m^{\hat{\varepsilon}_m}, (f_m, f_m^R)\rangle$ the <u>canonical effective sequence</u> of D^ε and call $\hat{\varepsilon}_\infty$ the <u>canonical effective basis</u> of D^ε.

<u>Corollary 4.4.4</u>

An indexed SFP object D^ε is an effectively given SFP object iff there is an effective sequence $\langle D_m^{\varepsilon_m}, (f_m, f_m^R)\rangle$ of computable projection pairs of effectively given normalized finite cpo's s.t. $D^\varepsilon \overset{e}{\cong} (D_\infty, \varepsilon_\infty)$.

With 4.4.1 and 4.4.4, we can now present an alternative characterization of computable elements in effectively given SFP objects.

<u>Theorem 4.4.5</u>

(1) Let $\langle D_m^{\varepsilon_m}, (f_m, f_m)\rangle$ be an effective sequence of computable projection pairs of finite cpo's. Then $x\in \text{Comp}(\lim\langle D_m^{\varepsilon_m}\rangle)$ iff there is a recursive function $c_x: N\to N$ s.t. $c_x(m)$ is a directed index of $f_{m\infty}^R(x)=x_m$.

(2) Let D^ε be an effectively given SFP object with the canonical effective sequence $\langle D_m^{\varepsilon_m}, (f_m, f_m^R)\rangle$, then $x\in \text{Comp}(D^\varepsilon)$ iff there is a recursive function $c_x: N\to N$ s.t. $c_x(m)$ is a directed index of $f_{m\infty}^R(x)=x_m$.

<u>proof</u> (1) By almost the same arguments as in the proof of 3.5.2.
(2) Immediate from (1). ∎

It is at least worthwhile to note that the equivalence in 4.4.5 is "effective".

4.5 Domain Constructors

Let D^{ε} and $D'^{\varepsilon'}$ be effectively given SFP objects. Define indexed SFP objects $D^{\varepsilon}\times D'^{\varepsilon'}$, and $D^{\varepsilon}+D'^{\varepsilon'}$ similarly to 2.1.3. We will observe that they are effectively given.

Lemma 4.5.1

Let D and D' be cpo's.

(1) For every $X\subseteq D\times D'$, $(x,y)\in U_{D\times D'}(X)$ iff $x\in U_D(\pi_1(X))$ & $y\in U_{D'}(\pi_2(X))$.

(2) For every $X\subseteq D+D'$:

$(0,u)\in U_{D+D'}(X)$ iff $r_1(X)=\{0\}$ and $u\in U_D(r_2(X))$

$(1,u')\in U_{D+D'}(X)$ iff $r_1(X)=\{1\}$ and $u'\in U_D(r_2(X))$

where $r_i((a,b))=\pi_i((a,b))$ and $r_i(\{\bot\})=\emptyset$.

Theorem 4.5.2

Let D^{ε} and $D'^{\varepsilon'}$ be effectively given SFP objects, then so are $D^{\varepsilon}\times D'^{\varepsilon'}$ and $D^{\varepsilon}+D'^{\varepsilon'}$.

proof It is sufficient to show that $\varepsilon\times\varepsilon'$ and $\varepsilon+\varepsilon'$ are effective indexings. First we have:

$$U_{E_D\times E_{D'}}(\{\varepsilon\times\varepsilon'(x_1),\ldots,\varepsilon\times\varepsilon'(x_n)\})$$

$$=U_{E_D\times E_{D'}}(\{(\varepsilon\cdot\pi_1(x_1),\ \varepsilon'\cdot\pi_2(x_1)),\ldots,(\varepsilon\cdot\pi_1(x_n),\varepsilon'\cdot\pi_2(x_n))\})$$

$$=U_{E_D}(\{\varepsilon\cdot\pi_1(x_1),\ldots,\varepsilon\cdot\pi_1(x_n)\})\times U_{E_{D'}}(\{\varepsilon'\cdot\pi_2(x_1),\ldots,\varepsilon'\cdot\pi_2(x_n)\}).$$

The last line is due to 4.5.1. Therefore $\varepsilon\times\varepsilon'$ is an effective indexing. For $\varepsilon+\varepsilon'$, we have three cases to be considered.

(Case 1) $X=\{\varepsilon+\varepsilon'(x_1),\ldots,\varepsilon+\varepsilon'(x_n)\}$ contains no element from E_D and $E_{D'}$. In this case $U_{E_D+E_{D'}}(X)=\{\bot\}$.

(Case 2) X contains no element from $E_{D'}$. In this case $X=\{(0,\varepsilon(k_1)),\ldots,(0,\varepsilon(k_n))\}$ where $x_j=2k_j+1$ $(1\le j\le n)$. Thus by 4.5.1 we have:

$$U_{E_D+E_{D'}}(X)=\{(0,u)\mid u\in U_{E_D}(\varepsilon(k_1),\ldots,\varepsilon(k_n)\}.$$

(Case 3) $X=\{\varepsilon+\varepsilon'(x_1),\ldots,\ \varepsilon+\varepsilon'(x_n)\}=\{(1,\varepsilon'(k_1),\ldots,(1,\varepsilon'(k_n))\}$ where $x_j=2k_j$ $(1\le j\le n)$. By 4.5.1 we have:

$$U_{E_D+E_{D'}}(X)=\{(1,u')\mid u'\in U_{E_{D'}}(\varepsilon'(k_1),\ldots,\varepsilon'(k_n))\}.$$

Under any of these cases, $\mathrm{Card}(U_{E_D+E_{D'}}(X))=m$ is recursive in $x_1,\ldots,x_n$ and m. Also $\varepsilon+\varepsilon'(k)\in U_{E_D+E_{D'}}(X)$ is recursive in $x_1,\ldots,x_n$, and k. Thus $\varepsilon+\varepsilon'$ is an effective indexing. ◻

Notice that in case D^{ε} and $D'^{\varepsilon'}$ are effectively given finite cpo's then so are $D^{\varepsilon}+D'^{\varepsilon'}$ and $D^{\varepsilon}\times D'^{\varepsilon'}$.

Let $\langle D_m^{\varepsilon_m},(f_m,f_m^R)\rangle$ and $\langle D_m'^{\varepsilon_m'},(f_m',f_m'^R)\rangle$ be the canonical sequences of D^{ε} and $D'^{\varepsilon'}$ respectively. It is quite straightforward to observe that $\langle D_m^{\varepsilon_m}\times D_m'^{\varepsilon_m'},(f_m\times f_m',f_m^R\times f_m'^R)\rangle$ and $\langle D_m^{\varepsilon_m}+D_m'^{\varepsilon_m'},(f_m+f_m',f_m^R+f_m'^R)\rangle$ are effective sequences of computable projection pairs of effectively given finite cpo's. Indeed we have:

$$D^{\varepsilon}\times D'^{\varepsilon'}\overset{e}{\cong}\underleftarrow{\lim}\langle D_m^{\varepsilon_m}\times D_m'^{\varepsilon_m'},(f_m\times f_m',f_m^R\times f_m'^R)\rangle \text{ and}$$

$$D^{\varepsilon}+D'^{\varepsilon'}\overset{e}{\cong}\underleftarrow{\lim}\langle D_m^{\varepsilon_m}+D_m'^{\varepsilon_m'},(f_m+f_m',f_m^R+f_m'^R)\rangle.$$

Therefore we can provide an alternative proof to 4.5.2. But since the proof of 4.5.2 is very simple, there is very little point in showing details of this alternative proof.

Given effectively given SFP objects D^{ε} and $D'^{\varepsilon'}$, define an indexed SFP objects $[D^{\varepsilon}\to D'^{\varepsilon'}]$ by:

$$[D^{\varepsilon}\to D'^{\varepsilon'}]=[D\to D']^{[\varepsilon\to\varepsilon']}$$

where $[\varepsilon\to\varepsilon']$ is as in 2.1.3.

In contrast to $\times$ and +, the direct attempt to establish effectiveness of $[\varepsilon\to\varepsilon']$ is not easy at all. Remember that this task was not so easy even for the bounded complete case (See Egli and Constable [1] and Rosen and Markowsky [15]).

However it is fairly easy to observe that if D^{ε} and $D'^{\varepsilon'}$ are "normalized" effectively given finite cpo's, then $[D^{\varepsilon}\to D'^{\varepsilon'}]$ is an effectively given finite cpo. Thence we can establish the closure under $\to$ of the class of effectively given SFP objects using 4.4.1.

Lemma 4.5.3

Let D^{ε} and $D'^{\varepsilon'}$ be normalized effectively given finite cpo's. Then $[D^{\varepsilon}\to D'^{\varepsilon'}]$ is an effectively given finite cpo.

proof It is sufficient to show that $[\varepsilon\to\varepsilon']$ is effective. Because D^{ε} and $D'^{\varepsilon'}$ are normalized, we can effectively obtain $\mathrm{Card}(D^{\varepsilon})$ and $\mathrm{Card}(D'^{\varepsilon'})$. Furthermore we have:

$$D=\{\varepsilon(i)\mid i\leq \mathrm{Card}(D^{\varepsilon})\}$$

$$D'=\{\varepsilon'(i)\mid i\leq \mathrm{Card}(D'^{\varepsilon'})\}.$$

Define a predicate *func* by:

$$func(n)= \underline{\text{if}}\ P_r(n)\not\subseteq \mathrm{Card}(D^{\varepsilon})\times \mathrm{Card}(D'^{\varepsilon'})\ \underline{\text{then}}\ 0\ \underline{\text{else}}$$
$$\underline{\text{if}}\ \{(\varepsilon(i),\varepsilon'(j))\mid \langle i,j\rangle\in P_r(n)\}\in[D\to D']\ \underline{\text{then}}\ 1$$
$$\underline{\text{else}}\ 0.$$

Evidently *func* is a recursive predicate. Now we introduce a partial indexing γ of $[D\to D']$ by: for every $f\in[D\to D']$,

$$f=\gamma(k)\ \text{if}\ f=\{(\varepsilon(i),\varepsilon(j))\mid \langle i,j\rangle\in P_r(n)\}$$

where n is the kth integer satisfying *func*(n). It is straightforward to observe that $\gamma(i)\sqsubseteq\gamma(j)$ is recursive in i and j. Since we know an upper bound of the cardinal of $[D\to D']$, and there is an effective way of obtaining k from $\langle i,j\rangle$ s.t. $[\varepsilon(i),\varepsilon(j)]=\gamma(k)$; $[\varepsilon\to\varepsilon']$ is an effective indexing. ∅

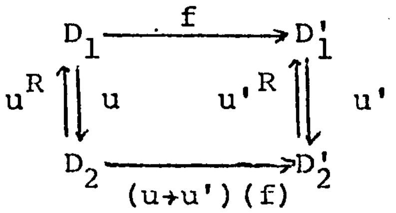
Lemma 4.5.4

Let $D_1^{\varepsilon_1},D_2^{\varepsilon_2},D_1'^{\varepsilon_1'},D_2'^{\varepsilon_2'}$ be normalized effectively given finite cpo's. Furthermore let $u:D_1\to D_2$ and $u':D_1'\to D_2'$ be effective embeddings from ε_1 to ε_2 and ε_1' to ε_2' respectively.

Then $(u \to u'):[D_1 \to D_1'] \to [D_2 \to D_2']$ defined by:

$$(u \to u')(f) = u' \cdot f \cdot u^R$$

is an effective imbedding from $[\varepsilon_1 \to \varepsilon_1']$ to $[\varepsilon_2 \to \varepsilon'_2]$.



<u>proof</u> It is sufficient to show that there is a recursive function $r_{(u \to u')}$ s.t. $(u \to u') \cdot [\varepsilon_1 \to \varepsilon_1'] = [\varepsilon_2 \to \varepsilon_2'] \cdot r_{(u \to u')}$. Given $n \in N$, we can effectively obtain $i_1, \ldots, i_{m_n}$ s.t. $\gamma_1(i_1) = [\varepsilon_1(k_1), \varepsilon_1'(k_1')], \ldots, \gamma_1(i_{m_n}) = [\varepsilon_1(k_{m_n}), \varepsilon_1'(i_{m_n})]$, where $\{\langle k_1, k_1' \rangle, \ldots, \langle k_{m_n}, k_{m_n}' \rangle\} \in P_r(n)$. Therefore, from $i_1, \ldots, i_{m_n}$, we can effectively obtain $\eta_1(n) \in N$ s.t. $[\varepsilon_1 \to \varepsilon_1'](n) = \gamma_1(\eta_1(n))$. By the effectiveness of the construction, we can regard η_1 as a recursive function. Therefore we have:

$$(u \to u')([\varepsilon_1 \to \varepsilon_1'](n)) = u' \cdot \gamma_1(\eta_1(n)) \cdot u^R.$$

Remember that there are recursive functions $r_{u'}$ and p_{u^r} s.t. $u' \cdot \varepsilon_1'(m) = \varepsilon_2' \cdot r_{u'}(m)$ and $u^R \cdot \varepsilon_2(m) = \varepsilon_1 \cdot p_{u^R}(m)$. Therefore there is a recursive function $w: N \to N$ s.t.

$$(u \to u')([\varepsilon_1 \to \varepsilon_1'](n)) = u' \cdot \gamma_1(\eta_1(n)) \cdot u^R$$
$$= \gamma_2(w(n)).$$

However we have a recursive function $\eta_2: N \to N$ s.t.

$$[\varepsilon_2 \to \varepsilon_2'](n) = \gamma_2(\eta_2(n)).$$

Now take $r_{(u \to u')}(n) = \mu k.[\gamma_2(\eta_2(k)) = \gamma_2(w(n))]$. ☐

Notice that in the above proof, we have constructed a recursive function w from $r_{u'}$ and p_{u^R}. Thence we have constructed $r_{(u \to u')}$ from w. Furthermore p_{u^R} was constructed from r_u via 4.4.3. Therefore we have constructed $r_{(u \to u')}$ from

r_u and $r_{u'}$. More precisely there is a recursive function fn s.t. if i,j,i',j' are acceptable indicies of $D_1^{\varepsilon}1, D_2^{\varepsilon}2, D_1'^{\varepsilon'}1, D_2'^{\varepsilon'}2$, and k, m are recursive indicies of u and u' respectively then $fn(i,j,k,m)$ is a recursive index of $(u \to u')$.

Furthermore it should be noticed that due to the effectiveness of the construction for $\to$, we can effectively obtain an acceptable index of $[D^{\varepsilon} \to D'^{\varepsilon'}]$ from the acceptable indices of normalized effectively given finite cpo's D^{ε} and $D'^{\varepsilon'}$.

These observations establish the following theorem:

Theorem 4.5.5 (The Function Space Theorem I)

Let $\langle D_m^{\varepsilon}m, (f_m, f_m^R)\rangle$ and $\langle D_m'^{\varepsilon'}m, (f_m', f_m'^R)\rangle$ be the canonical effective sequences of D^{ε} and $D'^{\varepsilon'}$ respectively. Then $\langle [D_m^{\varepsilon}m \to D_m'^{\varepsilon'}m], ((f_m \to f_m'), (f_m \to f_m')^R)\rangle$ is an effective sequence of computable projection pairs of effectively given finite cpo's. Then $\varprojlim \langle [D_m^{\varepsilon}m \to D_m'^{\varepsilon'}m], ((f_m \to f_m'), (f_m \to f_m')^R)\rangle$ is an effectively given SFP object.

The next theorem ensures that we get the right function space.

Theorem 4.5.6

Let D^{ε} and $D'^{\varepsilon'}$ be effectively given SFP objects with the canonical effective sequences $\langle D_m^{\varepsilon}m, (f_m, f_m^R)\rangle$ and $\langle D_m'^{\varepsilon'}m, (f_m', f_m'^R)\rangle$ respectively. Let $\Omega: E_{D_\infty} \to E_D$ and $\Omega': E_{D_\infty'} \to E_{D'}$ be effective isomorphisms. Then $g \in [D \to D']$ is computable wrt $(\varepsilon, \varepsilon')$ iff $\Phi(\hat{g}) \in$ Comp$(\varprojlim \langle [D_m^{\varepsilon}m \to D_m'^{\varepsilon'}m], ((f_m \to f_m'), (f_m \to f_m')^R)\rangle)$ where $\hat{g} = \bar{\Omega}'^R \cdot g \cdot \bar{\Omega}$ and Φ is as in the proof of 1.3.8.

Proof g is computable wrt $(\varepsilon, \varepsilon')$ iff $\hat{g}$ is computable wrt $(\varepsilon_\infty, \varepsilon_\infty')$. Now let $\langle \Gamma_m^{\gamma}m, (h_m, h_m^R)\rangle$

be the cannonical effective sequence of $\underleftarrow{\lim}$<[$D_m^{\varepsilon_m} \to D_m'^{\varepsilon_m'}$], $((f_m \to f_m'),(f_m \to f_m')^R)$>. Assume there is a recursive function c_g s.t. $\hat{g}_m = \zeta_{\gamma_m} \cdot c_g = h^R_{m\infty}(\hat{g})$ where $g \in \Gamma_\infty$. We will show that $\Phi^R(\hat{g}) = \lambda x. \bigsqcup_m f'_{m\infty} \cdot \hat{g}_m(x_m)$ is computable w.r.t. $(\varepsilon_\infty, \varepsilon'_\infty)$. In fact $\Phi^R(\hat{g})(\varepsilon_\infty(m)) = \bigsqcup_n f'_{n\infty} \cdot \hat{g}_n(f^R_{n\infty} \cdot \varepsilon_\infty(m))$. Thus $\varepsilon'_\infty(n) \sqsubseteq \Phi^R(\hat{g})(\varepsilon_\infty(m))$ iff $\varepsilon'_\infty(n) \sqsubseteq f'_{k\infty} \cdot \hat{g}_k(f^R_{k\infty} \cdot \varepsilon_\infty(m))$ for some $k \in N$. Before we prove further we will prove the following lemma:

Lemma 4.5.7

There is a recursive function v_m s.t. $f^R_{m\infty} \cdot \varepsilon_\infty = \varepsilon_m \cdot v_m$ whenever $\langle D_m^{\varepsilon_m}, (f_m, f_m^R)\rangle$ is the canonical sequence of D^ε.

proof There are recursive functions a, b s.t. $\varepsilon_\infty(n) = f_{a(n)\infty} \cdot \varepsilon_{a(n)}(b(n))$. Therefore we have:

$$f^R_{m\infty} \cdot \varepsilon_\infty(n) = \underline{\text{if}}\ m < a(n)\ \underline{\text{then}}\ f^R_m \cdot \ldots \cdot f^R_{a(n)-1}(\varepsilon_{a(n)} \cdot b(n))$$
$$\underline{\text{else}}\ \underline{\text{if}}\ m = a(n)\ \underline{\text{then}}\ \varepsilon_{a(n)}(b(n))$$
$$\underline{\text{else}}\ f_{m-1} \cdot \ldots \cdot f_{a(n)}(\varepsilon_{a(n)}(b(n))).$$

But since f_m is an effective embedding and 4.4.3 holds we have:

$$f^R_{m\infty} \cdot \varepsilon_\infty(n) = \underline{\text{if}}\ m < a(n)\ \underline{\text{then}}\ \varepsilon_m \cdot p_{f^R} \cdot \ldots \cdot p_{f^R}(b(n))$$
$$\underline{\text{else}}\ \underline{\text{if}}\ m = a(n)\ \underline{\text{then}}\ \varepsilon_m(b(n))$$
$$\underline{\text{else}}\ \varepsilon_m \cdot r_{f_{m-1}} \cdot \ldots \cdot r_{f_{a(n)}}(b(n)).$$

Remember that since $\langle D_m^{\varepsilon_m}, (f_m, f_m^R)\rangle$ is an effe ctive sequence, there is a recursive function q s.t. $\pi_1 \cdot q(m)$ is a recursive index of f_m and $\pi_2 \cdot q(m)$ is an acceptable index of $D_m^{\varepsilon_m}$. Let Apr be as in 4.4.3, then $q': N \to N$ defined by:

$$q'(m) = Apr(\pi_2 \cdot q(m), \pi_2 \cdot q(m+1), \pi_1 \cdot q(m))$$

is recursive and $q'(m)$ is a recursive index of p_{f^R}.

Define v_m by:

$$v_m(n) = \underline{\text{if}}\ a(n) > m\ \underline{\text{then}}\ \phi_{cp'(m,n)}(b(n))$$

$\underline{\text{else}}$ $\underline{\text{if}}$ $a(n)=m$ $\underline{\text{then}}$ $b(n)$

$\underline{\text{else}}$ $\phi_{cp(m,n)}(b(n))$

where $cp'(m,n)=A\text{-}cp(q'(m),A\text{-}cp(\ldots,A\text{-}cp(q'(a(n)),q'(a(n)-1))\ldots))$
and $cp(m,n)=A\text{-}cp(\pi_1\cdot q(m-1),A\text{-}cp(\ldots,(A\text{-}cp(\pi_1\cdot q(a(n)+1),$
$\pi_1\cdot q(a(n)))\ldots))$,
where $A\text{-}cp(i,j)$ is an acceptable index of $\phi_i\cdot\phi_j$. Evidently v_m is recursive and $f^R_{m\infty}\cdot\varepsilon_\infty=\varepsilon_m\cdot v_m$. ☒

Now we resume the proof of 4.5.6. There is a recursive function l s.t. $\varepsilon_k(l(k,m))=f^R_{k\infty}\cdot\varepsilon_\infty(m)$, indeed $l(k,m)=v_k(m)$. Also there is a recursive function z s.t. $f'_{k\infty}\cdot\varepsilon'_k(n)=\varepsilon'_\infty(z(k,n))$, because of the definition of ε'_∞. Remember $\hat{g}_m=\zeta_{\gamma_m}\cdot c_g(m)$. Thus:

$$f'_{k\infty}\cdot g_k\cdot f^R_{k\infty}\cdot\varepsilon_\infty(m)=f'_{k\infty}(\bigsqcup\{(\gamma_k(i))(f^R_{k\infty}\cdot\varepsilon_\infty(m))\mid i\in W_{d_{\gamma_m}}(c_x(m))\})$$
$$=\bigsqcup\{f'_{k\infty}\cdot(\gamma_k(i))(f^R_{k\infty}\cdot\varepsilon_\infty(m))\mid i\in W_{d_{\gamma_m}}(c_x(m))\}.$$

Thus $\varepsilon'_\infty(n)\sqsubseteq f'_{k\infty}\cdot\hat{g}_k\cdot f^R_{k\infty}\cdot\varepsilon_\infty(m)$ iff
$\varepsilon'_\infty(n)\sqsubseteq f'_{k\infty}\cdot\gamma_k(i)\cdot f^R_{k\infty}\cdot\varepsilon_\infty(m)$ for some $i\in W_{d_{\gamma_m}}(c_x(m))$.

But $\gamma_k(i)(\varepsilon_k(j))=\varepsilon'_k(y(k,i,j))$ for some recursive function y. Therefore there is a recursive function x s.t.

$$f'_{k\infty}\cdot\gamma_k(i)\cdot f^R_k\cdot\varepsilon_\infty(m)=\varepsilon'(x(k,m)).$$

Thus $\varepsilon'_\infty(n)\sqsubseteq f'_{k\infty}\cdot\hat{g}_k\cdot f^R_{k\infty}\cdot\varepsilon_\infty(m)$ is r.e. in n,m,k.
Thus $\varepsilon'_\infty(n)\sqsubseteq\Phi^R(\hat{g})(\varepsilon_\infty(m))$ is r.e. in n and m.
Thus $\Phi^R(\hat{g})$ is computable w.r.t. $(\varepsilon_\infty,\varepsilon'_\infty)$.
Conversely let g be computable w.r.t. $(\varepsilon_\infty,\varepsilon'_\infty)$. We will show that $\Phi(\hat{g})\in\text{Comp}(\underleftarrow{\lim}<[D^{\varepsilon_m}_m\to D'^{\varepsilon'_m}_m],((f_m\to f'_m),(f_m\to f'_m)^R)>)$. Notice that $\Phi(\hat{g})=<\hat{g}_{(m)}>$ (see 1.3.3). By virtue of 5.4.5 it is sufficient to show that we can effectively obtain a procedure which enumerates

the graph of $\hat{g}_{(m)}$ for every m. Since $f'_{m\infty}$ is an embedding we have:

$$\varepsilon'_m(k) \sqsubseteq \hat{g}_{(m)} \cdot \varepsilon_m(n) = f'^R_{m\infty} \cdot \hat{g} \cdot f_{m\infty} \cdot \varepsilon_m(n)$$

$$\text{iff} \quad f'_{m\infty} \cdot \varepsilon'_m(k) \sqsubseteq f'_{m\infty} \cdot f'^R_{m\infty} \cdot \hat{g} \cdot f_{m\infty} \cdot \varepsilon_m(n).$$

Therefore we have the following procedure which enumerates the graph of $g_{(m)}$:

--- enumerate N: 0,1,2,.....

--- for each n enumerated, enumerate k s.t.:

$$\varepsilon'_\infty(k) \sqsubseteq f'_{m\infty} \cdot f'^R_{m\infty} \cdot \hat{g} \cdot f_{m\infty} \cdot \varepsilon_m(n) = f'_{m\infty} \cdot f'^R_{m\infty} \cdot \hat{g} \cdot \varepsilon_\infty(\gamma_{f_{m\infty}}(n)).$$

Since $f'_{m\infty} \cdot f'^R_{m\infty} \cdot \hat{g}$ is computable we can recursively enumerate such k.

--- for each k enumerated, if $a(k)=m$ then output $(n, b(k))$ where k is enumerated for n.

Notice that the above procedure is constructed uniformly in m. Thus we have established the theorem. ∅

It must be at least mentioned that the equivalence established above is "effective". Indeed we can effectively go back and forth among graph indicies of g's and the directed indices of $\Phi(\hat{g})$'s.

<u>Theorem 4.5.8</u> (<u>The Function Space Theorem II</u>)

(1) Let D^ε and $D'^{\varepsilon'}$ be effectively given SFP objects with the canonical sequences $\langle D_m^{\varepsilon_m}, (f_m, f_m^R)\rangle$ and $\langle D'^{\varepsilon'_m}_m, (f'_m, f'^R_m)\rangle$ respectively. Then $[D^\varepsilon \to D'^{\varepsilon'}] \overset{e}{\cong} \varprojlim \langle [D_m^{\varepsilon_m} \to D'^{\varepsilon'_m}_m], ((f_m \to f'_m), (f_m \to f'_m)^R)\rangle$. Thence $[D^\varepsilon \to D'^{\varepsilon'}]$ is an effectively given SFP object.

(2) $g \in \mathrm{Comp}([D^\varepsilon \to D'^{\varepsilon'}])$ iff g is computable w.r.t. $(\varepsilon, \varepsilon')$.

<u>proof</u> (1) Φ is an effective isomorphism.

(2) By 4.5.7. ∅

Theorem 4.5.9

There are recursive functions *s-Prod*, *s-Sum*, *s-Func* s.t.

(1) $\overline{\rho}(i)\times\overline{\rho}(j) = \overline{\rho}(s\text{-}Prod(i,j))$

(2) $\overline{\rho}(i)+\overline{\rho}(j) = \overline{\rho}(s\text{-}Sum(i,j))$

(3) $[\overline{\rho}(i)\to\overline{\rho}(j)] = \overline{\rho}(s\text{-}Func(i,j))$ ∎

The following relations can immediately be observed:

(1) $Conv(Prod(i,j))=s\text{-}Prod(Conv(i),Conv(j))$

(2) $Conv(Sum(i,j))=s\text{-}Sum(Conv(i),Conv(j))$

(3) $Conv(Func(i,j))=s\text{-}Func(Conv(i),Conv(j))$,

where i and j are acceptable indicies of effectively given SFP objects.

Now we have function spaces and thus have a directed indexing for each $Comp([D^{\varepsilon}\to D'^{\varepsilon'}])$. It can readily be seen that we can effectively go back and forth between directed indexing and graph indexing of computable functions. This allows us to have similar results to 2.2.7 for effectively given SFP objects in place of 4.2.3. Also it can readily be seen that results similar to 2.5.1 and 2.5.2 hold for effectively given SFP objects.

4.6 Inverse Limits

Effective sequence of computable projection pairs of effectively given SFP objects can be defined as in 2.4.1. Since we can effectively go back and forth among recursive indexing and directed indexing, we can obtain an alternative characterization of effective sequences in terms of directed indices of effective embeddings as in 2.4.2.

By almost the same arguments as in 2.4.3, we have:

Theorem 4.6.1 (The Inverse Limit Theorem)

Let $<D_m^{\varepsilon_m},(f_m,f_m^R)>$ be an effective sequence of computable projection pairs of effectively given SFP objects. Then:

(1) $\underleftarrow{\lim}<D_m^{\varepsilon_m},(f_m,f_m^R)>=(D_\infty,\varepsilon_\infty)$ is an effectively given SFP object.

(2) $f_{m\infty}$ is an effective embedding from ε_m to ε_∞.

(3) There are recursive functions λ_d and δ_d s.t. $\lambda_d(m)$ and $\delta_d(m)$ are directed indicies of $f_{m\infty}$ and $f_{m\infty}^R$ respectively. ☒

We have theorems like 2.4.4 and 2.4.7, for effectively given SFP objects. Furthermore we can define effective isomorphisms of effective sequences of computable projection pairs of effectively given SFP objects and show the invariance of effective isomorphisms under limit construction.

4.7 The Power Domain Construction

In this section, we study that the power domain of an effectively given SFP object again is an effectively given SFP object.

Definition 4.7.1

Given an indexed SFP object (D,ε), define $\mathcal{F}(\varepsilon):N\to M(D)$ and $\mathcal{F}[\varepsilon]:N\to M[D]$ by:

$$\mathcal{F}(\varepsilon)(n)=\varepsilon(f_s(n))$$

$$\mathcal{F}[\varepsilon](n)=[\mathcal{F}(\varepsilon)(n)]$$

where $M(D)$ and $M[D]$ are as in 1.3 between 1.3.15 and 1.3.17. $(\mathcal{F}[D],\mathcal{F}[\varepsilon])$ is called the (strong) power domain of (D,ε). We write $\mathcal{F}[D^\varepsilon]$ to denote $(\mathcal{F}[D],\mathcal{F}[\varepsilon])$. □

Lemma 4.7.2

Let (D,ε) be a normalized effectively given finite cpo. Then $\mathcal{F}[D^{\varepsilon}]$ is an effectively given finite cpo. In fact there is a recursive function *f-Power* s.t. if n is an acceptable index of (D,ε) then *f-Power*(n) is an acceptable index of $\mathcal{F}[D^{\varepsilon}]$.

proof It can readily be seen that there is a recursive function *Ml* s.t. if n is an acceptable index of an effectively given SFP object then $\phi_{Ml(n)}$ is a recursive predicate satisfying:

$$\phi_{Ml(n)}(x,y)=0 \iff \varepsilon(f_s(x))\sqsubseteq_M\varepsilon(f_s(y)).$$

Notice that $(M(D),\sqsubseteq_M)$ is a pre-ordering and so $[x]\sqsubseteq_M/\equiv[y]$ iff $x\sqsubseteq_M y$ where $\equiv$ is the canonical equivalence obtained from $\sqsubseteq_M$. Thus we have:

$$[\varepsilon(f_s(a))]\sqsubseteq_M/\equiv[\varepsilon(f_s(b))]$$
$$\iff \varepsilon(f_s(a))\sqsubseteq_M\varepsilon(f_s(b))$$
$$\iff \phi_{Ml(n)}(a,b)=0.$$

Since (D,ε) is an effectively given "normalized" finite cpo, there is a recursive function C_{pw} s.t. $C_{pw}(n)=\text{Card}(\mathcal{F}[D])$. Now we can decide $\sqsubseteq_M/\equiv$ via $\phi_{Ml(n)}$ and we can check if we exhausted the whole elements of $\mathcal{F}[D]$, for $\sqsubseteq_M/\equiv$ check, via $C_{pw}(n)$. Thus there are recursive functions h and k s.t. $\phi_{h(n)}$ and $\phi_{k(n)}$ are recursive predicates satisfying:

$$\phi_{h(n)}(x,y)=0 \iff \text{Card}(\mathcal{U}_{M[D]}(\mathcal{F}[\varepsilon](f_s(x))))=y$$
$$\phi_{k(n)}(x,y)=0 \iff \mathcal{F}[\varepsilon](x)\in\mathcal{U}_{M[D]}(\mathcal{F}[\varepsilon](f_s(y))).$$

Let *f-Power* be a recursive function given by:

f-Power(n)=$\langle h(n),k(n)\rangle$. ∎

Lemma 4.7.3

There is a recursive function P_{pr} s.t. if (D,ε) and (D',ε') are normalized effectively given finite cpo's and $(p:D\to D',p^R:D'\to D)$

is a computable projection pair with a directed index x then $([p],[p^R]):[D]\to[D']$ is a computable projection pair with a directed index $P_{pr}(x)$.

<u>proof</u> Let x be a directed index of (p,p^R). By the remark after 4.5.9, p has a recursive index $R_d(x)$ for some recursive function R_d. Thus $\phi_{R_d(x)}$ is a recursive function s.t. $p\cdot\varepsilon=\varepsilon'\cdot\phi_{R_d(x)}$. Remember $[p]([x])=[p(x)]$. Thus we have:

$$\begin{aligned}[p](F[\varepsilon](n)) &= [p]([\varepsilon(f_s(n))]) \\ &= [p\cdot\varepsilon(f_s(n))] \\ &= [\varepsilon'\phi_{R_d(x)}(f_s(n))] \\ &= F[\varepsilon'](\phi_{g(x)}(n))\end{aligned}$$

where g is a recursive function s.t. $\phi_{g(x)}$ is a recursive function and $f_s(\phi_{g(x)}(n))=\phi_{R_d(x)}(f_s(n))$. Thus [p] has a recursive index $g(x)$. By the remark after 4.5.9, we have established the lemma. ∎

<u>Theorem 4.7.4</u>

Let (D,ε) be an effectively given SFP object and $\langle D_m^{\varepsilon}m,(f_m,f_m^R)\rangle$ be the canonical effective sequence of (D,ε). Then $\langle F[D_m^{\varepsilon}m],([f_m],[f_m^R])\rangle$ is an effective sequence of computable projection pairs of effectively given finite cpo's. Thus $\underleftarrow{\lim}\langle F[D_m^{\varepsilon}m],([f_m],[f_m^R])\rangle$ is an effectively given SFP object. Furthermore:

$$F[D^{\varepsilon}] \overset{e}{\cong} \underleftarrow{\lim}\langle F[D_m^{\varepsilon}m],([f_m],[f_m^R])\rangle.$$

Also there is a recursive function *Power* s.t. if n is an acceptable index of an effectively given SFP object then *Power*(n) is an acceptable index of the power domain of it.

<u>proof</u> By 4.7.2 and 4.7.3, $\langle F[D_m^{\varepsilon}m],([f_m]\ [f_m^R])\rangle$ is an effective sequence of computable projection pairs of effectively given finite cpo's. Thus by 4.4.1, $\underleftarrow{\lim}\langle F[D_m^{\varepsilon}m],([f_m],[f_m^R])\rangle$ is an

effectively given SFP object. The basis of $\mathcal{F}[\varprojlim\langle D^{\varepsilon_m}_m, (f_m, f^R_m)\rangle]$ is $B=[M(D_\infty)]=[\bigcup_m f_{m\infty}(M(D_m))]$ while the basis of $\varprojlim\langle\mathcal{F}[D_m], ([f_m],[f^R_m])\rangle$ is $B'=\bigcup_m[f_{m\infty}][M(D_m)]$. It can readily be seen that $h:B\to B'$ defined by:

$$h([f_{m\infty}(F_m)])=[f_{m\infty}][F_m]$$

where $F_m\subseteq M(D_m)$, is an isomorphism. Since we have:

$$\begin{aligned}\mathcal{F}[\varepsilon_\infty](f_s(x))&=[\varepsilon_\infty(f_s(x))]\\&=[f_{m\infty}(\varepsilon_m(f_s(x)))]\end{aligned}$$

where $m=\mu k.[\varepsilon_k(f_s(x))\sqsubseteq D_k]$, we can readily see that h is an effective isomorphism. Thus we have:

$$\begin{aligned}\mathcal{F}[D^\varepsilon]&\overset{e}{\cong}\mathcal{F}[\varprojlim\langle D^{\varepsilon_m}_m, (f_m, f^R_m)\rangle]\\&\overset{e}{\cong}\varprojlim\langle\mathcal{F}[D^{\varepsilon_m}_m], ([f_m],[f^R_m])\rangle.\end{aligned}$$

Notice that we can effectively obtain a sequence index of $\langle\mathcal{F}[D^{\varepsilon_m}_m], ([f_m],[f^R_m])\rangle$ from an acceptable index n of (D,ε). Therefore by the remarks after 4.5.9 and 4.6.1, we have a recursive function *Power* s.t. *Power*(n) is an acceptable index of $\mathcal{F}[D^\varepsilon]$. ∎

Theorem 4.7.5

Given two effectively given SFP objects (D,ε) and (D',ε'), for every computable function $f:D\to D'$, let $\tilde{f}:\mathcal{F}(D^\varepsilon)\to\mathcal{F}(D'^{\varepsilon'})$ be defined by $\tilde{f}(X)=f(X)=\{f(x)\mid x\in X\}$. Then $\hat{f}=[\tilde{f}]$ is a computable function. Indeed there is a recursive function *Ext* s.t. if n is a directed index of a computable function $f:D^\varepsilon\to D'^{\varepsilon'}$ and i and j are acceptable indices of D^ε and $D'^{\varepsilon'}$ respectively, then *Ext*(i,j,n) is a directed index of $\hat{f}$.

proof We have:

$$F[\varepsilon'](x) \sqsubseteq_{M'} /\!\equiv f(F[\varepsilon](y))$$

$$\Longleftrightarrow [\varepsilon'(f_s(x))] \sqsubseteq_{M'} /\!\equiv [f(\varepsilon(f_s(y)))]$$

$$\Longleftrightarrow \varepsilon'(f_s(x)) \sqsubseteq_M f(\varepsilon(f_s(y)))$$

$$\Longleftrightarrow \forall z \in N.[\varepsilon'(f_s(z)) \sqsubseteq_{M'} \varepsilon'(f_s(x)) \text{ implies } \varepsilon'(f_a(z)) \sqsubseteq_{M'} f \cdot \varepsilon(f_s(y))].$$

$$\Longleftrightarrow \text{For all } z \text{ s.t. } \varepsilon'(f_s(z)) \sqsubseteq_{M'} \varepsilon'(f_s(x)), \quad \varepsilon'(f_s(z)) \sqsubseteq_{M'} f \cdot \varepsilon(f_s(y)).$$

Notice that there is a recursive function r s.t.

$$f_s(r(x)) = \{z \in N \mid \varepsilon'(f_s(z)) \sqsubseteq_{M'} \varepsilon'(f_s(x))\}.$$

Furthermore we have:

$$\varepsilon'(f_s(z)) \sqsubseteq_{M'} f \cdot \varepsilon(f_s(y))$$

$$\Longleftrightarrow \forall a \in f_s(z).\exists b \in f_s(y).\varepsilon'(a) \sqsubseteq f \cdot \varepsilon(b) \quad \& \quad \forall b \in f_s(y).\exists a \in f_s(z).\varepsilon'(a) \sqsubseteq f \cdot \varepsilon(b).$$

Since f is computable, $\varepsilon'(a) \sqsubseteq f \cdot \varepsilon(b)$ is r.e. in a and b. Thus $\varepsilon'(f_s(z)) \sqsubseteq_{M'} f \cdot \varepsilon(f_s(y))$ is r.e. in z and y. Notice that the above argument involves a recursive generation of a set $\{\langle z,y\rangle \mid \varepsilon'(f_s(z)) \sqsubseteq_{M'} f \cdot \varepsilon(f_s(y))\}$ from that of $\{\langle a,b\rangle \mid \varepsilon'(a) \sqsubseteq f \cdot \varepsilon(b)\}$. Thus we can effectively obtain a graph index of $\hat{f}$ from that of f. ☐

The next theorem states that the power domain constructor preserves the effective isomorphism.

Theorem 4.7.6

Let D^{ε} and $D'^{\varepsilon'}$ be effectively given SFP objects s.t. $D^{\varepsilon} \overset{e}{\cong} D'^{\varepsilon'}$. Then we have:

$$\mathcal{F}[D^{\varepsilon}] \overset{e}{\cong} \mathcal{F}[D'^{\varepsilon'}].$$

proof Let $(h:D \to D', h^R:D' \to D)$ be a computable isomorphism pair. It can readily be seen that $(\hat{h}, \hat{h}^R)$ is an isomophism pair from $\mathcal{F}[D]$ to $\mathcal{F}[D']$. By the previous theorem $(\hat{h}, \hat{h}^R)$ is a computable isomorphism pair. ▨

By almost the same arguments as in 2.2.8, we can observe that $\times, +, \to$ preserve effective isomorphism. Thus, with 4.7.6, we can identify D^{ε} and $D'^{\varepsilon'}$ if they are effectively isomorphic.

In theory, we should be able to prove that the power of an effectively given SFP object is again an effectively given SFP object, by showing $M[D^{\varepsilon}]$ is an effective basis. As far as the author can see, our method seems to be simpler.

4.8 Effective SFP Objects

By essentially routine extension of the arguments in chapter 3, we can obtain the notion of effective SFP objects, each of which is the set of all computable elements of an effectively given SFP object, or is the 'effective completion' of an effective finitary poset.

One outstanding point about effective SFP objects is that we can characterise this notion as the 'effective inverse limlit' of effective sequences of computable projection pairs of effective(ly given) finite cpo's. Notice that every effecively given finite cop is an effective finite cpo as it is.

CHAPTER 5: EFFECTIVE CATEGORIES

"Was sich überhaupt sagen lässt, lässe sich klar sagen; und wovon man nicht reden kann, darüber muss man schweigen."

Ludwig Wittgenstein.

"Whenever I played for Richter, he looked immovably at my fingures and one day he said; 'My God! how I am oblidged to torment myself and sweat, and yet without obtaining applause; and for you, my friend, it is mere play!" "Yes' said I, 'I had to labor once in order not to labor now."

W. A. Mozart.

Plotkin and Smyth [24] emphasized the importance of category theory for solving recursive domain equations. They showed that a single theory based on categorical notion could allow us to solve recursive domain equations over various classes of non-effective domains.

This chapter is concerned with an attempt to make effective Plotkin and Smyth's categorical approach. Such an attempt is important for the purpose of considering solutions of recursive domain equations over the class of effectively given domains, effective domains, effectively given SFP objects, r.e. sets etc, which are under the effectiveness constraint.

5.1 Effectively Initial Algebras

It immediately follows from the previous chapters that effectively given domains & either computable functions or computable projection pairs, effective domains & either f-computable functions or f-computable projection pairs, effectively given SFP objetcs & either computable functions or computable projection pairs, and effective SFP objects & either f-computable functions or f-computable projection pairs form categories. All of them which are under the constraint of effectiveness, have indexings associated to their object sets and morphism sets. Furthermore we have observed that almost all interesting effective properties of these categories can be described only in terms of these indexings. So we will start with those categories with which are associated indexings of object sets and morphism sets. Thence we characterize effectiveness of categorical constructions in terms of these indices. The most primitive categorical construction obviously is the composition of morphisms. This gives rise to the following notion:

Definition 5.1.1

(1) An indexed category is a triple $(\underline{K},\kappa,\partial)$ where $\underline{K}$ is a category, κ is a partial indexing (called an object indexing) of $Ob(\underline{K})$ and ∂ is a partial family of partial indexings (called morphism indexings) s.t. a partial indexing $\partial(i,j)$ is defined iff both $\kappa(i)$ and $\kappa(j)$ are defined; and it is a partial indexing of $Hom(\kappa(i),\kappa(j))$ whenever it is defined.

(2) An indexed category $(\underline{K},\kappa,\partial)$ is an effective category iff there are recursive functions ∂-*Compose* and *Idt* s.t.

$$\partial(i,k)(\partial\text{-}Compose(i,j,k,m,n))=\partial(j,k)(n)\cdot\partial(i,j)(m)$$

$$id_{\kappa(i)}=\partial(i,i)(Idt(i))$$ □

Notice that we are using the usual convention for equations involving partial functions. More specifically, if f and g are partial functions, by $f(x)=g(y)$ we mean that both $f(x)$ and $g(y)$ are defined and equal, or both of them are undefined.

A possible alternative to the above is to assume that the indexing of Hom(A,B) is independent to the representations (indices) of A and B. For our primary models like the category of effectively given domains and computable functions together with the acceptable indexing as an object indexing and the directed indexings as morphism indexings, this assumption is true. But Plotkin [31] indicated that this might lack in generality. Indeed the author discovered an interesting example where this assumption collasps. We can define the notion of partial computable functions from an r.e. set to another, thence obtain a reasonable category of r.e. sets and partial computable functions. We can present a natural way of indexing this category to make it effective and show that it can happen that:

$$\partial(i,j)(k) \neq \partial(m,n)(k)$$

even if $\kappa(i)=\kappa(m)$ and $\kappa(j)=\kappa(n)$.

Before studying this problem, remember that for each indexd r.e. set W_i, there is a canonical enumeration $\phi_{dv2(i)}$ s.t. if $W_i=\emptyset$ then $\phi_{dv2(i)}$ is everywhere undefined and if $W_i \neq \emptyset$ then $\phi_{dv2(i)}$ is total. Also remember that our acceptable indexing system satisfies $W_i=\text{range}(\phi_i)$.

<u>Definition 5.1.1.1</u>

A function $f:W_i \to W_j$ is <u>partially computable</u> and has a <u>c_1-index</u> k iff the following diagram (of partial functions) commutes:

$$\begin{array}{ccc} N & \xrightarrow{\phi_{dv2(i)}} & W_i \\ \phi_k \downarrow & & \downarrow f \\ N & \xrightarrow{\phi_{dv2(j)}} & W_j \end{array}$$

□

Notice that each k determines a unique partial computable function from W_i to W_j since $\phi_{dv2(i)}$ and $\phi_{dv2(j)}$ are surjective.

There are several cases to be checked. If $W_j=\emptyset$ then $\phi_{dv2(j)}$ is everywhere undefined and f is an empty function, thus for each $k\in N$ the above diagram commutes. In case $W_j=\emptyset$ and $W_i\neq\emptyset$, f is everywhere undefuned and is not total. If $W_i=\emptyset$ then f is total and is the empty function. In case $W_i=\emptyset$ and $W_j\neq\emptyset$ then ϕ_k is the empty function $N\to N$, for $\phi_{dv2(i)}$ is everywhere undefined.

The following lemma indicates that our notion of partial computability is a natural one.

Lemma 5.1.1.2

(1) A function $f:W_i\to W_j$ is partially computable iff it is the restriction to W_i of a partial recursive function $\overline{f}:N\to N$.

(2) If $\overline{f}=\phi_k$, we say that f has a c_2-index k. There are recursive functions *Ot* and *To* s.t. if k is a c_1-index of $f:W_i\to W_j$ then *To*(k) is a c_2-index of f and if k is a c_2-index of f then *Ot*(k) is a c_1-index of f.

proof For each $(i,j)\in N\times N$, define $To:N\to N$ and $Ot:N\to N$ to be recursive functions satisfying:

$$\phi_{To(x)}(n)=\phi_{dv2(j)}\cdot\phi_x(\mu m.[\phi_{dv2(i)}(m)=n])$$
$$\phi_{Ot(x)}(n)=\mu m.[\phi_{dv2(j)}(m)=\phi_x\cdot\phi_{dv2(i)}(n)].$$

Assume $f:W_i\to W_j$ is partially computable and has a c_1-index k. Then $\phi_{To(k)}$ is a partial recursive function which extends f to N. To observe this we show,

$$\begin{aligned}\phi_{To(k)}(n)&=\phi_{dv2(j)}\cdot\phi_k(\mu m.[\phi_{dv2(i)}(m)=n])\\&=f\cdot\phi_{dv2(i)}(\mu m.[\phi_{dv2(i)}(m)=n])\\&=f(n).\end{aligned}$$

Notice that the above equations are equations of partial functions. Conversely assume that f is the restriction to W_i of a partial recursive function ϕ_k:N→N. Then we have:

$$\phi_{dv2(j)} \cdot \phi_{Ot(k)}(n)$$
$$=\phi_{dv2(j)}(\mu m.[\phi_{dv2(j)}(m)=\phi_k \cdot \phi_{dv2(i)}(n)])$$
$$=\phi_k \cdot \phi_{dv2(i)}(n)$$
$$=f \cdot \phi_{dv2(i)}(n) \qquad (\because \phi_{dv2(i)}(n) \in W_i).$$

☒

The above proof states that from a c_1-index k of a partially computable function $f:W_i \to W_j$, we can construct a program $\bar{f}=\phi_{Ot(k)}$:N→N which computes f as the restriction to W_i. In this sense k is a good finite representation of f.

It can readily be seen that the category REC of r.e. sets and partially computable functions is well defined. There should be no objection to indexing the object set of this category by the acceptable indexing $\langle W_i \rangle$. There are two natural ways of indexing morphism sets. When we take c_1-indexings for the morphism indexings, we denote the resulting indexed category by REC1. REC2 denotes the indexed category where c_2-indexings are taken for morphism indexings.

It is very important to notice that in REC2, $\kappa(i)=\kappa(n)$ and $\kappa(j)=\kappa(m)$ implies $\partial(i,j)(k)=\partial(n,m)(k)$. But this is not the case in REC1. This is an example where the generality of indexing Hom($\kappa(i),\kappa(j)$) by $\partial(i,j)$ is needed. But by virtue of 5.1.1.2, we can do every interesting things of REC1 in REC2. Therefore we still are in search of more convincing examples which require the full generality of 5.1.1.

Theorem 5.1.1.3

Both REC1 and REC2 are effective categories.

proof Let C_p be a recursive function s.t. $\phi_{C_p(m,n)}=\phi_n\cdot\phi_m$. Define ∂-*Compose*$(i,j,k,m,n)=C_p(m,n)$. It can readily be seen that $\partial(j,k)(n)\cdot\partial(i,j)(m)=\partial(i,k)(\partial\text{-}Compose(i,j,k,m,n))$ in both REC1 and REC2 by easy diagram chasing. Let id_N be the idientity function from N to N. Let *Idt* be a recursive function s.t. $\phi_{Idt(n)}=id_N$. It can readily be seen that $id_{W_n}=\partial(n,n)(Idt(n))$ in both REC1 and REC2. ☐

In order to observe the naturalness of the assumption that the composition of morphisms is recursive in indices, remember that the composition of two partial recursive functions is recursive w.r.t. the acceptable indices. In fact the category whose object is a singleton {N} and whose morphisms are partial recursive functions, together with an obvious object indexing, say {(n,n)}, and acceptable indexing as the morphism indexing is an effective category, whose object indexing is not total. We will write PR to denote this effective category.

Now we will define effectiveness of various universality of category theory.

Definition 5.1.2

Let $(\underline{K},\kappa,\partial)$ be an effective category.

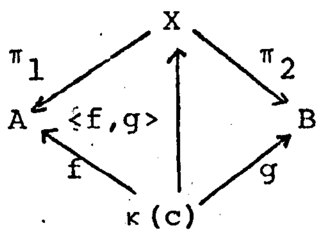
(1) An object $I\in Ob(\underline{K})$ is said to have initiality index $\langle i,j\rangle$ iff $\kappa(i)=I$ and $\phi_j:N\to N$ is a recursive function s.t. for every object index a, $\partial(i,a)(\phi_j(a))$ is the unique morphism from $\kappa(i)$ to $\kappa(a)$. An object I is an effectively initial object iff it it has an initiality index. We sometimes write Int_i for ϕ_j.

Effectively final objects can be defined as a dual to this.

(2) A triple $(X,\pi_1:X\to A,\pi_2:X\to B)$ has a (binary) product index $\langle\langle x,a,b,p_1,p_2\rangle,j\rangle$ iff $X=\kappa(x)$, $A=\kappa(a)$, $B=\kappa(b)$, $\pi_1=\partial(x,a)(p_1)$, $\pi_2=\partial(x,b)(p_2)$, and ϕ_j is a recursive function s.t. for any indexed morphisms $f=\partial(c,b)(m)$ and $g=\partial(c,a)(n)$,

$$\langle f,g\rangle=\partial(c,x)(\phi_j(c,m,n))$$

is the unique morphism which commutes the following diagram:



We write $A\times B$ for X. A triple $(X,\pi_1:X\to A,\pi_2:X\to B)$ is an effective (binary) product (of A and B) iff it has a product index. $(\underline{K},\kappa,\partial)$ is effective (binary) product closed iff there are recursive functions $Prod, P_1, P_2$, and $P\text{-}Med$ s.t. for every pair of object indicies (a,b),

$$(\kappa(Prod(a,b)),\partial(Prod(a,b),a)(P_1(a,b)),\partial(Prod(a,b),b)(P_2(a,b)))$$

is an effective product with a product index:

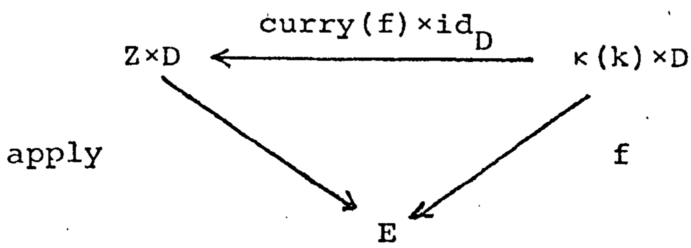
$$\langle\langle Prod(a,b),a,b,P_1(a,b),P_2(a,b)\ ,P\text{-}Med(a,b)\rangle.$$

Effective binary coproducts can be defined as dual to this.

(3) Assume $(\underline{K},\kappa,\partial)$ is effective binary product closed. An ordered pair $(z\in Ob(\underline{K}),apply:Z\times D\to E)$ has an exponentiation index $\langle\langle z,d,e,ap\rangle,j\rangle$ iff $Z=\kappa(z)$, $D=\kappa(d)$, $E=\kappa(e)$, $apply=\partial(Prod(z,d),e)(ap)$, and for any $f=\partial(Prod(k,d),e)(m)$,

$$curry(f)=\partial(k,z)(\phi_j(k,m))$$

is the unique morphism which commutes:



We write (D E) for Z. Also we write $Curry_{<z,d,e,ap>}$ for ϕ_j. An ordered pair (Z,apply:Z×D→E) is an effective exponentiation of (D and E) iff it has an exponentiation index. ($\underline{K}$,κ,∂) is said to be effectively Cartesian closed iff it has effectively final object and there are recursive functions *Apply*,*Curry*,*Exp* s.t. for any pair (d,e) of object indices a pair:

(κ(*Prod*(*Exp*(d,e),d)),∂(*Prod*(*Exp*(d,e),d),e)(*Apply*(d,e)))

is an effective exponentiation with an exponentiation index:

<<*Exp*(d,e),d,e,*Apply*(d,e)>,*Curry*(a,b)>.

□

It is a common exercise to represent objects subject to computation as partial recursive functions. For example, an r.e. set can be represented by a partial recursive function whose range is the r.e. set. In this way even the set of natural numbers can be indexed by a (non-r.e.) subset of natural numbers whenever we identify n={n}. Our fundamental philosophy is that when we talk about effective categories, we are essentially talking about systems of partial recursive functions (or programs). We claim that almost all categorical constructions are program transformations which always terminate. This observation support our decision to take ∂-*Compose*,P_1,P_2,*Prod*,*P-Med*,*Exp*,*Apply*,*Curry* etc. as "recursive" functions.

One might worry about the notion of recursive functions from non-r.e. set to another, which we might be forced to consider for κ and ∂ could be partial. But this notion is quite natural. Indeed this notion is a quite natural extension of the notion of partially computable functions from an r.e. set to another. For example, let $S_j \subseteq N$ and $F_j = \{\phi_i \mid i \in S_j\}$, $j=1,2$. We know: $\phi_i \cdot \phi_j = \phi_{Cp(i,j)}$ for all i and j. Regardless of whether S_1 and S_2 are r.e. or not, Cp maps from $S_1 \times S_2$ to $Cp(S_1 \times S_2)$. This states that the composition of functions in F_1 and F_2 is recursive in indices.

Definition 5.1.3

Let $(\underline{K}, \kappa, \partial)$ be an effective category.

(1) An ω-diagram is a functor $G: \underline{\omega} \to \underline{K}$, where $\underline{K}$ is the category: $0 \leq 1 \leq 2 \leq \cdots\cdots$. It has a codiagram index j iff ϕ_j is a recursive function s.t.:

$$G(n) = \kappa(\pi_1 \cdot \phi_j(n))$$

$$G(n \leq n+1) = \partial(\pi_1 \cdot \phi_j(n), \pi_1 \cdot \phi_j(n+1))(\pi_2 \cdot \phi_j(n)).$$

The ordered pair (G,j) is called an indexed ω-codiagram. Indexed ω-diagrams can be defined as dual to this.

(2) Given an indexed ω-codiagram (G,j), a cocone $\lambda = \langle \lambda_n : G(n) \to C \rangle$ is said to have a cocone index $\langle c,k \rangle$ iff $C = \kappa(c)$ and ϕ_k is a recursive function s.t.:

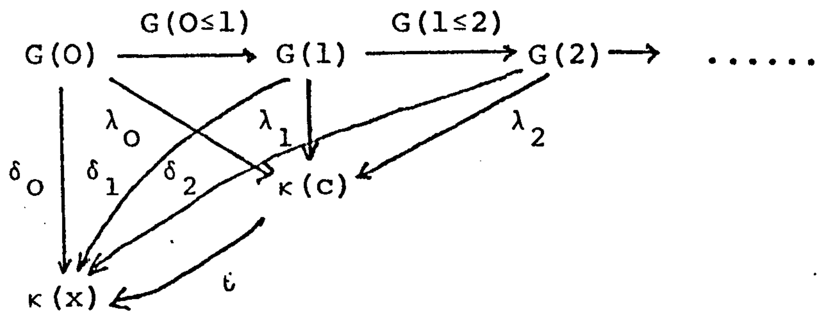
$$\lambda_n = \partial(\pi_1 \cdot \phi_j(n), c)(\phi_k(n)).$$

The ordered pair $(\lambda, \langle c,k \rangle)$ is called an indexed ω-cocone of (G,j). Notice that the effective generation of λ_n is dependent on the index of G. As dual to the above, we can define indexed ω-cone of an indexed ω-diagram.

(3) An ω-cocone λ of the indexed ω-codiagram (G,j) has an <u>ω-colimit index</u> $\langle\langle c,k\rangle,i\rangle$ iff $(\lambda,\langle c,k\rangle)$ is an indexed ω-cocone of (G,j) and $\phi_i:N\to N$ is a recursive function s.t.: for each indexed ω-cocone $(\delta,\langle x,y\rangle)$ of (G,j),

$$\theta=\partial(c,x)(\phi_i(x,y))$$

is the unique morphism which makes the following diagram commute :



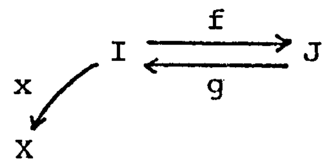
An ω-cocone of an indexed ω-codiagram is an <u>effective ω-colimiting cocone</u> of the indexed ω-codiagram iff it has an ω-colimit index. As dual to the above, we can define <u>effective ω-limiting cones</u> of indexed ω-diagrams.

(4) $(\underline{K},\kappa,\partial)$ is <u>effectively ω-cocomplete</u> iff there are recursive functions ω-*Colim*, ω-*Cocone*, and ω-*Comed* s.t. for every indexed ω-codiagram (G,j), there is an effective ω-colimiting cocone with an ω-colimit index $\langle\langle\omega\text{-}Colim(j),\omega\text{-}Cocone(j)\rangle,\omega\text{-}Comed(j)\rangle$. As dual to this, we can define <u>effective ω-completeness</u>.

□

<u>NOTE</u> To make the dependence of ω-*Colim*, ω-*Cocone* and ω-*Comed* on $(\underline{K},\kappa,\partial)$ explicit, we write ω-*Colim*$(\underline{K})$, ω-*Cocone*$(\underline{K})$ and ω-*Comed*$(\underline{K})$. The same convention will be applied for *Prod*, P_1, P_2, *P-Med*, *Apply*, *Curry*, *Exp*, ∂-*Compose*, *Idt* etc.

In non-constructive category theory, it is a common exercise to identify objects to within isomorphism. The reason why we can do this is that the universaity is invariant under the isomorphism. For example, let I be an initial object in $\underline{K}$ and $f:I\to J$ is an isomorphism with the inverse $g:J\to I$. Let x be the unique morphism from I to X.



f obviously is the unique morphism from I to J. We claim that x $g:J\to X$ is the unique morphism from J to X. Let $c:J\to X$ be a morphism from I to J. By the uniqueness of x, $c\cdot f=x\cdot g\cdot f$. Since f is an isomorphism it is an epimorphism. Thus $c=x\cdot g$. Therefore J is an initial object.

Our question is, when we can identify two objects in effective categories. Shall we ask some kind of effectiveness for this purpose? More specifically, need we have recursive functions Ism_1 and Ism_2 s.t. $\partial(i,j)(Ism_1(i,j))$ is an isomorphism from $\kappa(i)$ to $\kappa(j)$ with the inverse (adjoint) $\partial(j,i)(Ism_2(j,i))$? We claim that this condition is too strong. In fact two effectively given domains $\overline{\xi}(i)$ and $\overline{\xi}(j)$ are identical iff there exists a pair $(f:\overline{\xi}(i)\to\overline{\xi}(j), f^R:\overline{\xi}(j)\to\overline{\xi}(i))$ of computable functions s.t. $f\cdot f^R=id_{\overline{\xi}(i)}$ and $f^R\cdot f=id_{\overline{\xi}(j)}$. We do not need to be able to obtain directed indices of f and f^R from the acceptable indices i and j. As this case of effectively given domains indicates, we will identify two objects of an effective category to within isomorphism. We will demonstrate, in the following, that this identification is natural. More specifically, we will observe that effective universality is invariant under the isomorphism. Rather than proving this claim in general setting, we will check an important example.

Let $(\Delta=\langle A_n, f_n\rangle, i)$ be an indexed ω-codiagram of an effective category $(\underline{K},\kappa,\partial)$. Let μ be an effective ω-colimiting cocone of (Δ,i) with an index $\langle\langle a,k\rangle,j\rangle$. Furthermore let $A'=\kappa(a')$ be an object isomorphic to $A=\kappa(a)$, via an isomorphism pair $(h,h^R):\kappa(a)\to\kappa(a')$. Then for some recursive function $r:N\to N$, $\nu=\langle\nu_n=h\cdot\mu_n:A_n\to A'\rangle$ is an effectively generable cocone of (Δ,i) with an index $\langle a',r(k)\rangle$, for the composition is recursive in indices. Since μ is an effective ω-colimit with an index $\langle\langle a,k\rangle,j\rangle$, h is the unique mediating morphism from μ to ν. Also $h=\partial(a,a')(\phi_j(a',r(k)))$. Now let $(\delta:\Delta\to B,\langle b,x\rangle)$ be an indexed cocone of (Δ,i) Then $\theta=\partial(a,b)(\phi_j(b,x))$ is the unique mediating morphism from μ to δ. But $\theta\cdot h^R$ is the unique mediating morphism from ν to δ. Let $h^R=\partial(a',a)(n)$ for some n. Then we have:

$$\begin{aligned}\theta\cdot h^R&=\partial(a',b)(\partial\text{-}Compose(a',a,b,\phi_j(b,x),n))\\&=\partial(a',b)(\phi_{s(j)}(b,x))\end{aligned}$$

where s is a recursive function s.t.:

$$\phi_{s(j)}(b,x)=\partial\text{-}Compose(a',a,b,\phi_j(b,x),n).$$

Therefore ν is an effective ω-colimitimg cocone of (Δ,i) with an index $\langle\langle a',r(k)\rangle,s(j)\rangle$.

We are now in a position to define effectiveness of functors. Plotkin and Beynon kindly indicated to the author an error in the previous definition of effective (it was called semi-effective) functors.

<u>Definition 5.1.4</u>

Given effective categories $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$, a functor $F:\underline{K}\to\underline{K}'$ is <u>effective</u> (wrt (κ,∂) and (κ',∂')) iff there are recursive functions $f_{ob(F)}$ and $f_{mr(F)}$ s.t.:

$F(\kappa(n))=\kappa'(f_{ob(F)}(n))$

$F(\partial(i,j)(m))=\partial'(f_{ob(F)}(i),f_{ob(F)}(j))(f_{mr(F)}(i,j,m))$.

In case $f_{ob(F)}=\phi_i$ and $f_{ob(F)}=\phi_j$, we say that $\langle i,j\rangle$ is a <u>functor index</u> of F. We write $F:(\underline{K},\kappa,\partial)\to(\underline{K}',\kappa',\partial')$ to denote that F is an effective functor. □

The following lemma states that an effective functor maps effectively, indexed ω-codiagrams to indexed ω-codiagrams and indexed ω-cocones to indexed ω-cocones.

<u>Lemma 5.1.5</u>

Let $(\underline{K},\kappa,\partial)$ and (K',κ',∂') be effective categories.

(1) There is a recursive function f_{dg} s.t. if $F:\underline{K}\to\underline{K}'$ is an effective functor with a functor index $\langle x,y\rangle$ and (G,i) is an indexed ω-codiagram then $(F\cdot G,f_{dg}(i,x,y))$ is an indexed ω-codiagram in $(\underline{K}',\kappa',\partial')$.

(2) There is a recursive function f_{cocone} s.t. if (G,i) is an indexed ω-codiagram, $(\mu:G\to A,\langle a,j\rangle)$ is an indexed ω-cocone of (G,i) and $F:\underline{K}\to\underline{K}'$ is an effective functor with a functor index $\langle x,y\rangle$ then $F\mu:F\cdot G\to FA$ is an effectively generable ω-cocone of $(F\cdot G,f_{dg}(i,x,y))$ with an index $f_{cocone}(x,y,a,j)$.

<u>proof</u> (1) $F\cdot G(n)=\kappa'(\phi_x\cdot\pi_1\cdot\phi_i(n))$

$F\cdot G(n\le n+1)=\partial'(\phi_x\cdot\pi_1\cdot\phi_i(n),\phi_x\cdot\pi_1\cdot\phi_i(n+1))$
$(\phi_y(\phi_x\cdot\pi_1\cdot\phi_i(n),\phi_x\cdot\pi_1\cdot\phi_i(n+1),\pi_2\cdot\phi_i(n)))$

Define $g'(n)=\langle\phi_x\cdot\pi_1\cdot\phi_i(n),\phi_y(\phi_x\cdot\pi_1\cdot\phi_i(n),\phi_x\cdot\pi_1\cdot\phi_i(n+1),\pi_2\cdot\phi_i(i))\rangle$.

Then we have:

$F\cdot G(n)=\kappa'(\pi_1\cdot g'(n))$

$F\cdot G(n\le n+1)=\partial'(\pi_1\cdot g'(n),\pi_1\cdot g'(n+1))(\pi_2\cdot g'(n))$.

The construction of g' is uniform in i,x and y.

(2) $F\mu_n=\partial'(\phi_x\cdot\pi_1\cdot\phi_i(n),\phi_x(a))(\phi_y(\phi_x\cdot\pi_1\cdot\phi_i(n),\phi_x(a),\pi_2\cdot\phi_j(n)))$

$$=\partial'(\pi_1\cdot\phi_{f_{dg}(i,x,y)}(n),\phi_x(a))$$

$$(\phi_y(\pi_1\cdot\phi_{f_{dr}(i,x,y)}(n),\phi_x(a),\phi_x(a),\pi_2\cdot\phi_j(n))).$$

Define $f_{cocone}(x,y,a,j)=<\phi_x(a),h(y,j)>$, where $h(y,j)=C_p(y,C_p(e,j))$ and $\phi_e=\pi_2$. Then f_{cocone} is a recursive function and:

$$F(\mu_n)=\partial'(\pi_1\cdot\phi_{f_{dg}(i,x,y)}(n),\pi_1\cdot f_{cocone}(x,y,a,j))$$
$$(\phi_{\pi_2\cdot f_{Cocone}(x,y,a,j)}(n)).$$

▨

Definition 5.1.6

Let $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$ be effective categories and $F:\underline{K}\to\underline{K}'$ be an effective (wrt (κ,∂) and (κ',∂')) functor with a functor index $<i,j>$. F has a continuity index $<<i,j>,k>$ iff ϕ_k is a recursive function s.t. whenever μ is an effective ω-colimiting cocone of an indexed ω-codiagram (G,x) in $(\underline{K},\kappa,\partial)$, with an index $<<a,n>,y>$ then $F\mu$ is an effective ω-colimiting cocone of an indexed ω-codiagram $(F\cdot G,f_{dg}(x,i,j))$ with an ω-colimit index $<f_{cocone}(i,j,a,n),\phi_k(a,n,y,x)>$.

□

Notice that in 5.1.5 and 5.1.6, f_{cocone} and f_{dg} depend on $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$. To make this dependency explicit, we may write $f_{cocone}(\underline{K},\underline{K}')$ and $f_{dg}(\underline{K},\underline{K}')$.

Definition 5.1.7

Given an effective category $(\underline{K},\kappa,\partial)$ and an effective functor $F:\underline{K}\to\underline{K}'$, an F-algebra is a pair (A,α) where $\alpha:FA\to A$. An F-homomorphism from an F-algebra (A,α) to another (B,β) is a $\underline{K}$-morphism $f:A\to B$ which makes the following diagram commute:

$$\begin{array}{ccc} FA & \xrightarrow{\alpha} & A \\ \downarrow{\scriptstyle Ff} & & \downarrow{\scriptstyle f} \\ FB & \xrightarrow{\beta} & B \end{array} .$$

If $A=\kappa(a)$ and $\alpha=\partial(f_{ob(F)}(a),a)(i)$, we say that (A,α) has an <u>algebra index</u> $\langle a,i\rangle$. □

It can readily be seen that the category of F-algebras and F-homomorphisms together with the algebra indexing as the object indexing and the derived (from ∂) indexings as morphism indexings, in symbol $\underline{A}^F$, is an effective category. Notice that $id_{(A,\alpha)}=id_{\kappa(a)}$ where (A,α) has an index $\langle a,i\rangle$.

<u>Definition 5.1.8</u>

Let $(\underline{K},\kappa,\partial)$ be an effective category and $F:\underline{K}\to\underline{K}$ be an effective functor. An F-algebra (A,α) has an <u>initial algebra index</u> $\langle\langle a,i\rangle,k\rangle$ iff (A,α) has an algebra index $\langle a,i\rangle$ and ϕ_k is a recursive function s.t. for any F-algebra (B,β) with an algebra index $\langle b,j\rangle$,

$$\partial(a,b)(\phi_k(b,j))$$

is the unique F-homomorphism from (A,α) to (B,β). An F-algebra is an <u>effectively initial F-algebra</u> iff it has an initial algebra index. □

It can readily be seen that an initial algebra index is an initiality index in $\underline{A}^F$, thus an effectively initial F-algebra (I,ι) is an effectively initial object in $\underline{A}^F$. Furthermore it is an isomorphism, thus is a solution of a recursive object equation $F(X)=X$. Indeed it is the effectively initial object in the category of the solutions (to within isomorphism) of $F(X)=X$ together with the induced (from $\underline{A}^F$) indexings.

Theorem 5.1.9 (The effectively initial algebra theorem)

Let $(\underline{K},\kappa,\partial)$ be an effective category with the effectively initial object with an initiality index $\langle i,j\rangle$.

(1) There is a recursive function f_{gn} s.t. for every effective functor $F:\underline{K}\to\underline{K}$ with an index $\langle d,e\rangle$, $f_{gn}(d,e)$ is an index of the effectively generable ω-codiagram Δ defined by:

$$\Delta(0)=\bot \qquad \Delta(n)=F^n(\bot)$$

$$\Delta(0\le 1)=\bot_{F\bot} \qquad \Delta(n\le n+1)=F^n\bot_{F\bot}$$

where $\bot_{\kappa(x)}=\partial(i,x)(\phi_j(x))$ is the unique morphism from $\bot$ to $\kappa(x)$.

(2) Let $F:\underline{K}\to\underline{K}$ be an effective functor with an index $\langle d,e\rangle$. Assume $\mu:\Delta\to\kappa(a)$ is an effective ω-colimiting cocone of the indexed ω-codiagram $(\Delta,f_{gn}(d,e))$ with an index $\langle\langle a,k\rangle,x\rangle$. Also assume that $F\mu:F\cdot\Delta\to F(\kappa(a))$ is an effective ω-colimiting cocone of $(F\cdot\Delta,f_{dg}(f_{gn}(d,e),d,e))$ with an index $\langle f_{cocone}(d,e,a,k),y\rangle$. Then the effectively initial F-algebra exists.

(3) In case $(\underline{K},\kappa,\partial)$ is effectively ω-cocomplete, there is a recursive function *Efin* s.t. if $F:\underline{K}\to\underline{K}$ is an effectively continuous functor with a continuity index $\langle\langle d,e\rangle,c\rangle$, then *Efin*$(d,e,c)$ is an initial algebra index of the effectively initial F-algebra.

<u>proof</u> (1) $\Delta(n)=F^n\bot=\kappa(\phi_d^n(i))$. Also $\bot_{F\bot}=\partial(i,\phi_j(\phi_d(i)))$.

Thus:
$$\begin{aligned}\Delta(n\le n+1)&=F^n(\bot_{F\bot})\\&=\partial(\phi_d^n(i),\phi_d^{n+1}(i))\\&\qquad(\phi_e(\phi_d^n(i),\phi_d^{n+1}(i),\phi_e(\phi_d^{n-1}(i),\phi_d^n(i),\ldots\\&\qquad\qquad\phi_e(i,\phi_d(i),\phi_j(\phi_d(i)))\ldots))).\end{aligned}$$

Take $g(n)=$<u>if</u> $n=0$ <u>then</u> $\langle i,\phi_j(\phi_d(i))\rangle$

$$\underline{\text{else}}\ \langle \phi_d^n(i), \phi_e(\phi_d^n(i), \phi_d^{n+1}(i), \phi_e(\phi_d^{n-1}(i), \phi_d^n(i), \ldots \phi_e(i, \phi_d(i), \phi_j(\phi_d(i)))\ldots))\rangle.$$

Obviously g generates Δ. Since g is constructed uniformly in d and e, there is a recursive function f_{gn} s.t.

$$g=\phi_{f_{gn}(d,e)}.$$

(2) Since μ is an effective ω-colimiting cocone of $(\Delta, f_{gn}(d,e))$ with an index $\langle\langle a,k\rangle,x\rangle$, $(\langle\mu_n\rangle_{n\geq 1}, \langle a, suc(k)\rangle)$ is an indexed ω-cocone of $(F\cdot\Delta, f_{dg}(f_{gn}(d,e)))$ where suc is a recursive function satisfying: $\phi_{suc(z)}(x)=\phi_z(x)+1$.

Since $F\mu$ is an effective ω-colimiting cocone of $(F\cdot\Delta, f_{dg}(f_{gn}(d,e),d,e)$ with an index $\langle f_{cocone}(d,e,a,k),y\rangle$, there is a unique morphism $\alpha:F(\kappa(a))\to\kappa(a)$ s.t. $\mu_{i+1}=\alpha\cdot F(\mu_i)$. Indeed we have:

$$\alpha=\partial(\phi_d(a),a)(\phi_y(a,suc(k))).$$

We claim that $(\kappa(a),\alpha)$ is an effectively initial F-algebra. Let $\beta=\partial(\phi_d(b),b)(m)$ be an F-algebra with an algebra index $\langle b,m\rangle$.

Define $\nu_n:F^n\bot\to\kappa(b)$ by:

$$\nu_0=\bot_{\kappa(b)}$$

$$\nu_{n+1}=\beta\cdot F(\nu_n).$$

By induction on n, we can show that $\nu=\langle\nu_n\rangle$ is a cocone of Δ.

We have: $\nu_0=\bot_{\kappa(b)}=\partial(i,b)(\phi_j(b))$

$$\begin{aligned}\nu_{n+1}&=\beta\cdot F(\nu_n)\\ &=\partial(\phi_d(b),b)(m)\cdot\partial(\phi_d^{n+1}(i),\phi_d(b))\\ &\qquad(\phi_e(\phi_d^{n+1}(i),b,\phi_e(\phi_d^n(i),b,\ldots\phi_e(i,b,\phi_j(b))\ldots)))\\ &=\partial(\phi_d^{n+1}(i),b)\\ &\qquad(\partial\text{-}Compose(\phi_d(b),b,\phi_d^{n+1}(i),m,\\ &\qquad\qquad\phi_e(\phi_d^{n+1}(i),b,\ldots,\phi_e(i,b,\phi_j(b))\ldots)).\end{aligned}$$

Let $g(n)=$ _if_ $n=0$ _then_ $\langle i,\phi_j(b)\rangle$

$$\underline{\text{else}}\ \langle\phi_d^n(i),\partial\text{-}Compose(\phi_d(b),b,\phi_d^{n+1}(i),m,\ \phi_e(\phi_d^{n+1}(i),b,..,\phi_e(i,b,\phi_j(b))..))\rangle.$$

Obviously g is recursive and is constructed uniformly in b and m. Therefore $g=\phi_{r(b,m)}$ for some recursive function r. Also g generates $\nu=\langle\nu_n\rangle$. Thus $(\nu,\langle b,r(b,m)\rangle)$ is an indexed ω-cocone of $(\Delta,f_{gn}(d,e))$. Thus there is a unique morphism

$$\gamma=\partial(a,b)(\phi_x(b,r(b,m)))$$

s.t. $\gamma\cdot\mu_i=\nu_i$. By using the initiality of $\perp$, it can readily be seen that γ is the unique F-homomorphism from $(\kappa(a),\alpha)$ to $(\kappa(b),\beta)$. Now let f be a recursive function s.t.

$$\phi_{f(u)}(m,n)=\phi_u(m,r(m,n)).$$

Then $\langle\langle a,\phi_y(a,suc(k))\rangle,f(x)\rangle$ is an initial algebra index of $(\kappa(a),\alpha)$.

(3) In case $(\underline{K},\kappa,\partial)$ is effectively ω-cocomplete, we have:

$$a=\omega\text{-}Colim(f_{gn}(d,e))$$
$$k=\omega\text{-}Cocone(f_{gn}(d,e))$$
$$x=\omega\text{-}Comed(f_{gn}(d,e)).$$

Since F has a continuity index $\langle\langle d,e\rangle,c\rangle$, in the above proof we can take $y=\phi_c(a,k,x,f_{gn}(d,e))$. Thus $(\kappa(a),\alpha)$ has an initial algebra index:

$$\langle\langle\omega\text{-}Colim\ f_{gn}(d,e)),\ \phi_{\phi_c(\omega\text{-}Colim(f_{gn}(d,e)),\omega\text{-}Cocone(f_{gn}(d,e)),\omega\text{-}Comed(f_{gn}(d,e)),f_{gn}(d,e))}(\omega\text{-}Colim(f_{gn}(d,e)),suc(\omega\text{-}cocone(f_{gn}(d,e))))\rangle,\ f(\omega\text{-}comed(f_{gn}(d,e)))\rangle.$$

Define *Efin* by:

$$Efin(u,w,z)$$
$$=\langle\langle \omega\text{-}Colim(f_{gn}(u,w)),$$
$$\phi_{\phi_z(\omega\text{-}Colim(f_{gn}(u,w)),\omega\text{-}Cocone(f_{gn}(u,w)),\omega\text{-}Comed(f_{gn}(u,w)),}$$
$$f_{gn}(u,w))^{(\omega\text{-}Colim(f_{gn}(u,w)),suc(\omega\text{-}Cocone\ f_{gn}(u,w))))\rangle,}$$
$$f(\omega\text{-}Comed(f_{gn}(u,w)))\rangle.$$

☒

Definition 5.1.10

An effective ω-category is an effectively ω-cocomplete category with an effectively initial object. □

Corollary 5.1.10

Let $(\underline{K},\kappa,\partial)$ be an effective ω-category and $F:\underline{K}\to\underline{K}$ be an effectively continuous functor with a continuity index $\langle\langle d,e\rangle,c\rangle$. Then $Efin(d,e,c)$ is an initial algebra index of the effectively initial F-algebra. ☒

5.1.9-(3) and 5.1.11 are concerned with the effective construction of the effectively initial F-algebra, for each effectively continuous functor F.

It can readily be seen that the following (indexed) categories are all effective categories:

Category Names	Symbols
(I) effectively given domains & (strict) computable maps	EGD(EGD*)
(2) effective domains & (strict) f-computable maps	ED(ED*)
(3) effectively given SFP objects & (strict) computable maps	EGS(EGS*)
(4) effective SFP objects & (strict) f-computable maps	ES(ES*)

where the associated object indexings are acceptable indexings and the morphism indexings are directed indexings. It should be noted that before the work of the author, it was not quite clear even what should be taken for morphisms for the class of effectively given domains etc in order to form reasonable categories. Indeed without the notion of effective ω-category, it can hardly be seen that EGD etc behave usefully.

Furthermore we can easily observe that the following categories are all effective categories:

Category names	Symbols
(1) effectively given domains & computable projection pairs	EGD^P
(2) effective domains & f-computable projection pairs	ED^P
(3) effectively given SFP objects & computable projection pairs	EGS^P
(4) effective SFP objects & f-computable projection pairs	ES^P

where the associated indexings are as above.

As we will observe later, in 5.2.23, EGD^P, ED^P, EGS^P, and ES^P are all effective ω-categories.

A rather simpler example of effective ω-category is an effective domain. Indeed we have:

<u>Example 5.1.12</u>

(1) An effective domain X^ε (regarded as a category) together with the directed indexing χ_ε as an object indexing and the obvious morphism indexings is an effective category. Furthermore it is an effective ω-category, for we have 3.1.2.

(2) An f-computable function is an effectively continuous functor. If $f:X^\varepsilon \to X^\varepsilon$ is an f-computable function, then the least fixed-point of f given by:

$$fix(f)=\bigsqcup_i f^i(\bot)$$

is the effectively initial solution of the recursive object equation f(x)=x. Notice that by 3.1.2, fix(f) can effectively be obtained from f, an example of 5.1.11. ∎

Definition 5.1.13

Given indexed categories $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$, let $(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')$ be the following indexed category:

$$(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')=(\underline{K}\times\underline{K}',\kappa\times\kappa',\partial\times\partial')$$

where $\underline{K}\times\underline{K}'$ is the product category of $\underline{K}$ and $\underline{K}'$, and

$$\kappa\times\kappa'(n)=(\kappa(\pi_1(n)),\kappa'(\pi_2(n)))$$

$$\partial\times\partial'(i,j)(n)$$

$$=(\partial(\pi_1(i),\pi_1(j))(\pi_1(n)),\partial(\pi_2(i),\pi_2(j))(\pi_2(n))).$$

∎

Lemma 5.1.14

Let $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$ be effective (ω-)categories, then so is $(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')$.

proof Obviously $(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')$ is effective. There are recursive functions lt and rt s.t. $\phi_{lt(x)}=\pi_1\cdot\phi_x$ and $\phi_{rt(x)}=\pi_2\cdot\phi_x$. Since $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$ are effective ω-categories, there are recursive functions ω-*Colim*($\underline{K}$), ω-*Colim*($\underline{K}'$), ω-*Cocone*($\underline{K}$), ω-*Cocone*($\underline{K}'$), ω-*Comed*($\underline{K}$) and ω-*Comed*($\underline{K}'$) which behave as in (4)-5.1.3. Define recursive functions ω-*Colim*($\underline{K}$ $\underline{K}'$), ω-*Cocone*($\underline{K}$ $\underline{K}'$) and ω-*Comed*($\underline{K}$ $\underline{K}'$) by:

$$\omega\text{-}Colim(\underline{K}\ \underline{K}')=\langle\omega\text{-}Colim(\underline{K})(lt(x)),\omega\text{-}Colim(\underline{K}')(rt(x))\rangle$$

$$\omega\text{-}Cocone(\underline{K}\ \underline{K}')=pair(\omega\text{-}Cocone(\underline{K})(lt(x)),\omega\text{-}Cocone(\underline{K}')(rt(x)))$$

$$\omega\text{-}Comed(\underline{K}\ \underline{K}')=pair(\omega\text{-}Comed(\underline{K})(lt(x)),\omega\text{-}Comed(\underline{K}')(rt(x)))$$

where *pair* is a recursive function satisfying:

$$\phi_{pair(i,j)}(x)=\langle\phi_i(x),\phi_j(x)\rangle.$$

Now let $G = \langle (A_n, A'_n), (f_n, f'_n) \rangle$ be an effectively generable ω-codiagram in $(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')$ with an index d. Then $G^L = \langle A_n, f_n \rangle$ is an effectively generable ω-codiagram in $(\underline{K},\kappa,\partial)$ with an index *lt*(d), and $G^R = \langle A'_n, f'_n \rangle$ is an effectively generable ω-codiagram in $(\underline{K}',\kappa',\partial')$ with an index *rt*(d). Since both $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$ are effectively ω-cocomplete, there are effective ω-colimiting cocones δ of $(G^L, lt(d))$ and δ' of $(G^R, rt(d))$ with indices:

$\langle\langle \omega\text{-}Colim(\underline{K})(lt(d)), \omega\text{-}Cocone(\underline{K})(lt(d))\rangle, \omega\text{-}Comed(\underline{K})(lt(d))\rangle$,

$\langle\langle \omega\text{-}Colim(\underline{K}')(rt(d)), \omega\text{-}Cocone(\underline{K}')(rt(d))\rangle, \omega\text{-}Comed(\underline{K}')(rt(d))\rangle$

respectively. It is obvious that if $\delta\times\delta' = \langle(\delta_n, \delta'_n)\rangle$, then $\delta\times\delta'$ is an effective ω-colimiting cocone of (G,d) with an index:

$\langle\langle \omega\text{-}Colim(\underline{K}\times\underline{K}')(d), \omega\text{-}Cocone(\underline{K}\times\underline{K}')(d)\rangle, \omega\text{-}Comed(\underline{K}\ \underline{K}')(d)\rangle$.

Thus $(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')$ is effectively ω-cocomplete. Also if $\bot$ and $\bot'$ are effectively initial objects of $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$ respectively then $(\bot,\bot')$ is an effectively initial object in $(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')$.

☑

<u>Lemma 5.1.15</u>

Let $(\underline{K},\kappa,\partial)$, $(\underline{K}',\kappa',\partial')$ and $(\underline{K}'',\kappa'',\partial'')$ be effective. There is a recursive function *f-Compose* s.t. if ⟨a,b⟩ and ⟨c,d⟩ are functor indices of effective functors $F:\underline{K}\rightarrow\underline{K}'$ and $F':\underline{K}'\rightarrow\underline{K}''$ respectively, then *f-Compose*(a,b,c,d) is a functor index of $F'\cdot F$. Also there is a recursive function *cf-Compose* s.t. if F and F' are effectively continuous functors with continuity indices ⟨⟨a,b⟩,x⟩ and ⟨⟨c,d⟩,y⟩ respectively, then $F'\cdot F$ is effectively continuos and has a continuity index *cf-Compose*(a,b,x,c,d,y).

<u>proof</u> Define h to be a recursive function satisfying:

$$\phi_{h(x,y,z,w)}=\lambda i.\lambda j.\lambda n.\phi_w(\phi_{Cp(x,z)}(i),\phi_{Cp(x,z)}(j),\phi_y(i,j,n)).$$

Let $f\text{-}Compose(x,y,z,w)=\langle Cp(x,z),h(x,y,z,w)\rangle$.

We have: $f_{Ob(F'\cdot F)}(x)=f_{Ob(F')}\cdot f_{Ob(F)}(x)=\phi_c\cdot\phi_a(x)=\phi_{Cp(a,c)}(x)$,

$$f_{mr(F'\cdot F)}(i,j,n)=f_{mr(F')}(f_{Ob(F'\cdot F)}(i),f_{Ob(F'\cdot F)}(j),f_{mr(F)}(i,j,n))$$
$$=\phi_d(\phi_{Cp(a,c)}(i),\phi_{Cp(a,c)}(j),\phi_b(i,j,n))$$
$$=\phi_{h(a,b,c,d)}(i,j,n).$$

Thus $f\text{-}Compose(a,b,c,d)=\langle Cp(a,c),h(a,b,c,d)\rangle$ is a functor index of $F'\cdot F$. Define $cf\text{-}Compose$ by:

$$cf\text{-}Compose(e,f,g,h,i,n)=\langle f\text{-}Compose(e,f,h,i),M\text{-}Compose(e,f,g,m)\rangle$$

where $M\text{-}Compose$ is a recursive function satisfying:

$$\phi_{M\text{-}Compose(e,f,g,m)}(x,y,z,i)$$
$$=\phi_m(\pi_1\cdot f_{cocone}(e,f,x,y),\pi_2\cdot f_{cocone}(e,f,x,y),$$
$$\phi_g(x,y,z,i),f_{dg}(i,e,f)).$$

It can readily be seen that $M\text{-}Compose$ is the desired one.

∎

<u>Lemma 5.1.16</u>

Let $(\underline{K},\kappa,\partial)$, $(\underline{K}',\kappa',\partial')$ and $(\underline{K}'',\kappa'',\partial'')$ be effective categories. There are recursive functions $f\text{-}left$ and $f\text{-}right$ s.t. if $\langle x,y\rangle$ is a functor index of $F:(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')\to(\underline{K}'',\kappa'',\partial'')$, then $f\text{-}left(a,x,y)$ and $f\text{-}right(b',x,y)$ are functor indices of $F(\kappa(a),-):(\underline{K}',\kappa',\partial')\to(\underline{K}''.\kappa'',\partial'')$ and $F(-,\kappa'(b)):(\underline{K},\kappa,\partial)\to(\underline{K}'',\kappa'',\partial'')$ respectively. Also there are recursive functions $cf\text{-}letf$ and $cf\text{-}right$ s.t. if F is an effectively continuous functor with a continuity index $\langle\langle x,y\rangle,z\rangle$ then $cf\text{-}left(a,x,y,z)$ and $cf\text{-}right(b,x,y,z)$ are continuity indices of $F(\kappa(a),-)$ and $F(-,\kappa'(b))$ respectively.

<u>proof</u> We prove only for $F(\kappa(a),-)$. Similar proof works for $F(-,\kappa'(b))$. Define h to be a recursive function satisfying:

$$\phi_{h(m,n,k)}(a')=\phi_n(\langle m,a'\rangle).$$

Also define g to be a recursive function satisfying:

$$\phi_{g(n,m,k)}(a',c',d)=\phi_k(\langle m,a'\rangle,\langle m,c'\rangle,\langle Idt(m),d\rangle)$$

Notice that we have:

$$F(\kappa(a),-)(\kappa'(a'))=F(\kappa(a),\kappa'(a'))=\kappa''(\phi_x(\langle a,a'\rangle)),$$

$$F(\kappa(a),-)(\partial'(a',c')(d))$$
$$=F(\partial(a,a)(Idt(a)),\partial'(a',c')(d))$$
$$(\phi_y(\langle a,a'\rangle,\langle a,c'\rangle,\langle Idt(a),d\rangle)).$$

Therefore *f-left* defined by:

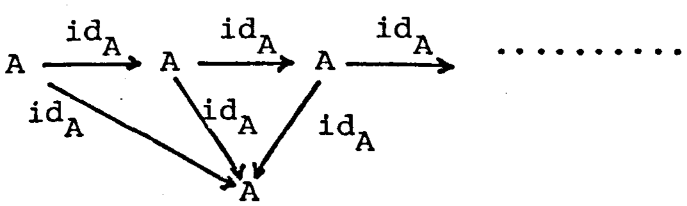
$$f\text{-}left(m,n,k)=\langle h(m,n,k),g(m,n,k)\rangle$$

is the desired recursive function, for we have:

$$\phi_{h(a,x,y)}(a')=\phi_x(\langle a,a'\rangle),$$

$$\phi_{g(a,x,y)}(a',c',d)=\phi_y(\langle a,a'\rangle,\langle a,c'\rangle,\langle Idt(a),d\rangle)).$$

Let $(G',i)=(\langle A_n',f_n'\rangle,i)$ be an indexed ω-codiagram in $(\underline{K}',\kappa',\partial')$. Let μ be an effective ω-colimiting cocone of (G',i) with a colimit index $\langle\langle a',k'\rangle,u'\rangle$. Then $((\langle(A,A_n')\rangle,(id_A,f_n')),t(i))$ is an indexed ω-codiagram for some recursive function t, where $A=\kappa(a)$. Notice that:



is an effective ω-colimiting cocone in $(\underline{K},\kappa,\partial)$ with a colimit index $\langle\langle a,Idt(a)\rangle,m\rangle$ where $\phi_m(c,e)=\phi_e(0)$. Therefore $\langle(id_A,\mu_n)\rangle$ is an effective ω-colimiting cocone of $(\langle(A,A_n'),(id_A,f_n')\rangle,t(i))$ with a colimit index $\langle\langle\langle a,a'\rangle,pr(k')\rangle,r(m,u')\rangle$ where pr and r

are recursive functions satisfying:

$$\phi_{pr(u)}(n)=\langle Idt(a),\phi_u(n)\rangle,$$

$$\phi_{r(u,v)}(c,e,b',e')=\langle\phi_{u(c,e)},\phi_{v(b',e')}\rangle.$$

Since F has a continuity index $\langle\langle x,y\rangle,z\rangle$, $F(\kappa(a),-)$ has a colimit index:

$$f_{cocone(F(\kappa(a),-))}(\pi_1\cdot f\text{-}left(a,x,y),\pi_2\cdot f\text{-}left(a,x,y),\langle a,a'\rangle,k'),$$
$$\phi_z(\langle a,a'\rangle,pr(k'),r(m,u'),t(i))\rangle.$$

Define *cf-left* by:

$$cf\text{-}left(c,d,e,j)=\langle f\text{-}left(c,d,e),s(c,d,e,j)\rangle$$

where s is a recursive function satisfying:

$$\phi_{s(c,d,e,j)}(a',k',u',i)=\phi_z(\langle c,a'\rangle,pr(k'),r(m,u'),t(i)).$$

It is now obvious that *cf-left* is the desired one.

$\square$

Notice that in the above proof, we have used the effectiveness of the identity morphism as in 5.1.1.

Example 5.1.17

Given effective domains (X_1,ε_1), (X_2,ε_2) and (X_3,ε_3), $(X_1,\varepsilon_1)\times(X_2,\varepsilon_2)$ is an effective domain as established in chapter 3. Also if $f:X_1\times X_2\to X_3$ is f-computable, then it is f-computable in both the first and second arguments. Furthermore the process of obtaining $f(x,-)$ and $f(-,y)$ is recursive in directed indices of x,y and f, uniformly in the domain and codomain of f. $\square$

Lemma 5.1.18

Let $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$ be effective categories. There is a recursive function *f-dup* s.t. if $\langle a,b\rangle$ is a functor index of $F:\underline{K}\times\underline{K}\to\underline{K}'$ then *f-dup*(a,b) is a functor index of $\lambda S.F(S,S):\underline{K}\to\underline{K}'$ defined by:

$\lambda S.F(S,S)(A)=F(A,A)$,

$\lambda S.F(S,S)(f)=F(f,f)$.

Also there is a recursive function *cf-dup* s.t. if $\langle\langle a,b\rangle,c\rangle$ is a continuity index of F, then *cf-dup*(a,b,c) is a continuity index of $\lambda S.F(S,S)$.

<u>proof</u> Let $f\text{-}dup_1$ be a recursive function satisfying:

$$f\text{-}dup_{1}(x,y)(z)=\phi_x(z,z).$$

Let $f\text{-}dup_2$ be a recursive function satisfying:

$$f\text{-}dup_{2}(x,y)(i,j,k)=\phi_y(\langle i,i\rangle,\langle j,j\rangle,\langle k,k\rangle).$$

Define *f-dup* by $f\text{-}dup(x,y)=\langle f\text{-}dup_1(x,y),f\text{-}dup_2(x,y)\rangle$.

It is obvious that this *f-dup* is the one desired. Let $(\langle D_n,f_n\rangle,i)$ be an indexed ω-codiagram in $(\underline{K},\kappa,\partial)$. Obviously $(\langle (D_n,D_n),(f_n,f_n)\rangle,p(i))$ is an indexed ω-codiagram in $(\underline{K},\kappa,\partial)$ for a recursive function p s.t.:

$$\phi_{p(j)}(x)=\langle\phi_i(x),\phi_i(x)\rangle.$$

Let $\mu:\langle D_n,f_n\rangle\to D$ be an effective ω-colimiting cocone with a colimit index $\langle\langle d,k\rangle,e\rangle$. Then (μ,μ) is an effective ω-colimiting cocne of $(\langle D_n,D_n),(f_n,f_n)\rangle,p(i))$ with a colimit index:

$$\langle\langle\langle d,d\rangle,p(k)\rangle,p'(e)\rangle$$

where p' is a recursive function satisfying:

$$\phi_{p'(j)}(x,y)=\langle\phi_j(x,y),\phi_j(x,y)\rangle.$$

Since F has a continuity index $\langle\langle a,b\rangle,c\rangle$, $\lambda S.F(S,S)(\mu)=F(\mu,\mu)$ has a colimit index:

$$\langle f_{cocone(F)}(a,b,d,k),\phi_c(\langle d,d\rangle,p(k),p'(e),p(i))\rangle .$$

Define *cf-dup* by:

$$cf\text{-}dup(x,y,z)=\langle f\text{-}dup(x,y),t(x,y,z)\rangle$$

where t is a recursive function satisfying:

$$\phi_{t(x,y,z)}(d,k,e,i)=\phi_z(\langle d,d\rangle,p(k),p'(e),p(i)).$$

It is obvious that this *cf-dup* is the one desired.

□

As we will observe later in 5.3.24 5.3.26, the domain constructors $\times,+,\to$ induce effectively continuous functors $\times^P,+^P,\to^P:(\underline{K},\kappa,\partial)^2\to(\underline{K},\kappa,\partial)$ where $(\underline{K},\kappa,\partial)$ is either EGD^P, ED^P, EGS^P, or ES^P. Also $F[\,]$ induces an effectively continuous functor $F[\,]^P:(\underline{K},\kappa,\partial)\to(\underline{K},\kappa,\partial)$ where $(\underline{K},\kappa,\partial)$ is either EGS^P or ES^P. Therefore we can effectively obtain effectively initial solutions of recursive domain equations which involve only $+^P,\times^P,\to^P$ as domain constructors, over EGD^P, ED^P, EGS^P and ES^P. Also we can effectively obtain effectively initial solutions of recursive domain equations which also involve $F[\,]^P$, over EGS^P and ES^P.

5.2 Category of Recursively Enumerable Sets

Between 5.1.1 and 5.1.2, we studied the category of r.e. sets and partially computable functions with two different indexing systems, namely REC1 and REC2, and observed that they are effective categories. In this section, we study a sub-category, namely the category of r.e. sets and (total) computable functions, as a non-domain model of effective ω-category. We also study even smaller category, namely the category of r.e. sets and inclusion maps, and observe that Kleene 1st recursion theorem of the enumeration operators is an instance of the effectively initial algebra theorem.

A partially computable function $f:W_i \to W_j$ is a computable function iff it is a total function.

Lemma 5.2.1

$f:W_i \to W_j$ is computable iff either

(1) $W_i = \emptyset$ or

(2) $W_i \neq \emptyset$, $W_j \neq \emptyset$ and there is a total recursive function $h:N \to N$ s.t. $f \cdot \phi_{dv2(i)} = \phi_{dv2(j)} \cdot h$.

proof (Sufficiencey) Obviously (1) implies f is total. Assume h is as in (2). Obviously $f(n) = \phi_{dv2(j)}(h(\mu m. \phi_{dv2(i)}(m) = n))$. Thus f is total, since $\phi_{dv2(i)}$ is an enumeration of W_i.
(Neccesity) Assume $W_i \neq \emptyset$ and $W_j = \emptyset$. Then f is not toatl. Now assume in (2), h is not total. Let $h(x)$ be undefined. Then $\phi_{dv2(j)} \cdot h(x)$ is undefined. But $f \cdot \phi_{dv2(i)}(x)$ is defined and $f \cdot \phi_{dv2(i)}(x) = \phi_{dv2(j)} \cdot h(x)$. Thus contradiction.

∎

Let REC1' and REC2' be the indexed categories obtained from REC1 and REC2 by restricting morphisms to computable functions. It can readily be seen that both REC1' and REC2' are effective categories, for the class of computable functions is closed under composition and the identity function on an no-empty r.e.set and the empty function with the empty domain are both computable.

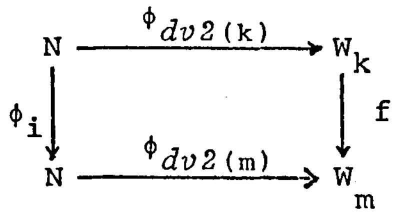
Lemma 5.2.2

The empty set is an effectively initial object in REC1, REC1',REC2 and REC2'. It also is an effectively final object in REC1 and REC2. A singleton is an effectively final object in REC1' and REC2'.

proof Let ϕ_i be a recursive function s.t. $\phi_i(y)=k$ where k is a natural number s.t. $W_k=\emptyset$. Then for every $j\in N$ s.t. $W_j=\emptyset$, $<i,j>$ is an initiality index of $\emptyset$ in REC1 and REC1'. Thus $\emptyset$ is an effectively initial object in REC1 and REC1'. By virtue of (2)-5.1.1.2, it also is effectively intial in both REC2 and REC2'. Let ϕ_i be a recursive function s.t. $\phi_i(y)=y$. Then for every $j\in N$ s.t. $W_j=\emptyset$, $\partial(y,j)(\phi_i(y))$ is the unique computable function from W_y to W_j in REC1. Indeed it is the empty function. But unless $W_y=\emptyset$, it is every where undefined. Thus $\emptyset$ has a finality index $<j,i>$ for every $j\in N$ s.t. $W_j=\emptyset$. Thus by virtue of (2)-5.1.1.2, $\emptyset$ is effectively final in both REC1 and REC2. Assume that W_m is a singleton. Obviously $W_m=\{\phi_{dv2(m)}(0)\}$. Let ϕ_i be a partial recursive function defined by:

$$\phi_i(k)=\phi_{dv2(m)}(0)+0\times\phi_{dv2(k)}(0).$$

It can readily be seen that the following diagram in REC1' commutes, regardless of if W_k is empty or not:



Thus W_m has a finality index $<m,i>$ in REC1'. By virtue of (2)-5.1.1.2, a singleton is an effectively final object in both REC1' and REC2'. ☒

Definition 5.2.3

Given r.e. sets W_i and W_j, define:

$$W_i \times W_j = \{<n,m> \mid n \in W_i, m \in W_j\}$$

$$W_i + W_j = \{<0,n> \mid n \in W_i\} \cup \{<1,m> \mid m \in W_j\}. \qquad \square$$

Lemma 5.2.4

$W_i \times W_j$ and $W_i + W_j$ are r.e.sets. Indeed there are recursive functions *Prod* and *Sum* s.t.:

$$W_i \times W_j = W_{Prod(i,j)}$$

$$W_i + W_j = W_{Sum(i,j)}.$$

proof $\lambda n.\lambda m.< \phi_i(n), \phi_j(m) > = \phi_i \times \phi_j$ is a partial recursive function whose range is $W_i \times W_j$. Thus *Prod* is a recursive function s.t.:

$$\phi_{Prod(i,j)} = \phi_i \times \phi_j.$$

Let *Sum* be a recursive function s.t. *Sum* (i,j) is an acceptable index of a program which, in parallel, enumerates W_i and W_j using $\phi_{dv2(i)}$ and $\phi_{dv2(j)}$, and which outputs $<0,n>$ once $\phi_{dv2(i)}$ generates n and $<1,m>$ once $\phi_{dv2(j)}$ generates m.

☒

Definition 5.2.5

Given an r.e. set W_i and a computable equivalence predicate $E: W_i \times W_i \to \{0,1\}$, there is a partial recursive function f_E s.t.

$$f_E(n) = <\phi_{dv2(i)}(\mu k \leq n.[\phi_{dv2(i)}(k) E \phi_{dv2(i)}(n)]), e>

where e is a c_1 index of E. Note that f_E is recursive if $W_i \neq \emptyset$, otherwise f_E is everywhere undefined. Also notice that if $W_i=\emptyset$ then $[\phi_{dv2(i)}(n)]=\emptyset$ and $W_i/E=\emptyset$. Thus $f_E(n)$ contains enough infromation to generate $[\phi_{dv2(i)}(n)]$. Thus we define:

$$W_i/E=\text{range}(f_E). \qquad \square$$

<u>Lemma 5.2.6</u>

There is a recursive function *Qut* s.t. if E is a computable equivalence predicate $E:W_i \times W_i \to \{0,1\}$ with a c_1-index e, then:

$$W_{Qut(i,e)}=W_i/E.$$

<u>proof</u> Construction of f_E is uniform in i and e.

$\square$

By virtue of (2)-5.1.1.2, in 5.2.5 and 5.2.6, we can take e as a c_2-index of E.

<u>Theorem 5.2.7</u>

REC1,REC1',REC2 and REC2' are effective binary product and effective binary coproduct closed.

<u>proof</u> Note that $W_i \times W_j = W_{Prod(i,j)}$. Define $\pi_1:W_i \times W_j \to W_i$ by $\pi_1(<m,n>)=m$ if $W_i \times W_j \neq \emptyset$, otherwise the empty function. Let $\overline{\pi}_1: N\times N \to N$ be a recursive function s.t. $\overline{\pi}_1(<m,n>)=m$. Obviously π_1 is the restriction to $W_i \times W_j$ of $\overline{\pi}_1$. Define $P2_1$ to be a recursive function s.t. $\phi_{P2_1(i,j)}=\overline{\pi}_1$. By virtue of (2)-5.1.1.2, we also have a recursive function $P1_1$ s.t. $P_{1_1}(i,j)$ is a c_1-index of π_1. Let $f:W_k \to W_i$ and $g:W_k \to W_i$ be partially computable function with c_2-index x and y respectively. Define $<f,g>:W_k \to W_i \times W_j$ by:

$$<f,g>(z)=<f(z),g(z)> \text{ if } W_k \neq \emptyset$$

otherwise the empty function. Let $<\overline{f},\overline{g}>:N^2 \to N: a \mapsto <\overline{f}(a),\overline{g}(a)>$ where $\overline{f}$ and $\overline{g}$ are partial recursive functions and f and g are the restriction to W_k of them, i.e. $\overline{f}=\phi_x$ and $\overline{g}=\phi_y$. Obviously

$<\overline{f},\overline{g}>$ is a partial recursive function and $<f,g>$ is the restriction to W_k of it. Thus $<f,g>$ is a partially computable function. Since the construction of $<\overline{f},\overline{g}>$ is uniform in $\overline{f}$ and $\overline{g}$, there is a recursive function h s.t. $h(i,j,k,x,y)$ is a c_2-index of $<f,g>$. By S-m-n theorem, there is a recursive function $P\text{-}Med2:N^2\to N$ s.t.:

$$\phi_{P\text{-}Med2(i,j)}(k,x,y)=\phi_{h(i,j,k,x,y)}=<\overline{f},\overline{g}>.$$

Thus $(W_{Prod(i,j)},\pi_1:W_{Prod(i,j)}\to W_i,\pi_2:W_{Prod(i,j)}\to W_j)$ is an effective product with a product index:

$$<<Prod(i,j),i,j,P2_1(i,j),P2_2(i,j)>,P\text{-}med2(i,j)> .$$

Note $P2_2$ is similarly defined, for π_2 in REC2, to $P2_1$ for π_2 in REC1. Thus REC2 is effectively binary product closed. By virtue of (2)-5.1.1.2, REC1 also is effective binary product closed. In the above proof, π_1 and π_2 are obviously total. It can readily be seen that $<f,g>$ is total. Thus both REC1' and REC2' are effective binary product closed. Effective binary coproduct of W_i and W_j is $W_{Sum(i,j)}$ with $i_1:W_i\to W_{Sum(i,j)}$ and $i_2:W_j\to W_{Sum(i,j)}$ s.t.

$i_1(n)=<0,n>$ if $W_i\neq\emptyset$ otherwise the empty function,

$i_2(n)=<1,n>$ if $W_j\neq\emptyset$ otherwise the empty function.

∅

Theorem 5.2.7

REC1' and REC2' are effectively ω-cocomplete.

proof Think of the follwoing effectively generable ω-codiagram G in REC2' with index k:

$$W_{g(0)}\xrightarrow{f_0}W_{g(1)}\xrightarrow{f_1}W_{g(2)}\longrightarrow\cdots\cdots$$

where $g=\pi_1\cdot\phi_k$ and $f_n=\partial(g(n),g(n+1))(\pi_2\cdot\phi_k(n))$. For $i<j$ let $f_{ij}=f_j\cdot\ldots\cdot f_i$. It can readily be seen that the ω-sum $\coprod_i W_{g(i)}$ of G is an r.e. set and for some recursive function *du*:

$$\coprod_i W_{g(i)}=W_{du(k)}.$$

Obviously $R=\{\langle\langle i,x\rangle,\langle i,y\rangle\rangle \mid f_{ij}(x)=y\}$ is a computable binary predicate over $\coprod_i W_{g(i)}$. Let E be the smallest equivalence relation over $\coprod_i W_{g(i)}$ containing R. It can readily be seen that E is a computable equivalence with a c_2-index $Equ(k)$ for some recursive function *Equ*. By 5.2.6, there is a recursive function ω-*Colim* s.t.:

$$(\coprod_i W_{g(i)})/E=W_{\omega\text{-}Colim(k)}.$$

Define $\lambda_n:W_{g(n)}\to W_{\omega\text{-}Colim(k)}$ by:

$$\lambda_n(x)=f_E(\mu m.\langle n,x\rangle=\phi_{dv2(\omega\text{-}Colim(k))}(m)) \quad \text{if } W_{\omega\text{-}Colim(k)}\neq\emptyset$$

otherwise the empty function.

Define $\overline{\lambda}_n:N\to N$ by:

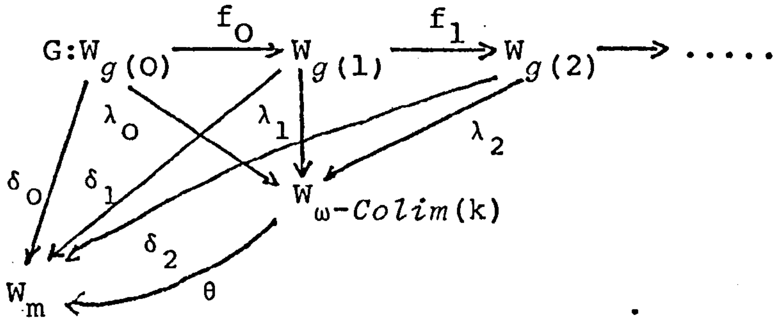
$$\overline{\lambda}_n(x)=f_E(\mu m.\langle n,x\rangle=\phi_{dv2(\omega\text{-}Colim(k))}(m)).$$

Obviously $\overline{\lambda}_n$ is a partial recursive function and λ_n is the restriction to $W_{g(n)}$ of it. Also λ_n is total. Thus λ_n is a computable function. Since the construction of $\overline{\lambda}_n$ is uniform in k, there is a recursive function ω-*Cocone* s.t. $\phi_{\omega\text{-}Cocone(k)}$ is recursive and $\phi_{\omega\text{-}Cocone(k)}(n)$ is a c_2-index of λ_n. Let $\langle\delta_n:W_{g(n)}\to W_m\rangle$ be an effectively generable ω-cocone of (G,k) with an index $\langle m,b\rangle$. Define $\theta:W_{\omega\text{-}Colim(k)}\to W_m$ by:

$$\theta(\langle\langle n,x\rangle,Equ(k)\rangle)=\delta_n(x) \quad \text{if } W_{\omega\text{-}Colim(k)}\neq\emptyset$$

otherwise the empty function.

It can readily be seen that θ is the unique total computable function which makes the following diagram commutes:



Define $\overline{\theta}:N\to N$ by:

$$\overline{\theta}(y)=\delta_{\pi_1\cdot\pi_1(y)}(\pi_2\cdot\pi_1(y))$$
$$=\phi_{\phi_b(\pi_1\cdot\pi_1(y))}(\pi_2\cdot\pi_1(y)).$$

Obviously $\overline{\theta}$ is partial recursive and θ is the restriction to $W_{\omega\text{-}Colim(k)}$ of it. Let ω-*Comed* be the following recursive function:

$$\phi_{\phi_{\omega\text{-}Comed(k)}(m,b)}(y)=\phi_{\phi_b(\pi_1\cdot\pi_1(y))}(\pi_2\cdot\pi_1(y)).$$

Thus we have established that REC2' is effectively ω-cocomplete. By virtue of 5.1.1.2, REC1' is also effectively ω-cocomplete.

◻

Corollary 5.2.9

Both REC1' and REC2' are effective ω-categories.

proof By 5.2.2 and 5.2.8. ◻

Let REC' be the category of r.e. sets and computable functions. Consider the following two functors:

$\times$:REC'$\times$REC'$\to$REC'	+:REC"$\times$REC'$\to$REC'
$\times$(W,W')=W$\times$W'	+(W,W')=W+W'
$\times$(f:W$\to$W',g:W'$\to$W")	+(f:W$\to$W',g:W'$\to$W")
=f$\times$g:W$\times$W'$\mapsto$W'$\times$W"	=f+g:W+W'$\mapsto$W'+W"
:<a,b>$\mapsto$<f(a),g(b)>	:<i,x>$\mapsto$if i=0 then <i,f(x)> else <i,g(x)>.

Lemma 5.2.10

+ and × defined above are effectively continuous functors w.r.t. both REC1' and REC2'.

proof Notice that an object in REC2'×REC2' with an index x is $(W_{\pi_1(x)}, W_{\pi_2(x)})$. Therefore $f_{ob(\times)}(x)=Prod(\pi_1(x),\pi_2(x))$. Also a morphism from $(W_{\pi_1(x)}, W_{\pi_2(x)})$ to $(W_{\pi_1(y)}, W_{\pi_2(y)})$ with an index i in REC2'×REC2' is:

$$(\partial(\pi_1(x),\pi_1(y))(\pi_1(i)),\partial(\pi_2(x),\pi_2(y))(\pi_2(i))$$

where ∂ is the morphism indexing of REC2'. Let r be a recursive function satisfying:

$$\phi_x \times \phi_y = \phi_{r(x,y)} : N \to N : \langle a,b \rangle \mapsto \langle \phi_x(a),\phi_y(b) \rangle .$$

Thus $f_{mr(\times)}(x,y,i)=r(\pi_1(i),\pi_2(i))$. Thus × is an effective functor from REC2'×REC2' to REC2'. Let <i,j> be a functor index of ×. Let (G,k) be an indexed ω-codiagram in REC2'×REC2'. Let μ be an effective ω-colimiting cocone of (G,k) with an index <<x,y>,z>. Obviously $\times(\mu)$ is an effectively generable ω-colimiting cocone of $(\times\cdot G, f_{dg}(i,j,k))$.

Notice that for any indexed ω-cocone $(\delta,\langle a,b\rangle)$ of $(\times\cdot G, f_{dg}(i,j,k))$, $\gamma=(\pi_1\cdot\delta,\pi_2\cdot\delta)=(\langle\pi_1\cdot\delta_n\rangle,\langle\pi_2\cdot\delta_m\rangle)$ is an effectively generable ω-cocone of (G,k) with an index $s(a,b)$ for some recursive function s. Obviously $\times(\gamma)=\delta$. Remember that $\phi_z(\pi_1\cdot s(a,b),\pi_2\cdot s(a,b))$ is an index of the mediating morphism θ from μ to γ. It can readily be seen that $\times(\theta)$ is the mediating morphism from $\times(\mu)$ to $\times(\gamma)=\mu$. Notice that $\times(\theta)$ has a c_2-index:

$$r(\pi_1\cdot\phi_z(\pi_1\cdot s(a,b),\pi_2\cdot s(a,b)),\pi_2\cdot\phi_z(\pi_1\cdot s(a,b),\pi_2\cdot s(a,b))).$$

Let h be a recursive function satisfying:

$$\phi_{h(z)}(a,b)=r(\pi_1\cdot\phi_z(\pi_1\cdot s(a,b),\pi_2\cdot s(a,b)),$$
$$\pi_2\cdot\phi_z(\pi_1\cdot s(a,b),\pi_2\cdot s(a,b))).$$

Then obviously $\times(\mu)$ has a colimiting index:

$$\langle f_{cocone}(i,j,x,y),h(z)\rangle.$$

Thus $\times$ is effectively continuous wrt REC2'. By virtue of 5.1.1.2, $\times$ is effectively continuous wrt REC1'. Similar proof establishes that + is effectively continuous wrt both REC1' and REC2'. ∅

As a familiar example of the effectively initial algebras, we can think of Kleene 1st recursion theorem on enumeration operators. Let REI1 and REI2 be the indexed categories obtained from REC1' and REC2' by restricting the morphisms to inclusion maps. It can readily be seen that both REC1 and REC2 are effective ω-categories. Let Φ_z be an enumeration operator (see Rogers[14]) of an index z. We can regard this as an effectively continuous functor REI1→REI1 by:

$$\Phi_z(W_i\subseteq W_j)=\Phi_z(W_i)\subseteq\Phi_z(W_j).$$

This obviously is well-defined since Φ_z is monotone wrt $\subseteq$. In fact there is a recursive function c s.t. $c(z)$ is a continuity index of Φ_z. The least fixed-point of Φ_z, whose existence guaranteed by the 1st recursion theorem is the effectively initial Φ_z-algebra. The same argument holds for REI2. In the next section, we will study another characterization of the Kleene 1st recursion theorem.

5.3 Effective O-categories

As for the case of non-constructive categories, a lot of concrete effective categories have some effective po structure on their hom sets. This leads to the following notion of effective O-category, which is an effective version of Wand [29] O-category.

Definition 5.3.1

An effective O-category is an effective category $(\underline{K},\kappa,\partial)$ satisfying:

(1) For each ordered pair (a,b) of object indices, $Hom(\kappa(a),\kappa(b))$ is a poset which is $\partial(a,b)$-effective complete. More specifically, given any non-empty chain $<f_n>$ in $Hom(\kappa(a),\kappa(b))$ s.t. for some recursive function c, $f_n=\partial(a,b)(c(n))$, we have $\sqcup f_n \in Hom(\kappa(a),\kappa(b))$. In case $\phi_j=c$, we say that this $\partial(a,b)$-effective chain $<f_n>$ has a chain index j. Furthermore, there is a recursive function *Lub* s.t. if $<f_n>$ has a chain index j, then

$$\sqcup f_n=\partial(a,b)(Lub(a,b,j)).$$

(2) The composition of morphisms is effectively continuous, i.e., if $<f_i>$ and $<g_i>$ are $\partial(a,b)$-effective chain and $\partial(b,c)$-effective chain respectively, then we have:

$$(\sqcup g_i)\cdot(\sqcup f_i)=\sqcup g_i\cdot f_i$$

where $g_i\cdot f_i$ obviously is a $\partial(a,c)$-effective chain, since $(\underline{K},\kappa,\partial)$ is an effective category.

It can readily be seen that $EGD^{(*)}$, $ED^{(*)}$, $EGS^{(*)}$ and $ES^{(*)}$ are all effective O-categories. To observe this, remember 2.1.11, 3.1.2, etc. It is at least worth while to mention that REC1 is an effective O-category with the extensional ordering as the partial ordering on hom sets.

To observe this, notice that if $f,g:W_i \to W_j$ are partially computable and have c_1-indices n and m respectively, then:

$$f \sqsubseteq g \quad \text{iff } \phi_n \sqsubseteq \phi_m.$$

Furthermore if $\langle f_n : W_i \to W_j \rangle$ is a $\partial(i,j)$-effective chain with a chain index k then:

$$f_n = \partial(i,j)(a) \quad \text{where } \phi_a = \sqcup \phi_{\phi_k}(n).$$

But there is a recursive function *Lub* s.t.:

$$\sqcup \phi_{Lub(i,j,k)} = \sqcup \phi_{\phi_k}(n) = \sqcup \phi_{\phi_k}(n).$$

Definition 5.2.2

Let $(\underline{K}, \kappa, \partial)$ be an effective O-category. A pair $(f:A \to B, g:B \to A)$ of $\underline{K}$ morphisms is a projection pair from A to B iff $f \cdot g \sqsubseteq id_B$ and $g \cdot f = id_A$. We call g the projection (of this pair) anf f the embedding (of this pair). □

It can readily be seen that an embedding f (or a projection g) uniquely determines a morphism f^R (or g^L) s.t. (f,f^R) or (g^L,g)) is a projection pair. We will call f^R the (right) adjoint of f and g^L the (left) adjoint of g.

One might expect some kind of effectiveness constraint in the definition of projection pairs. For example, one might ask what if we make the unique correspondence of an embedding and its adjoint effective. More precisely, ask the existence of recursive functions *Lad* and *Rad* s.t. if $f=\partial(i,j)(n)$ is an embedding then $f^R=\partial(i,j)(Rad(i,j,n))$ and if $g=\partial(i,j)(m)$ is a projection, then $g^L=\partial(j,i)(Lad(i,j,m))$. But in our main model of effective O-category, which is the category of effectively given domains and computable functions with acceptable indexing for the object indexing and the directed indexings for the morphism indexings, the existence of *Rad* and *Lad* is doubtful.

Definition 5.3.3

Given an effective O-category $(\underline{K},\kappa,\partial)$, the category of projection pairs of $(\underline{K},\kappa,\partial)$, in symbols $(\underline{K},\kappa,\partial)^P$ is a triple $(\underline{K}^P,\kappa^P,\partial^P)$ where $\underline{K}^P$ is as in 1.4.10, $\kappa=\kappa^P$ and if $(\partial(a,b)(i),\partial(b,a)(j))$ is a projection pair then:

$$\partial^P(a,b)(<i,j>)=(\partial(a,b)(i),\partial(b,a)(j)).$$

We will denote $Hom_{\underline{K}^P}(A,B)$ and $id_A^{\underline{K}^P}$ by $Hom^P(A,B)$ and id_A^P respectively. □

It can readily be seen that $(\underline{K},\kappa,\partial)^P$ is an effective category.

Definition 5.3.4

We say an effective O-category $(\underline{K},\kappa,\partial)$ is effectively empty chain complete iff

(1) Hom(A,B) has the least element $\bot_{A,B}$ and there is a recursive function *Bottom* s.t.:

$$\bot_{\kappa(a),\kappa(b)}=\partial(a,b)(Bottom(a,b)).$$

(2) $\bot_{B,C}\cdot f=\bot_{A,C}$ for every $f:A\to B$. □

Notice that (1) is concerned with the effective existence of the lub of the empty chain, and (2) is concerned with the effective continuity of the composition wrt the empty chain. To characterise the effective continuity of the composition wrt the empty chain, we need one more condition, which we will discuss later between 5.3.12 and 5.3.14.

<u>Lemma 5.3.5</u>

If $(\underline{K},\kappa,\partial)$ is an effectively empty chain complete effective O-category, then the final object $\perp$, if any, is effectively initial in $(\underline{K},\kappa,\partial)^P$.

<u>proof</u> Let $\kappa(i)=\perp$. If $f,f':\perp\to\kappa(a)$ are embeddings then they are the same, for they have the same right adjoint:

$$\partial(a,i)(\mathit{Bottom}(a,i))=\perp_{\kappa(a),\perp}$$

for $\perp$ is the final object. It can readily be seen that $\perp_{\perp,\kappa(a)}$ is the embedding with the right adjoint $\perp_{\kappa(a),\perp}$. But we have:

$$\perp_{\perp,\kappa(a)}=\partial(i,a)(\mathit{Bottom}(i,a)).$$

Thus $\perp$ is effectively initial in $(\underline{K},\kappa,\partial)^P$, for it has an initiality index $\langle i,j\rangle$ where:

$$\phi_j(a)=\langle \mathit{Bottom}(i,a),\mathit{Bottom}(a,i)\rangle. \qquad \square$$

Let $(\underline{K},\kappa,\partial)$ be an effective O-category, then there are recursive functions pr and em s.t. if $(\Delta=\langle A_n,(f_n,f_n^R)\rangle,i)$ is an indexed ω-codiagram in $(\underline{K},\kappa,\partial)^P$ then:

(1) $(\Delta^P=\langle A_n,f_n^R\rangle,pr(i))$ is an indexed ω-diagram in $(\underline{K},\kappa,\partial)$.

(2) $(\Delta^E=\langle A_n,f_n\rangle,em(i))$ is an indexed ω-codiagram in $(\underline{K},\kappa,\partial)$.

<u>Lemma 5.3.6</u>

Let $(\underline{K},\kappa,\partial)$ be an effective O-category and $(\Delta=\langle A_n,(f_n,f_n^R)\rangle,i)$ be an indexed ω-codiagram in $(\underline{K},\kappa,\partial)^P$. If $\nu=\langle\nu_n:A\to A_n\rangle$ is an effective ω-limiting cone of $(\Delta^P,pr(i))$, with an index $\langle\langle a,k\rangle,x\rangle$, then ν_n is a projection for each n and $\langle\nu_n^L\cdot\nu_n\rangle$ is a $\partial(a,a)$-effective chain s.t. $id_A=\nu_n^L\cdot\nu_n$. In fact there is a recursive function Lc s.t. $Lc(i,a,k,x)$ is a chain index of $\langle\nu_n^L\cdot\nu_n\rangle$.

<u>proof</u> Define $(f_{mn},f_{mn}^R)=(f_{n-1},f_{n-1}^R)\cdot\ldots\cdot(f_m,f_m^R):A_m\to A_n$ for $m\le n$.

We use a convention $(f_{mm},f^R_{mm})=(f_m,f^R_m)$. For each $A_m=\kappa(\pi_1\cdot\phi_i(m))$ $=\kappa(\pi_1\cdot\phi_{pr(i)}(m))$, define a cone $\nu^{(m)}=<\nu_n^{(m)}:A_m\to A_n>$ of Δ^P by:

$$\nu_n^{(m)}=\underline{\text{if}}\ m\leq n\ \underline{\text{then}}\ f_{mn}\ \underline{\text{else}}\ f^R_{mn}.$$

It can readily be seen that there is a recursive function r s.t. $(\nu^{(m)},<\pi_1\cdot\phi_i(m),r(i,m)>)$ is an indexed ω-cone of $(\Delta^P,pr(i))$. Since ν is an effective ω-limiting cone of $(\Delta^P,pr(i))$ with an index $<<a,k>,x>$, there is a unique morphism $\theta_m:A_m\to A$ s.t. $\nu_n\cdot\theta_m=\nu_n^{(m)}$ for each n. In fact

$$\theta_m=\partial(\pi_1\cdot\phi_i(m),a)(\phi_x(\pi_1\cdot\phi_i(m),r(i,m))).$$

It can readily be seen that $\theta_m\cdot\nu_m\sqsubseteq\theta_{m+1}\cdot\nu_{m+1}$.
Since $\nu_n=\partial(a,\pi_1\cdot\phi_i(n))(\phi_k(n))$, we have:

$$\begin{aligned}\theta_m\cdot\nu_m&=\partial(\pi_1\cdot\phi_i(m),a)(\phi_x(\pi_1\cdot\phi_i(m),r(i,m))\\&\quad\ \partial(a,\pi_1\cdot\phi_i(n))(\phi_k(n))\\&=\partial(a,a)\\&\qquad(\partial\text{-}Compose(\pi_1\cdot\phi_i(m),a,\pi_1\cdot\phi_i(m),\\&\qquad\qquad\phi_x(\pi_1\cdot\phi_i(m),r(i,m)),\phi_k(m))).\end{aligned}$$

Thus $<\theta_m\cdot\nu_m>$ is a $\partial(a,a)$-effective chain with a chain index $Lc(i,a,k,x)$ where Lc is a recursive function satisfying:

$$\phi_{Lc(i,a,k,x)}(m)=\partial\text{-}Compose(\pi_1\cdot\phi_i(m),a,\pi_1\cdot\phi_i(m),\ \phi_x(\pi_1\cdot\phi_i(m),r(i,m)),\phi_k(m)).$$

It can readily be seen that $\sqcup\theta_m\cdot\nu_m=id_{\kappa(a)}$ and (θ_m,ν_m) is a projection pair. ∅

<u>Lemma 5.3.7</u>

Let $(\underline{K},\kappa,\partial)$ be an effective O-category and $(\Delta=<A_n,(f_n,f^R_n)>,$ $i)$ be an indexed ω-codiagram in $(\underline{K},\kappa,\partial)^P$. Assume that (μ,μ^R) $=<(\mu_n,\mu^R_n):A_n\to A>$ is an effectively generable ω-cocone with an index $<a,k>$ of (Δ,i) s.t. $<\mu_n\cdot\mu^R_n>$ is a ($\partial(a,a)$-effective) chain

and $\sqcup\mu_n\cdot\mu_n^R=id_A$. There are recursive functions u and v s.t.:
(1) (μ,μ^R) is an effective ω-colimiting cocone of (Δ,i) with an index $\langle\langle a,k\rangle,u(a,k,i)\rangle$.
(2) μ is an effective ω-colilmiting cocone of $(\Delta^E,em(i))$ with an index $\langle\langle a,lt(k)\rangle,v(a,k,i)\rangle$, where lt is as in the proof of 5.1.14.

<u>proof</u> Let $(\mu'=\langle\mu_n':A_n\to A\ ,\langle a',k'\rangle)$ be an indexed ω-cocone of $(\Delta^E,em(i))$ in $(\underline{K},\kappa,\partial)$. It can readily be seen that $\langle\mu_n'\cdot\mu_n^R\rangle$ is a $\partial(a,a')$-effective chain with a chain index $h(a,k,a'k',i)$ for some recursive function h. In fact:

$$\phi_{h(a,k,a',k',i)}(n)=\partial\text{-}Compose(a,\pi_1\cdot\phi_i(n),a',\pi_2\cdot\pi_2\cdot\phi_i(n),\pi_2\cdot\phi_{em(i)}(n)).$$

Let $\theta=\sqcup\mu_n'\cdot\mu_n^R$. It can readily be seen that θ is the unique morphism which mediates from μ to μ'. By the first axiom of the effective O-category, we have:

$$\theta=\partial(a,a')(Lub(a,a',h(a,k,a',k',i)).$$

Let u be a recursive function satisfying:

$$\phi_{u(a,k,i)}(a',k')=Lub(a,a',h(a,k,a',k',i)).$$

Then μ is an effective ω-colimiting cococne of $(\Delta^E,em(i))$ with an index $\langle\langle a,lt(k)\rangle,u(a,k,i)\rangle$.

For (μ,μ^R), assume that $((\mu',\mu'^R):\Delta\to A',\langle a',k'\rangle)$ is an indexed ω-cocone of (Δ,i). Then $(\mu'=\langle\mu_n':A_n\to A'\rangle,\langle a',lt(k')\rangle)$ is an indexed ω-cocone of $(\Delta^E,em(i))$. Thus

$$\theta=\partial(a,a')(\phi_{u(a,k,i)}(a',lt(k')))$$

is the unique morphism mediating from μ to μ'. Now $\langle\mu_n\cdot\mu_n'^R\rangle$ can readily be seen to be a $\partial(a',a)$-effective chain with a chain index $g(a',k',a,k,i)$ for some recursive function g. Let $\theta^R=\sqcup\mu_n\cdot\mu_n'^R$. Then we have:

$\theta^R = \partial(a',a)(Lub(a',a,g(a',k',a,k,i)).$

It can readily be seen that (θ,θ^R) is a projection pair. Now let v be a recursive function s.t.:

$\phi_{v(a,k,i)}(a',k') = \langle \phi_{u(a,k,i)}(a',lt(k')), Lub(a',a,g(a',k',a,k,i))\rangle.$

Thus (μ,μ^R) is an effective ω-colimiting cocone of (Δ,i) with an index $\langle\langle a,k\rangle, v(a,k,i)\rangle$. ∎

The following effective version of the limit and colimit coincidence gives us a sufficient condition for an effective O-category to yield an effectively ω-cocomplete category.

Definition 5.3.8

An effective O-category $(\underline{K},\kappa,\partial)$ has the effective S-property iff for every indexed ω-codiagram (Δ,i) in $(\underline{K},\kappa,\partial)^P$, there exists an indexed ω-cocone $((\mu,\mu^R),\langle a,k\rangle)$ of (Δ,i) s.t. μ^R is an effective ω-colimiting cone of an indexed ω-diagram $(\Delta^F, pr(i))$ in $(\underline{K},\kappa,\partial)$, with an index $\langle\langle a,rt(k)\rangle,x\rangle$ for some x. $(((\mu,\mu^R),\langle a,k\rangle),x)$ is called an effective S-cocone of (Δ,i). Such $(\underline{K},\kappa,\partial)$ has the effective S-complete property iff there are recursive functions ω-*lim*, ω-*cocone* and ω-*med* s.t. for every ω-codiagram (Δ,i) in $(\underline{K},\kappa,\partial)^P$,

$$(((\mu,\mu^R),\langle \omega\text{-}lim(i),\omega\text{-}cocone(i)\rangle),\omega\text{-}med(i))$$

is an effective S-cocone of (Δ,i).

□

Theorem 5.3.9

Let $(\underline{K},\kappa,\partial)$ be an effective O-category with the effective S-property. Then for every indexed ω-codiagram (Δ,i) of $(\underline{K},\kappa,\partial)^P$, if $(((\mu,\mu^R),\langle a,k\rangle),a)$ is an effective S-cocone of (Δ,i) then (μ,μ^R) is an effective ω-colimiting cocne with an index

$<<a,k>,v(a,k,i)>$. Also $\mu:\Delta^E \to \kappa(a)$ is an effective ω-colimiting cocone of $(\Delta^E, em(i))$ with an index $<<a,lt(k)>,u(a,k,i)>$. Note u and v are recursive functions defined in the proof of 5.3.7.

proof $(\Delta^P, pr(i))$ is an indexed ω-codiagram in $(\underline{K},\kappa,\partial)$. Since $(((\mu,\mu^R),<a,k>),x)$ is an effective S-cocone of (Δ,i), $\mu^R:\kappa(a)\to\Delta^P$ is an effective ω-limiting cone of $(\Delta^P, pr(i))$ in $(\underline{K},\kappa,\partial)$ with an index $<<a,rt(k)>,x>$. By 5.3.6, μ_n^R is a projection for each n, and $<\mu_n\cdot\mu_n^R>$ is a $\partial(a,a)$-effective chain with a chain index $Lc(i,a,rt(k),x)$ s.t. $id_{\kappa(a)}=\sqcup\mu_n\cdot\mu_n^R$. thus by 5.3.7, (μ,μ^R) is an effective ω-colimiting cocone of (Δ,i) with an index $<<a,k>,v(a,k,i)>$, and μ is an effective ω-colimiting cocone of $(\Delta^E,em(i))$ in $(\underline{K},\kappa,\partial)$ with an index $<<a,lt(k)>,u(a,k,i)>$.

∅

Notice that this theorem indicates the limit and colimit coincidence, for μ^R is an effective ω-limiting cone. The next theorem is an even stronger reflection of the limit and colimit coincidence in effective O-categories with effective S-property.

Theorem 5.3.10

Let $(\underline{K},\kappa,\partial)$ be an effective O-category with the effective S-property and $(\Delta=<A_n,(f_n,f_n^R)>,i)$ be an indexed ω-codiagram in $(\underline{K},\kappa,\partial)^P$. For an indexed ω-cocone $((\nu,\nu^R):\Delta\to\kappa(a),<a,k>)$ of (Δ, i), the following satatements are all equivalent:

(1) (ν,ν^R) is an effective ω-colimiting cocone of (Δ,i) with an index $<<a,k>,j>$ for some $j\in N$.

(2) ν is an effective ω-colimiting cocone of $(\Delta^E,em(i))$ with an index $<<a,lt(k)>,j>$ for some $j\in N$.

(3) $<\nu_n\cdot\nu_n^R>$ is a $\partial(a,a)$-effective chain s.t.:

$$id_{\kappa(a)}=\sqcup\nu_n\cdot\nu_n^R.$$

(4) ν^R is an effective ω-limiting cone of $(\Delta^P, pr(i))$ with an index $\langle\langle a, rt(k)\rangle, j\rangle$ for some $j \in N$.

<u>proof</u> By 5.3.6 (4) implies (3). By 5.3.7, (3) implies (1) and (2). Now assume (1). By the effective S-property we have an effective S-cocone $(((\mu,\mu^R),\langle x,y\rangle),z)$ of the indexed ω-codiagram (Δ,i) in $(\underline{K},\kappa,\partial)^P$. As we proved in the previous theorem, (μ,μ^R) satisfies (1),(2),(3) and (4). By the universality, we have: $\kappa(x)\cong\kappa(a)$ via $(\theta,\theta^R):\kappa(x)\to\kappa(a)$ where

$$\theta=\partial(x,a)(\phi_{v(x,y,i)}(a,k)).$$

Also $\nu_n=\theta\cdot\mu_n$ for each n. Therefore we have:

$$\begin{aligned}\sqcup\nu_n\cdot\nu_n^R &= \sqcup\theta\cdot\mu_n\cdot\mu_n^R\cdot\theta^R\\ &=\theta\cdot(\sqcup\mu_n\cdot\mu_n^R)\cdot\theta^R\\ &=\theta\cdot id_{\kappa(a)}\cdot\theta^R\\ &=id_{\kappa(a)}.\end{aligned}$$

Therefore (1) implies (3), thus by 5.3.7, (2). Notice μ^R is an effective ω-limiting cone of $(\Delta^P, pr(i))$. Also $\kappa(a)\cong\kappa(x)$ via $(\theta,\theta^R):\kappa(x)\to\kappa(a)$. Therefore by the argument between 5.1.3 and 5.1.4, ν^R is an effective ω-limiting cone of $(\Delta^P, pr(i))$ with an index $\langle\langle a, rt(x)\rangle, j\rangle$ for some $j\in N$. Thus (1) implies (4). Assume (2). By universality $\kappa(x)\cong\kappa(a)$, thus by the argument between 5.1.3 and 5.1.4, (ν,ν^R) is an effective ω-colimiting cocone of (Δ,i). Thus (2) implies (1).

◻

<u>Definition 5.3.11</u>

An effectively empty chain complete effective O-category with the final object and the effective S-property is an <u>effective Dom-category</u>. ◻

Theorem 5.3.12

If $(\underline{K},\kappa,\partial)$ is an effective Dom-category then $(\underline{K},\kappa,\partial)^P$ is an effective ω-category.

proof By 5.3.9 and 5.3.5. ∎

In the above, we have studied when an effective O-category induces an effective ω-coategory as its projection pair category. As we will see later in 5.3.23, $EGD^{(*)}$, $ED^{(*)}$, $EGS^{(*)}$ and $ES^{(*)}$ are all effective Dom-categories and:

$$(EGD)^P=(EGD^*)^P=EGD^P$$
$$(ED)^P=(ED^*)^P=ED^P$$
$$(EGS)^P=(EGS^*)^P=EGS^P$$
$$(ES)^P=(ES^*)^P=ES^P.$$

This indicates that we have properly categorized the process of restricting morphisms of effective O-categories to projection pairs. It should be noted that both REC1 and REC2 are effective Dom-categories and so, $(REC1)^P$ and $(REC2)^P$ are effective ω-categories. Here the morphisms are inclusion maps with the partial identity maps as the adjoints. It can readily be seen that $(REC1)^P$ and $(REC2)^P$ behave samely as REI1 and REI2 respectively, as effective ω-categories.

So far, the only known improtant example of an effective ω-category which can not "naturally" be obtained from an effective Dom-category is an effective domain as a category together with the directed indexing as the object indexing and the trivial indexings as the morphism indexings.

As in case of non-effective O-categories, surveyed in 1.4, the empty chain completeness condition of 5.3.4 omitted the right half of the effective continuity of the composition wrt the lub of the empty chain, more specifically:

$$f\cdot\bot_{A,B}=\bot_{A,C} \text{ for all } f:B\to C.$$

This condition has the effect of restrciting morphisms to strict maps. Given an effective Dom-category $(\underline{K},\kappa,\partial)$, let $(\underline{K},\kappa,\partial)^*$ be the effective Dom-category obtained from $(\underline{K},\kappa,\partial)$ by restricting morphisms to those satisfying above condition.

Lemma 5.3.13

Let $(\underline{K},\kappa,\partial)$ be an effective Dom-category. Then the final object $\bot$ is an effectively initial object in $(\underline{K},\kappa,\partial)^*$.

proof Let $\kappa(i)=\bot$. By the same argument as in 1.4.19, $\bot_{\bot,A}:\bot\to A$ is the unique morphism from $\bot$ to A in $(\underline{K},\kappa,\partial)^*$. But $\bot_{\bot,A}=\partial(i,a)(\mathit{Bottom}(i,a))$ where $A=\kappa(a)$. Therefore $\bot$ has an initiality index $\langle i,j\rangle$ in $(\underline{K},\kappa,\partial)^*$ where $\phi_j(a)=\mathit{Bottom}(i,a)$.

☒

Lemma 5.3.14

If $(\underline{K},\kappa,\partial)=(\underline{K}',\kappa'\partial')^*$ for some effective Dom-category $(\underline{K}',\kappa',\partial')$ then $(\underline{K},\kappa,\partial)^*=(\underline{K},\kappa,\partial)$. ☒

Definition 5.3.15

We say that an effective O-category $(\underline{K},\kappa,\partial)$ has locally determined effective ω-colimits of embeddings iff for every indexed ω-cocone $((\mu,\mu^R):\Delta\to\kappa(a),\langle a,k\rangle)$ of an indexed ω-codiagram (Δ,i) in $(\underline{K},\kappa,\partial)^P$ the following statements are equal:

(1) (μ,μ^R) is an effective ω-colimiting cocone of (Δ,i) with an index $\langle\langle a,k\rangle,j\rangle$.

(2) $\langle\mu_n\cdot\mu_n^R\rangle$ is a $\partial(a,a)$-effective chain s.t. $id_{\kappa(a)}=\sqcup\mu_n\cdot\mu_n^R$.

□

Corollary 5.3.16

Every effective O-category with the effective S-property has locally determined effective ω-colimits of embeddings.

proof By 5.3.7 and the proof of 5.3.9. ☒

Lemma 5.3.17

Let $(\underline{K},\kappa,\partial)$ and $(\underline{K}',\kappa',\partial')$ be effective O-categories. Then we have:

(1) $(\underline{K},\kappa,\partial)^{OP}=(\underline{K}^{OP},\kappa,\partial^{OP})$ where $\partial^{OP}(a,b)(m)=\partial(b,a)(m)$ is an effective O-category.

(2) $(\underline{K},\kappa,\partial)\times(\underline{K}',\kappa',\partial')$ is an effective O-category.

For the proof of this lemma, remember the partial orderings of $\underline{K}^{OP}$ and $\underline{K}\times\underline{K}'$ as in the remark immediately after 1.4.7.

☒

We consider throughout the rest of this section, three effective O-categories $(\underline{K},\kappa,\partial)$, $(\underline{L},\eta,\psi)$ and $(\underline{M},\tau,\zeta)$, and covariant functors $T:\underline{K}^{OP}\times\underline{L}\to\underline{M}$. The reason for doing this is to cope with the function space functor which is contravariant on the 1st argument. This restriction does not harm the generality of arguments, for a pure covariant functor can be obtained by taking $\underline{K}$ to be the one point category and a contravariant one can be obtained by taking $\underline{L}$ to be the one point category. For details see Plotkin & Smyth [24].

Definition 5.3.18

A functor $T:\underline{K}^{OP}\times\underline{L}\to\underline{M}$ is locally effectively monotone wrt $((\kappa\times\eta,\tau),(\partial^{OP}\times\psi,\zeta))$ iff T is effective wrt $((\kappa\times\eta,\tau),(\partial^{OP}\times\psi,\zeta))$ and for $f,f':A\to B$ in $\underline{K}^{OP}$ and $g,g':C\to D$ in $\underline{L}$, $f\sqsubseteq f'$ & $g\sqsubseteq g'$ implies $T(f,g)\sqsubseteq T(f',g')$ □

Lemma 5.3.19

There is a recursive function Pm s.t. if $T:\underline{K}^{op}\times\underline{L}\to\underline{M}$ is a locally effectively monotone functor with an index $\langle x,y\rangle$, then a functor $T^P:(\underline{K},\kappa,\partial)^P\times(\underline{L},\eta,\psi)^P\to(\underline{M},\tau,\zeta)^P$ defined by:

$$T^P(A,B)=T(A,B)$$

$$T^P((f,f^R),(g,g^R))=(T(f^R,g),T(f,g^R))$$

is effective wrt $((\kappa\times\eta,\tau),(\partial^P\times\psi^P,\zeta^P))$ and has an index $Pm(\langle x,y\rangle)$. We will write $T^P((f,f^R),(g,g^R))_1$ for $T(f^R,g)$ and $T^P((f,f^R),(g,g^R))_2$ for $T(f,g^R)$.

proof Obvious. ☐

Definition 5.3.20

A functor $T:\underline{K}^{op}\times\underline{L}\to\underline{M}$ is locally effectively continuous iff it is effectively continuous on morphism sets, more specifically if $\langle f_n\rangle$ is a $\partial^{op}(a,b)$-effective chain and $\langle g_n\rangle$ is a $\partial(c,d)$-effective chain then:

$$T(\sqcup f_n,\sqcup g_n)=\sqcup T(f_n,g_n).$$

Notice that $T(f_n,g_n)$ is a $\tau(f_{ob(T)}(\langle a,c\rangle),f_{ob(T)}(\langle b,d\rangle))$-effective chain for T is effective. Thus $T(\sqcup f_n,\sqcup g_n)$ is well-defined. ☐

Now the following theorem tells when an effective functor from an effective O-category to another induces an effectively continuous functor via the construction of 5.3.19.

Theorem 5.3.21

Suppose $T:(\underline{K},\kappa,\partial)^{op}\times(\underline{L},\eta,\psi)\to(\underline{M},\tau,\zeta)$ is a locally effectively continuous functor and both $(\underline{K},\kappa,\partial)$ and $(\underline{L},\eta,\psi)$ have locally determined effective colimits of embeddings. Then $T^P:(\underline{K},\kappa,\partial)^P\times(\underline{L},\eta,\psi)^P\to(\underline{M},\tau,\zeta)^P$ is an effectively continuous functor.

<u>proof</u> Let $(\Delta=<(A_n,B_n),((f_n,f_n^R),(g_n,g_n^R))>,i)$ be an indexed ω-codiagram in $(\underline{K},\kappa,\partial)^P \times (\underline{L},\eta,\psi)^P \to (\underline{M},\tau,\zeta)^P$. Also let $((\sigma,\sigma^R),(\delta,\delta^R)):\Delta\to(A,B)$ be an effective ω-colimiting coccone of (Δ,i) with an inedx $<<<a,b>,k>,x>$. Evidently $(\sigma,\sigma^R):\Delta^L\to A$ is an effective ω-colimiting cocone of $(\Delta^L,lt(i))$ with an index $<<a,lt(k)>,lt(x)>$. Also $(\delta,\delta^R):\Delta^R\to B$ is an effective ω-colimiting cocone of $(\Delta^R,rt(i))$ with an index $<<b,rt(k)>,rt(x)>$, where lt and rt are as in the proof of 5.1.14. By the assumption of the theorem, $<\delta_n\cdot\delta_n^R>$ and $<\sigma_n\cdot\sigma_n^R>$ are effective chains and $id_A=\sqcup\sigma_n\cdot\sigma_n^R$, $id_B=\sqcup\delta_n\cdot\delta_n^R$. Let $<c,d>$ be a functor index of T^P. Then $(T^P\cdot\Delta, f_{dg}(i,c,d))$ is an indexed ω-codiagram in $(\underline{M},\tau,\zeta)^P$ and $(T^P((\sigma,\sigma^R),(\delta,\delta^R)):T^P\cdot\Delta\to T^P(A,B), f_{cocone}(c,d,<a,b>,k))$ is an indexed ω-cocone of $(T^P\cdot\Delta, f_{dg}(i,c,d))$, for T^P is effective and we have 5.1.5. Now we have:

$$T^P((\sigma_n,\sigma_n^R),(\delta_n,\delta_n^R))_1\cdot T^P((\sigma_n,\sigma_n^R),(\delta_n,\delta_n^R))_2$$
$$=T(\sigma_n\cdot\sigma_n^R,\delta_n\cdot\delta_n^R).$$

Thus $T^P((\sigma_n,\sigma_n^R),(\delta_n,\delta_n^R))_1\cdot T^P((\sigma_n,\sigma_n^R),(\delta_n,\delta_n^R))_2$ is an effective chain since T is locally effectively continuous. Furthermore it can readily be seen that:

$$\sqcup T^P((\sigma_n,\sigma_n^R),(\delta_n,\delta_n^R))_1\cdot T^P((\sigma_n,\sigma_n^R),(\delta_n,\delta_n^R))_2$$
$$=id_{T(A,B)}.$$

Therefore by 5.3.7, $T^P((\sigma,\sigma^R),(\delta,\delta^R))$ is an effective ω-colimiting cocone with an index: $s(a,b,k,x,i)$

$$s(a,b,k,x,i)$$
$$=<f_{cocone}(c,d,<a,b>,k),$$
$$u(\pi_1\cdot f_{cocone}(c,d,<a,b>,k),\pi_2\cdot f_{cocone}(c,d,<a,b>,k),f_{dg}(i,c,d))>.$$

s above can obviously be considered as a recursive function. Thus T^P has a continuity index $<<c,d>,j>$ where $\phi_j=s$.

∎

Corollary 5.3.22

Let $(\underline{K},\kappa,\partial)$ and $(\underline{L},\eta,\psi)$ be effective O-categories with the effective S-property and $(\underline{M},\tau,\zeta)$ be an effective O-category. If $T:(\underline{K},\kappa,\partial)^{op}\times(\underline{L},\eta,\psi)\to(\underline{M},\tau,\zeta)$ is locally effectively continuous, then $T^P:(\underline{K},\kappa,\partial)^P\times(\underline{L},\eta,\psi)^P\to(\underline{M},\tau,\zeta)^P$ is an effectively continuous functor.

proof By 5.3.21 and 5.3.16. ∎

Now we will check if we have obtained the right kind of abstraction.

Theorem 5.3.23

EGD,ED,EGS and ES are all effective Dom-categories. Thus by 5.3.12, EGD^P,ED^P,EGS^P and ES^P are all effective ω-categories.

proof We prove this for EGD. Almost the same argument establishes this theorem for the others. By 2.1.7, the composition is recursive wrt indices. It is evident that from an acceptable index $\langle i,j\rangle$ of an effectively given domain (D,ε), we can construct the graph of id_D, which is $\{\langle n,m\rangle \mid \varepsilon(m)\sqsubseteq\varepsilon(n)\}$. By 2.1.6, in EGD, we have a recursive function *Idt* s.t. $Id_{\kappa(a)}=\partial(a,a)(Idt(a))$. Thus EGD is an effective category. It is obvious that EGD has the final object, namely the effectively given one point cpo. $Hom(\kappa(i),\kappa(j))$ always has the least element $\perp_{\kappa(i),\kappa(j)}=\lambda x\in\kappa(i).\perp_{\kappa(j)}$ where $\perp_{\kappa(j)}$ is the bottom of $\kappa(j)$. It is obvious that there is a recursive function *Bottom* s.t. $\partial(i,j)(Bottom(i,j))=\perp_{\kappa(i),\kappa(j)}$. Obviously $\perp_{\kappa(i),\kappa(j)}\cdot f=\perp_{\kappa(m),\kappa(j)}$ for all $f:\kappa(m)\to\kappa(i)$. Thus EGD is effectively empty chain complete. By 2.1.11 EGD is an effective O-category. Now let $(\Delta=\langle D^{\varepsilon}_{n}n,(f_n,f^R_n)\rangle,i)$ be an indexed ω-codiagram in EGD^P. Evidently Δ is an effective sequence of computable projection pairs with a sequence index i.

Let $(f_\infty, f_\infty^R) = \langle (f_{m\infty}: D_m \to D_\infty, f_{m\infty}^R : D_\infty \to D_m) \rangle$ be the universal cocone of Δ, where $(D_\infty, \varepsilon_\infty)$ is the inverse limit of Δ. Obviously $((f_\infty, f_\infty^R), \langle \mathit{Ivlim}(i), \mathit{Ucocone}(i) \rangle)$ is an indexed ω-cocone of (Δ, i) where *Ivlim* and *Ucocone* are as in 2.4.4 and 2.4.7. Thus $(f_\infty^R, \langle \mathit{Ivlim}(i), \mathit{rt}(\mathit{Ucocone}(i)) \rangle)$ is an indexed ω-cocone of $(\Delta^P, \mathit{pr}(i))$.
Let $(\nu = \langle \nu_n \rangle : D^\varepsilon \to \Delta^R, \langle a, k \rangle)$ be another indexed ω-cocone of $(\Delta^P, \mathit{pr}(i))$. Let $\theta : D \to D_\infty$ be the following computable function:

$$\theta(d) = \langle \nu_0(d), \ldots, \nu_n(d), \ldots\ldots \rangle.$$

It is obvious that θ is the unique mediating map from f_∞^R to ν, i.e., $\nu_m = f_m^R \cdot \theta$. Remember that $\theta(d) = \sqcup f_{m\infty} \cdot \nu_m(d)$. But we have:

$$f_{m\infty} = \partial(\pi_1 \cdot \phi_i(m), \mathit{Ivlim}(i)(\pi_1 \cdot \phi_{\mathit{Ucocone}(i)}(m)),$$

$$\nu_m = \partial(a, \pi_1 \cdot \phi_i(m))(\phi_k(m)).$$

Since EGD is an effective category, we have:

$$f_{m\infty} \cdot \nu_m = \partial\text{-}\mathit{Compose}(\pi_1 \cdot \phi_i(m), \mathit{Ivlim}(i), a, \pi_1 \cdot \phi_{\mathit{Ucocone}(i)}(m), \phi_k(m)).$$

Let c be a recursive function s.t.:

$$\phi_{c(i,a,k)}(m) = f_{m\infty} \cdot \nu_m.$$

It can readily be seen that $f_{m\infty} \cdot \nu_m \sqsubseteq f_{(m+1)\infty} \cdot \nu_{m+1}$. Therefore $\langle f_{m\infty} \cdot \nu_m \rangle$ is a $\partial(a, \mathit{Ivlim}(i))$-effective chain with a chain index $c(i,a,k)$. Thus $\theta = \sqcup f_{m\infty} \cdot \nu_m$. By the first axiom of the effective O-category, we have:

$$\begin{aligned} \theta &= \sqcup f_{m\infty} \cdot \nu_m \\ &= \partial(a, \mathit{Ivlim}(i))(\mathit{Lub}(a, \mathit{Ivlim}(i), c(i,a,k))). \end{aligned}$$

Let ω-*med* be a recursive function s.t.

$$\phi_{\omega\text{-}\mathit{med}(i)}(x,y) = \mathit{Lub}(x, \mathit{Ivlim}(i), c(i,x,y)).$$

Let ω-*lim*=*Ivlim* and ω-*cocone*=*Ucocone*. Then EGD has the effective S-property. In summary, we have established the theorem. ∎

Now we will observe that the domain constructors induce right kind of functors.

Let $\underline{1}$ be the single object category and ρ, η be indexings which make $(\underline{1}, \rho, \eta)$ an effective Dom-category. Define $\times : (\underline{1}, \rho, \eta)^{op} \times (EGD \times EGD) \to EGD$ by:

$$\times(1, (A^{\alpha}, B^{\beta})) = A^{\alpha} \times B^{\beta}$$

$$\times(id_1, (f: A^{\alpha} \to A'^{\alpha'}, g: B^{\beta} \to B'^{\beta'})) = f \times g.$$

Notice that essentially $\times$ is a functor $\times : EGD \times EGD \to EGD$ s.t. $\times(A^{\alpha}, B^{\beta}) = A^{\alpha} \times B^{\beta}$ and $\times(f,g) = f \times g$. Then for $\times^P : (1, \rho, \eta)^P \times (EGD \times EGD)^P \to (EGD)^P$, we have:

$$\times^P((id_1, id_1), ((f,g), (f^R, g^R)))$$
$$= (\times(id_1, (f,g)), \times(id_1, (f^R, g^R)))$$
$$= (f \times g, f^R \times g^R).$$

Also $\times^P(1, (A^{\alpha}, B^{\beta})) = \times(1, (A^{\alpha}, B^{\beta})) = A^{\alpha} \times B^{\beta}$. Since $(EGD)^P = EGD^P$ and $(ECD \times EGD)^P = EGD^P \times EGD^P$, $\times^P$ can be considered as a functor $\times^P : EGD^P \times EGD^P \to EGD^P$ s.t.:

$$\times^P(A^{\alpha}, B^{\beta}) = A^{\alpha} \times B^{\beta},$$

$$\times^P((f, f^R), (g, g^R)) = (f \times g, f^R \times g^R).$$

<u>Theorem 5.3.24</u>

$\times$ is a locally effectively continuous functor. Thus $\times^P$ is an effectively continuous functor.

<u>proof</u> It can readily be seen that from the graph of $f: \overline{\xi}(i) \to \overline{\xi}(j)$ and $g: \overline{\xi}(k) \to \overline{\xi}(m)$, we can construct the graph of $\times(f,g)$. Thus is an effective functor for we have 2.1.4. If $\langle f_n \rangle$ and $\langle g_n \rangle$ are $\zeta(i,j)$-effective chain and $\zeta(k,m)$-effective chain respectively then $\langle \times(id_1, (f_n, g_n)) \rangle = \langle f_n \times g_n \rangle$ is a $\zeta(\mathit{Prod}(i,k), \mathit{Prod}(j,m))$-effective chain. Also $\bigsqcup \times(id_1, (f_n, g_n)) = \bigsqcup (f_n \times g_n) = \bigsqcup f_n \times \bigsqcup g_n = \times(id_1, \bigsqcup(f_n, g_n))$. Thus we have established the theorem. ∎

Remember $\bar{\ }$ and ζ are the object indexing and the morphism indexings of EGD respectively. Also remember that 5.2.24 implies that $\times^P$ is an effectively continuous functor.

By almost the same argument as above, we can define an locally effectively continuous functor $+:(\underline{1},\rho,\eta)^{op}\times(EGD\times EGD)\to EGD$ and an effectively continuous functor $+^P$, for the domain constructor +.

Definition 5.3.25

Define a functor $\to:(EGD)^{op}\times EGD\to EGD$ by:

$$\to(\bar{\xi}(i),\bar{\xi}(j))=[\bar{\xi}(i)\to\bar{\xi}(j)]$$

$$(f:\bar{\xi}(i)\to\bar{\xi}(j),g:\bar{\xi}(k)\to\bar{\xi}(m))=\lambda h\in[\bar{\xi}(i)\to\bar{\xi}(k)].g\cdot h\cdot f.$$

□

For $(\to)^P=\to^P:EGD^P\times EGD^P\to EGD^P$, we have:

$$\to^P(A^{\alpha},B^{\beta})=[A^{\alpha}\to B^{\beta}]$$

$$\to^P((f,f^R),(g,g^R))=(\to(f^R,g),\to(f,g^R))$$

$$=(\lambda h\in[A\to A'].g\cdot h\cdot f^R,\lambda h\in[B\to B'].g^R\cdot h\cdot f)$$

where $(f,f^R):A^{\alpha}\to B^{\beta}$ and $(g,g^R):A'^{\alpha'}\to B'^{\beta'}$.

Theorem 5.3.26

$\to$ is a locally effectively continuous functor. Thence $\to^P$ is an effectively continuous functor.

proof It can readily be seen that from the graph of $f:\bar{\xi}(i)\to\bar{\xi}(j)$ and that of $g:\bar{\xi}(k)\to\bar{\xi}(m)$, we can effectively construct the graph of $\to(f,g)$. Therefore $\to$ is an effective functor, for we have 2.1.4. Thus given $\zeta(i,j)$-effective chain $\langle f_n\rangle$ and $\zeta(k,m)$-chain $\langle g_n\rangle$, $\langle\to(f_n,g_n)\rangle$ is a $\zeta(\mathit{Func}(j,k),\mathit{Func}(i,m))$-effective chain. Furthermore $\bigsqcup\to(f_n,g_n)=\bigsqcup\lambda h\in[\bar{\xi}(j)\to\bar{\xi}(k)].g_n\cdot h\cdot f_n=\lambda h\in[\bar{\xi}(j)\to\bar{\xi}(k)].\bigsqcup g_n\cdot h\cdot f_n=\to(\bigsqcup f_n,\bigsqcup g_n)$. ∅

We can define locally effectively continuous functors $\times$, $+$ and $\to$ over ED, EGS and ES. Also we can define effectively continuous functors $\times^P$, $+^P$ and $\to^P$ over ED^P, EGS^P and ES^P. Details of these arguments are almost the same as above.

<u>Definition 5.3.27</u>

Definine a functor $F[]:(\underline{1},\rho,\eta)^{op}\times EGS\to EGS$ by:

$$F[](1,A^\alpha)=F[A^\alpha]$$

$$F[](id_1,f:A^\alpha\to B^\beta)=\hat{f}:F[A^\alpha]\to F[B^\beta].$$

□

<u>Theorem 5.3.28</u>

$F[]$ is a locally effectively continuous functor. Thus $F[]^P$ is an effectively continuous functor.

<u>proof</u> By 4.7.4 and 4.7.5, $F[]$ is an effective functor. Let $\langle f_n\rangle$ be an effective chain. By definition of f, $\langle \hat{f}_n\rangle$ is an effective chain. Furthermore:

$$\sqcup F[](id_1,f_n)=\sqcup\hat{f}_n=\widehat{\sqcup f}_n=F[](id_1,\sqcup f_n).$$

Thus $F[]$ is a locally effectively continuous functor.

☒

We can define a locally effectively continuous functor $F[]$ over ES. Also we can define an effectively continuous functor $F[]^P$ over ES^P. Details of this is almost the same as above.

It should be noted that for $F[]^P:(\underline{1},\rho,\eta)^P\times EGS^P\to EGS^P$, we have:

$$F[]^P((id_1,id_1),(f,f^R))$$
$$=(F[](id_1,f),F[](id_1,f^R))$$
$$=(\hat{f},\hat{f}^R).$$

5.4 More on Effectively Initial Algebras

In the previous section, we observed that if $(\underline{K},\kappa,\partial)$ is an effective Dom-category then the final object $\perp$ is an effectively initial object in $(\underline{K},\kappa,\partial)^*$. In this section, we study a very important implication of this coincidence.

Definition 5.4.1

Given an effective Dom-category $(\underline{K},\kappa,\partial)$, define $E_E,E_P:(\underline{K},\kappa,\partial)^P \to (\underline{K},\kappa,\partial)$ by:

$$E_E(K)=E_P(K)=K$$

$$E_E((f,f^R))=f \qquad\qquad E_P((f,f^R))=f^R.$$

□

Lemma 5.4.2

E_E is an effectively continuous functor and it effectively reflects effective ω-colimits. In fact for some recursive *Ref*; if (Δ,i) is an indexed ω-codiagram in $(\underline{K},\kappa,\partial)^P$, $((\nu,\nu^R),\langle a,k\rangle)$ is an indexed ω-cocone of $(\Delta^E,em(i))=E_E((\Delta,i))$ with an index $\langle\langle a,lt(k)\rangle,z\rangle$ then (ν,ν^R) is an effective ω-colimiting cocone of (Δ,i) with an index $\langle\langle a,k\rangle,Ref(a,k,z)\rangle$.

proof Obviously E_E is an effectively continuous functor. Since $(\underline{K},\kappa,\partial)$ has the effective S-property, there is an effective S-cocone $(((\mu,\mu^R),\langle x,y\rangle)$ of (Δ,i). As proved in 5.3.9, $\mu:\Delta^E\to\kappa(x)$ is an effective ω-colimiting cocone of $(\Delta^E,em(i))$ with an index $\langle\langle x,lt(y)\rangle,u(x,y,i)\rangle$. Therefore $\kappa(x)\cong\kappa(a)$ via $(\theta,\theta^R):\kappa(x)\to\kappa(a)$ where

$$\theta=\partial(x,a)(\phi_{u(x,y,i)}(a,lt(k))) \quad \text{and}$$

$$\theta^R=\partial(a,x)(\phi_z(x,lt(k))).$$

As observed in 5.3.9, (μ,μ^R) is an effective ω-colimiting cocone of (Δ,i) with an index $\langle\langle x,y\rangle,\upsilon(x,y,i)\rangle$. Therefore by the arguments between 5.1.3 and 5.1.4, if $(\lambda,\lambda^R):\Delta\to\kappa(c)$ is an effectively generable cocone with an index $\langle c,d\rangle$, then $(\sigma\cdot\theta^R,\theta\cdot\sigma^R)$ is the unique mediating morphism from (ν,ν^R) to (λ,λ^R) where (σ,σ^R) is the mediating morphism from (μ,μ^R) to (λ,λ^R) given by:

$$\sigma=\partial(x,c)(\pi_1\cdot\phi_{\upsilon(x,y,i)}(c,d))=\partial(x,c)(\phi_{lt(\upsilon(x,y,i))}(c,d))$$

$$\sigma^R=\partial(c,x)(\pi_2\cdot\phi_{\upsilon(x,y,i)}(c,d))=\partial(c,x)(\phi_{rt(\upsilon(x,y,i))}(c,d)).$$

Thus we have:

$$\sigma\cdot\theta^R=\partial(a,c)(\partial\text{-}Compose(a,x,c,\phi_{lt(\upsilon(x,y,i))}(c,d),\phi_z(x,lt(k)))),$$

$$\theta\cdot\sigma^R=\partial(a,c)(\partial\text{-}Compose(c,x,a,\phi_{u(x,y,i)}(a,lt(k)),\phi_{rt(\upsilon(x,y,i))}(c,d))).$$

Thus we have established the lemma.

∎

Note that for the definitions of *lt*, *em*, and *u*, readers are refered to the proof of 5.1.14, the comment right after 5.3.5, and 5.3.7.

<u>Theorem 5.4.3</u>

Let $(\underline{K},\kappa,\partial)$ be an effective Dom-category, $T':(\underline{K},\kappa,\partial)^*\to(\underline{K},\kappa,\partial)^*$ be an effectively continuous functor and $T:(\underline{K},\kappa,\partial)^P\to(\underline{K},\kappa,\partial)^P$ be a functor satisfying:

$$E_E\cdot T=T'\cdot E_E$$

then T is an effectively continuous functor. Furthermore if $\phi:TI\to I$ is an effectively initial T-algebra then $E_E(\phi):T'I\to I$ is an effectively initial T'-algebra.

<u>proof</u> Since E_E is effectively continuous, $T'\cdot E_E=E_E\cdot T$ is an effectively continuous functor. Since E_E effectively reflects

effective ω-colimits, T is an effectively continuous functor. Let $\langle\langle d,e\rangle,z\rangle$ be a continuity index of T. Let $(\Delta,f_{gn}(d,e))$ be an indexed ω-codiagram as in 5.1.9. Let $\mu:\Delta\to\kappa(i)$ be an effectively generable ω-colimiting cocone of $(\Delta,f_{gn}(d,e))$ with an index $\langle\langle i,k\rangle,x\rangle$. Then $T(\mu)$ is an effective ω-colimiting cocone of $(T\cdot\Delta,f_{dg}(f_{gn}(d,e),d,e))$ with an index:

$$\langle f_{cocone}(d,e,i,k),\phi_z(i,k,x,f_{gn}(d,e))\rangle.$$

Thus as shown in the proof of 5.1.9, $(\kappa(i),\alpha)$ given by:

$$\alpha=\partial(\phi_d(i),i)(\phi_{\phi_z(i,k,x,f_{gn}(d,e))}(i,suc(k)))$$

is the effectively initial T-algebra with an index:

$$\langle\langle i,\phi_{z(i,k,x,f_{gn}(d,e))}(i,suc(k))\rangle,f(x)\rangle$$

where f is as in the proof of (2)-5.1.9 and suc is a recursive function satisfying: $\phi_{suc(v)}(x)=\phi_v(x)+1$. Obviously $E_E(\mu)$ is an effective ω-colimiting cocone of $(E_E\cdot\Delta,f_{dg}(f_{gn}(d,e),j,m))$ with an index $\langle\langle i,lt(k)\rangle,lt(x)\rangle$, where $\langle j,m\rangle$ is a functor index of E_E. Therefore $E_E(\alpha):T'\kappa(i)\to\kappa(i)$ is the unique $\underline{K}$-morphism s.t. $E_E(\mu_{n+1})=(E_E(\alpha))\cdot T'(E_E(\mu_n))$. Now let $\beta:T'B\to B$ be a T'-algebra with an algebra index $\langle b,c\rangle$. Define $\nu_n:T^n\bot\to B$ by:

$$\nu_0=E_E(\bot_B)$$

$$\nu_{n+1}=\beta\cdot T'(\nu_n)$$

where $\bot_B$ is the unique morphism from $\bot$ to B in $(\underline{K},\kappa,\partial)^P$. Since $E_E(\bot_B)$ is the unique morphism from $\bot$ to B in $(\underline{K},\kappa,\partial)^*$, we can readily observe that $\langle\nu_n\rangle$ is a cocone of $E_L\cdot\Delta$. Since the definition of ν_n is iterative, it can readily be seen that $\langle b,t(b,c)\rangle$ is a cocone index of $\langle\nu_n\rangle$, for some recursive t. Thus $\gamma=\partial(i,b)(\phi_{lt(x)}(b,t(b,c)))$ is the unique morphism s.t. $\gamma\cdot E_L(\mu_n)=\nu_n$. Now by almost the same argument as in the proof of 5.1.9, we can show that this γ is the unique T'-homomorphism from $(\kappa(i), E_E(\alpha))$ to (B,β). Let w be a recursive function satisfying:

$$\phi_{w(x)}(b,c)=\phi_{lt(x)}(b,t(b,c)).$$

Then $(\kappa(i), E_E(\alpha))$ is an effectively initial T'-algebra with an index:

$$<<i,\phi_m(\phi_{\phi_z}(i,k,x,f_{gn}(d,e))(i,suc(k)))>,w(x)>.$$

□

This important theorem states that if we get an effectively continuous functor $T:(\underline{K},\kappa,\partial)^P\to(\underline{K},\kappa,\partial)^P$ which can be naturally extended to an effectively continuous functor $T':(\underline{K},\kappa,\partial)^*\to(\underline{K},\kappa,\partial)^*$, then the effectively initial T-algebra is also an effectively initial T'-algebra over $(\underline{K},\kappa,\partial)^*$.

Since the initial algebra index of ϕ is effectively obtainable from the continuity index of T, which is effectively obtainable from that of T', it can readily be seen that the initial algebra index of $E_E(\phi)$ is effectively obtainable from the continuity index of T'.

TOPICS FOR FURTHER RESEARCH

To conclude this dissertation, we consider several interesting topics of further research.

First it should be made clear whether non-algebraic continuous cpo's have a useful general theory. The only convincing example known so far which suggests the necessity for such cpo's is the interval lattice. But recently Weihrauch & Schreiber [28] announced that with the aid of a metric suitably defined on algebraic cpo's, we can handle interval lattice without regarding it as a non-algebraic continuous cpo.

Roughly speaking, our notion of effective categories is the categories whose categorical constructions, like composition of morphisms, universality etc, are effective. We observed that this notion worked out quite smoothly for the purpose of solving recursive domain equations effectively. It should be worthwhile to investigate further what other applications there might be. A similar theme was proposed by Ehrig [2].

Compared with the universal domain approach, our categorical approach lacks a natural notation for computable objects. We conjecture the following on this issue:

Conjecture

Let $T: EGD^P \to EGD^P$ be an effective functor composed from $\times^P$, $+^P$, and $\to^P$. Then each of these functors is associated with a collection of computable functions. For example, $\{\pi_1, \pi_2, cons\}$ for $\times^P$. We claim that there exists a a suitable function defining schema Θ s.t. if $\phi: TA \to A$ is the effectively initial T-algebra, then every computable function over A can be defined from the collection of functions associated with T and ϕ and ϕ^{-1} using the schema Θ. For want is such Θ.

Notice that this conjecture is a natural extension of J. McCarthy's [8] relative computability thesis. A hint for this problem can be found in Plotkin [11]. Readers are requested to pay attention to the fact that we are making a claim rather than a thesis.

It is a very interesting topic to search for a class of po-structures for each of which the notion of computability of elements does not depend on the effective indexings of the basis. In other words, a class $\mathcal{C}$ of po-structures s.t. for every $C \in \mathcal{C}$, and effective bases ε and ε' of C, $C^{\varepsilon'} \stackrel{e}{=} C^{\varepsilon}$.

REFERENCES

[1] Egli & Constable, Computability concepts for programming language semantics, Theoretical Computer Science, Vol. 2, (1976).

[2] Ehrig, Kreowski & Padawitz, Stepwise specification and implementation of abstract data types, Technical Report, Technical University of Berlin, FB 20, (1977).

[3] Kanda, A., Data types as effective objects, Theory of Computation Report, No. 22, Warwick University, (1977).

[4] Kanda, A., Fully effective solutions of recursive domain equations, Proc. of the MFCS'79, Lecture Notes in Computer Science, No. 74, (1979).

[5] Kanda & Park, When are two effectively given domains identical?, Proc. of the 4th GI Theoretical Computer Science Symposium, Lecture Note in Computer Science, No. 67, (1979).

[6] Kleene, S., Introduction to metamathematics, North-Holland Publishing Co., Amsterdam, (1952)

[7] Milner, R., Models of LCF, AIM-186/CS-332, Computer Science Department, Stanford University, (1973).

[8] McCarthy, J., A basis for a mathematical theory of computation, in Computer Programs and Formal Systems, North-Holland, Amsterdam, (1963).

[9] Park, D., Unpublished note, Warwick University, (1970).

[10] Plotkin, G., A power domain construction, SIAM Journal on Computing, Vol. 5, No. 3, (1976).

[11] Plotkin, G., LCF as a programming language, Theoretical Computer Science, Vol. , (1976).

[12] Plotkin, G., Unpublished note, Antiebe, (1977).

[13] Reynolds, J., Notes on a lattice theoretic approach to the theory of computation, Systems and Information Sicence Report, Syracuse University, (1972).

[14] Rogers, H., Theory of recursive functions and effective computability, McGraw-Hill, New York, (1967).

[15] Rosen & Markowsky, Bases for chain complete posets, IBM Journal of Research and Development, Vol. 20, (1976).

[16] Scott, D., Outline of a mathematical theory of computation, Proc. of the 4th Annual Pronceton Conference on Information Science and Systems, (1970).

[17] Scott, D., Continuous lattices, Lecture Note in Mathematics, No. 274, (1971).

[18] Scott, D., Data types as lattices, Unpublished Lecture Note, Amsterdam, (1972).

[19] Scott, D., Data types as lattices, SIAM Journal on Computing, Vol. 5, (1976).

[20] Scott, D., Mathematical theory of computation, Unpublished Lecture Note, Oxford, (1977).

[21] Scott & Strachey, Towards a mathematical semantics of computer languages, Proc. of Symposium on Computer and Automata, Polytechnical Institute of Brooklin, (1971).

[22] Smyth, M., Effectively given domains, Theoretical Computer Science, Vol. 5, (1977).

[23] Smyth & Lehmann, Algebraic specification of data types, A synthetic approach, Technical Report, No. 115, Department of Computer Studies, Leeds University, (1978).

[24] Smyth & Plotkin, The categorical solution of recursive domain equations, Proc. of the 18th. FOCS Conference (1977). Also in Research Report, No. 60, D.A.I., Edinburgh Univeristy. (1978).

[25] Stoy, J., Denotational semantics, The Scott-Strachey approach to the theory of programming languages, MIT Press, (1978).

[26] Tang, A., Recursion theory and descriptive set theory in effectively given T_0-spaces, Ph.D. Thesis, Princeton University, (1974).

[27] Tennent, R., The denotational semantics of programming languages, CACM Vol. 19, No. 18, (1977).

[28] Weihrauch & Schreiber, Weighted cpo's and metric spaces, Proc. of the FCS'79, Lecture Note in Computer Science, No. 75, (1979).

[29] Wand, M., Fixed-point constructions in order-enriched categories, Thecnical Report No. 23, Computer Science Department, Indiana University, (1977).

[30] Smyth, M., Power domains, J.c.S.S., Vol. 16, No.1, (1978).

[31] Plotkin G., Private Communication, (1980).

[32] Scott, D., Lattice theory, data types and semantics, Formal Semantics of Programing Languages, Courant Computer Science Symposium 2, Editor Randall Rustin, Prentice-Hall, (1972).