 Electronic Notes in Theoretical Computer Science 167 (2007) 345–364 

[www.elsevier.com/locate/entcs](http://www.elsevier.com/locate/entcs)

On Computable Metrization

# Tanja Grubba[1](#_bookmark0)

*Department of Mathematics and Computer Science University of Hagen*

*Hagen,Germany*

# Klaus Weihrauch[2](#_bookmark0)

*Department of Mathematics and Computer Science University of Hagen*

*Hagen,Germany*

Abstract

Every second-countable regular topological space *X* is metrizable. For a given “computable” topo- logical space satisfying an axiom of computable regularity M. Schr¨oder [[9](#_bookmark34)] has constructed a com- putable metric. In this article we study whether this metric space (*X, d*) can be considered com- putationally as a subspace of some computable metric space [[13](#_bookmark37)]. While Schr¨oder’s construction is “pointless”, i.e., only sets of a countable base but no concrete points are known, for a computable metric space a concrete dense set of computable points is needed. By partial completion we extend (*X, d*) to a metric space (*X*˜ *, d*˜) with computable metric and canonical representation. We construct a computable sequence (*xi*)*i*∈N of points which is dense in (*X*˜ *, d*˜). The isometric embedding of *X* into *X*˜ is computable. Its inverse is computable if some further computability axiom holds true. The space (*X*˜ *, d*˜) can be embedded computationally into the computable metric space generated by the sequence (*xi*)*i*∈N of points. The inverse of this embedding is continuous.

*Keywords:* Computable Analysis, TTE, computable metrization, computable embedding, computable metric space

# Introduction

Computable Analysis connects Computability/Computational Complexity with Analysis/Numerical Computation by combining concepts of approxima-

1 Email:[tanja.grubba@fernuni-hagen.de](mailto:tanja.grubba@fernuni-hagen.de)

2 Email:[klaus.weihrauch@fernuni-hagen.de](mailto:klaus.weihrauch@fernuni-hagen.de)

1571-0661 © 2007 Elsevier B.V. Open access under [CC BY-NC-ND license.](http://creativecommons.org/licenses/by-nc-nd/3.0/)

doi:10.1016/j.entcs.2006.08.020

tion and of computation. During the last 70 years various mutually non- equivalent models of real number computation have been proposed ([[11](#_bookmark36)], Chap. 9 in [[13](#_bookmark37)]). Among these models the representation approach (Type-2 Theory of Effectivity, TTE) [[13](#_bookmark37)] introduced for real functions by Grzegorczyk and Lacombe [[6](#_bookmark29),[7](#_bookmark30)] seems to be particularly realistic, flexible and expressive. So far the study of computability on sets of points, sets (open, closed, compact) and continuous functions has developed mainly bottom-up, i.e., from the real numbers to Euclidean space and metric spaces [[17](#_bookmark42),[2](#_bookmark27),[15](#_bookmark40),[13](#_bookmark37),[18](#_bookmark43),[1](#_bookmark26),[19](#_bookmark44)]. But often generalisations to more general spaces are needed (locally compact Hausdorff spaces [[3](#_bookmark28)], non-metrizable spaces [[16](#_bookmark41)], second countable *T*0-spaces [[9](#_bookmark34),[5](#_bookmark31)]). In this article we study computability in basic topology using the representation approach to computable analysis.

Schr¨oder [[9](#_bookmark34)] has shown that every “computably regular” space has a com- putable metric generating the same topology. In his proof he effectivizes a classical proof where the metric is defined from a sequence of pseudo-metics each of which is defined via an Urysohn function [[4](#_bookmark32)] which can be interpreted as a contour map on the space. On the other hand, in TTE *computable met- ric spaces* are defined axiomatically [[13](#_bookmark37)] by a dense sequence of points with computable distances. So the question arises whether the “computably regu- lar” space with computable metric can be considered as a computable metric space. Schr¨oder’s top-down approach is pointless, that is, only basic open sets “are known” and handled via code names. Concrete points, in particular computable ones, are not considered. It is even possible that the space con- tains no computable points at all. Since every computable metric space must have a dense set of computable points a computably metrizable space cannot be a computable metric spaces in general. In this article we study whether Schr¨oder’s space with computable metric can be embedded into a computable metric space.

It has turned out that the concept of a computable topological space from

[[13](#_bookmark37)] is not optimal for a top-down study of computability in topology. There- fore we start from a slightly different concept of “computable T0-spaces” (al- ready considered in [[5](#_bookmark31)]). In Section [3](#_bookmark1) we discuss the new definition, introduce (standard) representations of the points and the open sets and show that they meet our intuition very well by considering robustness and a basic intuitive idea of “*x* ∈ *O*” for an open set *O*.

Since for our construction of computable points we need details of Schr¨oder’s proof of computable metrization [[9](#_bookmark34)] (which as part of a TR is not easily accessible) in Section [4](#_bookmark11) we outline his proof in a slightly modified terminology. Most of the proofs from Sections [3](#_bookmark1) and [4](#_bookmark11) are omitted.

In Section [5](#_bookmark16) we show that the represented metric space (*X, δ*X) has a

“weak completion” (*X*˜*, δ*˜) in which we can find a computable dense set of points. The embedding from (*X, δ*X) to (*X*˜*, δ*˜X) is computable, its inverse is continuous and even computable in some cases. The computable dense set induces a Cauchy representation *δC* of the space. The embedding from (*X*˜*, δ*˜) to (*X*˜*, δC*) is computable, its inverse is at least continuous.

# Preliminaries

In this article we use the framework of TTE (Type-2 theory of effectivity), see [[13](#_bookmark37)] for more details. A partial function from *X* to *Y* is denoted by *f* : ⊆ *X* → *Y* . We assume that Σ is a fixed finite alphabet containing the symbols 0 and 1 and consider computable functions on finite and infinite sequences of symbols Σ∗ and Σ*ω*, respectively, which can be defined, for example, by Type-2 machines, i.e., Turing machines reading from and writing on finite or infinite tapes. We use the “wrapping function” *ι* : Σ∗ → Σ∗, *ι*(*a*1*a*2 *... ak*) := 110*a*10*a*20 *... ak*011 for coding words such that *ι*(*u*) and *ι*(*v*) cannot overlap properly. We consider standard functions for finite or countable tupling on Σ∗ and Σ*ω* denoted by ⟨·⟩ . By “ ” we denote the subword relation.

We use the concept of multi-functions. A *multi-valued partial function*, or *multi-function* for short, from *A* to *B* is a triple *f* = (*A, B,* R*f* ) such that R*f* ⊆*A* × *B* (the *graph* of *f* ). Usually we will denote a multi-function *f* from *A* to *B* by *f* : ⊆ *A* ⇒ *B*. For *X*⊆*A* let *f* [*X*] := {*b* ∈ *B* | (∃*a* ∈ *X*)(*a, b*) ∈ R*f* }

and for *a* ∈ *A* define *f* (*a*) := *f* [{*a*}]. Notice that *f* is well-defined by the

values *f* (*a*)⊆*B* for all *a* ∈ *A*. We define dom(*f* ) := {*a* ∈ *A* | *f* (*a*) /= ∅}. In the applications we have in mind, for a multi-function *f* : ⊆ *A* ⇒ *B*, *f* (*a*) is

interpreted as the set of all results which are “acceptable” on input *a* ∈ *A*.

Any concrete computation will produce on input *a* ∈ dom(*f* ) some element *b* ∈ *f* (*a*), but usually there is no method to select a specific one. In accordance with this interpretation the “functional” composition *g* ◦ *f* : ⊆ *A* ⇒ *D* of

*f* : ⊆ *A* ⇒ *B* and *g* : ⊆ *C* ⇒ *D* is defined by dom(*g* ◦ *f* ) := {*a* ∈ *A* |

*a* ∈ dom(*f* ) and *f* (*a*)⊆dom(*g*)} and *g* ◦ *f* (*a*) := *g*[*f* (*a*)] (in contrast to “non-

deterministic” or “relational” composition *gf* defined by *g f* (*a*) := *g*[*f* (*a*)] for

*all a* ∈ *A*).

Notations *ν* : ⊆ Σ∗ → *M* and representations *δ* : ⊆ Σ*ω* → *M* are used for introducing relative continuity and computability on “abstract” sets *M* . For a representation *δ* : ⊆ Σ*ω* → *M* , if *δ*(*p*) = *x* then the point *x* ∈ *M* can be identified by the “name” *p* ∈ Σ*ω*.

For naming systems *γi* : ⊆ *Yi* → *Mi* (*i* = 0*,... , k*), a function *h* : ⊆ *Y*1 ×

*...* × *Yk* → *Y*0 is a (*γ*1*,... , γk, γ*0)-realization of *f* : ⊆ *M*1 × *...* × *Mk* ⇒ *M*0, if

*γ*0 ◦ *h*(*p*1*,... , pk*) ∈ *f* (*γ*1(*p*1)*,... , γk*(*pk*)) whenever *f* (*γ*1(*p*1)*,... , γk*(*pk*) exists.

The multi-function *f* is (*γ*1*,... , γk, γ*0)-continuous (–computable), if it has a continuous (computable) (*γ*1*,... , γk, γ*0)-realization.

For naming systems *γ* : ⊆ *Y* ⇒ *M* and *γ*' : ⊆ *Y* ' → *M* ' (*Y, Y* ' ∈ {Σ∗*,* Σ*ω*}), let *γ* ≤*t γ*' (t-reducible) and *γ* ≤ *γ*' (reducible), iff the identity id : *a* '→ *a*

(*a* ∈ *M* ) is (*γ, γ*')-continuous and (*γ, γ*')–computable, respectively. Define *t*- equivalence and equivalence as follows: *γ* ≡*t γ*' ⇐⇒ (*γ* ≤*t γ*' and *γ*' ≤*t γ*) and *γ* ≡ *γ*' ⇐⇒ (*γ* ≤ *γ*' and *γ*' ≤ *γ*), respectively.

Two representations induce the same continuity or computability, iff they are *t*-equivalent or equivalent, respectively. If multi-functions on represented sets have realizations, then their composition is realized by the composition of the realizations. In particular, the computable multi-functions on represented sets are closed under composition. Much more generally, the computable multi-functions on represented sets are closed under flowchart programming with indirect addressing [[14](#_bookmark39)]. This result allows convenient informal construc- tion of new computable functions on multi-represented sets from given ones.

Let *ν*N : ⊆Σ∗ → N be some standard notation of the natural numbers, let *ρ, ρ<* and *ρ>* be the Cauchy representation, the lower representation and the upper representation of the real numbers, respectively. For representations *δ* : ⊆ Σ*ω* → *M* and *δ*' : ⊆ Σ*ω* → *M* ' define [*δ, δ*']⟨*p, p*'⟩ := (*δ*(*p*)*, δ*(*p*')), *δ* ∧

*δ*'⟨*p, p*'⟩ = *x*, iff *δ*(*p*) = *δ*'(*p*') = *x* and [*δ*]*ω*⟨*p*0*, p*1*, p*2*,.. .*⟩ := *δ*(*p*0) × *δ*(*p*1) ×

*δ*(*p*2) × *...* [[13](#_bookmark37)].

# Representations of Points and Open Sets

In this section we define computable T0-spaces and summarize arguments showing that the standard representations of points and open subsets are very natural.

A topological space X = (*X, τ* ) is a T0-space, if for all *x, y* ∈ *X* such that *x* /= *y* there is an open set *U* ∈ *τ* such that *x* ∈ *U* ⇐⇒ *y* /∈ *U* . In a T0-space every point can be identified by the set of its neighbourhoods *U* ∈ *τ* . A space X is called *second-countable*, if it has a countable base [[4](#_bookmark32)]. In the following we consider only second countable *T*0-spaces. For introducing concepts of effectivity we assume that some notation *ν* of a base *β* with recursive domain is given.

Definition 3.1 [effective/computable T0-space]

* 1. An effective T0-space is a tuple X = (*X, τ, β, ν*) such that (*X, τ* ) is a second countable T0-space and *ν* : ⊆Σ∗ → *β* is a notation of a base *β* of *τ* with recursive domain. We assume that *U* /= ∅ for *U* ∈ *β*.
  2. We call an effective T0-space X a computable T0-space, iff it has com-

putable intersection, that is, if there is a computable function *h* : ⊆

Σ∗ × Σ∗ → Σ*ω* such that for all *u, v* ∈ dom(*ν*),

* + 1. *ν*(*u*) ∩ *ν*(*v*)= {*ν*(*w*) | *w* ∈ dom(*ν*) and *ι*(*w*) *h*(*u, v*)} *.*

Notice that for every effective *T*0-space there is a function *h* : ⊆ Σ∗ × Σ∗ → Σ*ω*, which is *continuous* since Σ∗ is discrete, such that ([1](#_bookmark2)) holds true. Therefore, Part ([ii](#_bookmark2)) in Def. [3.1](#_bookmark3) does not restrict the topology. In the definition of a computable *T*0-space the computable function *h* can be replaced by an

r.e. set: X is computable, iff there is an r.e. set *B*⊆(dom(*ν*))3 such that for all *u, v* ∈ dom(*ν*),

* + 1. *ν*(*u*) ∩ *ν*(*v*)= {*ν*(*w*) | (*u, v, w*) ∈ *B*} *.*

Starting from a notation *ν* with r.e. domain is not more general since then

there is an equivalent notation *ν*˜ with recursive domain such that the repre- sentations of points and sets introduced below by *ν* and *ν*˜ become equivalent.

In [[13](#_bookmark37)] Def. 3.2.1 an “effective topological space” is a triple S = (*X, σ, ν*') such that *ν*' : ⊆ Σ∗ → *σ* is a notation of a countable *subbase σ* of a T0- topology *τ* on *X*. S is called “computable”, if its equivalence problem {(*u, v*) ∈ (dom(*ν*'))2 | *ν*'(*u*) = *ν*'(*v*)} is recursively enumerable. For a foundation of Computable Topology, this new definition [3.1](#_bookmark3) seems to be more useful than the former definition from [[13](#_bookmark37)]. We give a short review of the differences:

While *ν* now is a notation of a base *β*, *ν*' is a notation of a subbase *σ*. Of course, *β* is also a subbase. On the other hand, from *ν*' the canonical notation *ν*¯ of the finite intersections of subbase elements is a notation of a base. But generally empty intersections cannot be excluded computationally.

We assume now that dom(*ν*) is recursive, but dom(*ν*') may be not even r.e. for an effective topological space but must be r.e. for a computable topological space.

While for an effective topological space the notation *ν*¯ satisfies ([1](#_bookmark2)) triv- ially, we require now ([1](#_bookmark2)) explicitly for a computable T0-space. Finally, for a computable topological space the equivalence problem of *ν*' must be r.e. while for the equivalence problem of *ν* there is no restriction.

In the following let X = (*X, τ, β, ν*) be an effective *T*0-space. For technical reasons we introduce

* + 1. *Dν* := {*q* ∈ Σ*ω* | *w* ∈ dom(*ν*) if *ι*(*w*) *q*} *.*

First, we define the standard representation of *X*.

Definition 3.2 [standard representation of *X*]

Define the standard representation *δ*X : ⊆Σ*ω* → *X* of X as follows: dom(*δ*X)⊆*Dν* and for all *x* ∈ *X* and *p* ∈ *Dν*,

* + 1. *δ*X(*p*)= *x* : ⇐⇒ {*w* ∈ dom(*ν*) | *x* ∈ *ν*(*w*)} = {*w* | *ι*(*w*) *p*} *.*

A *δ*X-name *p* of an element *x* ∈ *X* is a list of *all* words *w* such that

*x* ∈ *ν*(*w*). The definition of *δ*X corresponds to the definition of *δ*'

S

in [[13](#_bookmark37)],

Lemma 3.2.3, in particular, *δ*X is *admissible* with final topology *τ* (Sec. 3.2 in [[13](#_bookmark37)]).

The definition of *δ*X is “robust”.

Definition 3.3 [recursively related spaces]

We call two effective *T*0-spaces X1 = (*X, τ, β*1*, ν*1) and X2 = (*X, τ, β*2*, ν*2)

*recursively related*, iff there are computable functions *g, g*' : ⊆ Σ∗ → Σ*ω* auch that

* + 1. ​

*ν*1(*u*)=

*ι*(*w*)Q*g*(*u*)

*ν*2(*w*) and *ν*2(*v*)=

*ι*(*w*)Q*g*'(*v*)

*ν*1(*w*) *.*

Equivalently, X1 and X2 are recursively related, iff there are r.e. sets

*C, C*'⊆Σ∗ × Σ∗ such that

* + 1. ​

*ν*1(*u*)=

(*u,w*)∈*C*

*ν*2(*w*) and *ν*2(*v*)=

(*v,w*)∈*C*'

*ν*1(*w*) *.*

In particular, X1 and X2 are recursively related, if *ν*1 ≡ *ν*2.

Lemma 3.4 (*δ*X is robust)

*For effective* T0*-spaces* X1 = (*X, τ, β*1*, ν*1) *and* X2 = (*X, τ, β*2*, ν*2)*,*

1. *δ*X1 ≡*t δ*X2 ;
2. *δ*X1 ≡ *δ*X2 *if* X1 *and* X2 *are recursively related.*

For the representation *δ*X the relation *x* ∈ *U* for *x* ∈ *X* and *U* ∈ *β* becomes recursively enumerable (more precisely [*δ*X*, ν*]-r.e., Definitions 3.1.3 and 3.3.3 in [[13](#_bookmark37)], see Lemma [3.7](#_bookmark5)). We introduce the inner representation of the the set *τ* of the open and the outer representation of the set *τc* := {*Oc* | *O* ∈ *τ* } of closed sets.

Definition 3.5 [representations of open and closed sets]

1. Define the inner representation *θ<* : ⊆Σ*ω* → *τ* by dom(*θ<*) := *Dν* and
   * 1. ​

*θ<*(*p*) :=

*ι*(*w*)Q*p*

*ν*(*w*) *.*

1. Define the outer reperesentation *ψ>* : ⊆Σ*ω* → *τc* by

*ψ>*(*p*) := *X* \ *θ<*(*p*) *.*

Thus, *θ<*(*p*) is the union of all *ν*(*w*) such that *w* is listed by *p*. The outer representation of the closed sets is defined by the inner representation of their

complements. In [[13](#_bookmark37)] for subsets of the Euclidean space, *θ<* is called *θ*en and

*<*

*ψ>* is called *ψ*en. In [[2](#_bookmark27),[1](#_bookmark26)] for computable metric spaces, *ψ>* is called *δ*union.

*>*

The computable version of “*β* ⊆*τ* ” is *ν* ≤ *θ<*. In fact, the computable function *h* : *v* '→ *ι*(*v*)000 *...* translates *ν* to *θ<*. Recursively related spaces induce equivalent set representations:

Lemma 3.6 (*θ<* and *ψ>* are robust)

*For effective* T0*-spaces* X1 = (*X, τ, β*1*, ν*1) *and* X2 = (*X, τ, β*2*, ν*2)*,*

* 1. *θ<* ≡*t θ< and ψ>* ≡*t ψ> ;*

1 2 1 2

* 1. *θ<* ≡ *θ< and ψ>* ≡ *ψ>, if* X1 *and* X2 *are recursively related.*

1 2 1 2

The representation

*θ*˜ of the open sets where a name of *O* lists *all u* ∈

dom(*ν*1) such that *ν*1(*u*)⊆*O* is not useful since it is not robust in general (Theorem 5.1.14 in [[13](#_bookmark37)]).

As a fundamental idea about “open” the property *x* ∈ *O* can be verified by looking close enough to (zooming in on) *x* and *O*. In terms of computability the relation should be recursively enumerable: *x* ∈ *O* is true, iff we will know this fact after a finite computation. For a discussion of the concept of open see [[10](#_bookmark35)]. By the following lemma the representations *δ*X and *θ<* can be characterized up to equivalence as maximal elements among representations for which the element relation is open or r.e., respectively. It substantiates again that the representations *δ*X and *θ<* are very “natural”. Let “*x* ∈ *U* ”:=

{(*x, U* ) ∈ *X* × *β* | *x* ∈ *U* } and “*x* ∈ *O*”:= {(*x, O*) ∈ *X* × *τ* | *x* ∈ *O*}.

Lemma 3.7 *Let* X = (*X, τ, β, ν*) *be an effective* T0*-space. Then for every representation δ* : ⊆Σ*ω* → *X and θ* : ⊆Σ*ω* → *τ,*

* + 1. *“x* ∈ *U” is* (*δ, ν*) *-open* ⇐⇒ *“x* ∈ *O” is* (*δ, θ<*) *-open* ⇐⇒ *δ* ≤*t δ*X *,*
    2. *“x* ∈ *U” is* (*δ, ν*) *-r.e.* ⇐⇒ *“x* ∈ *O” is* (*δ, θ<*) *-r.e.* ⇐⇒ *δ* ≤ *δ*X *,*
    3. *“x* ∈ *O” is* (*δ*X*, θ*) *-open* ⇐⇒ *θ* ≤*t θ< ,*
    4. *“x* ∈ *O” is* (*δ*X*, θ*) *-r.e.* ⇐⇒ *θ* ≤ *θ<* (*for computable* X)*.*

Finite and countable union and finite intersection on open sets are com- putable, more precisely:

Lemma 3.8

*Let* X = (*X, τ, β, ν*) *be an effective* T0*-space. Then*

* + - 1. *countable union on τ is* ([*θ<*]*ω, θ<*)*–computable,*
      2. *intersection is* (*θ<, θ<, θ<*)*–computable, if* X *is computable.*

Instead of listing *all* basic open sets containing *x* we could list “sufficiently many” basic neighbourhoods.

Definition 3.9

Define a reperesentation *δb*

X

: ⊆Σ*ω* → *X* of X as follows: dom(*δb* )⊆*Dν* and

for all *x* ∈ *X* and *p* ∈ *Dν*,

X

* + 1. *δb* (*p*)= *x* : ⇐⇒ {*ν*(*u*) | *ι*(*u*) *p*} is a neighborhood base of *x.*

X

Lemma 3.10

* + - 1. *δ*X ≤ *δb ,*

X

* + - 1. *δb* ≤*t δ*X*,*

X

* + - 1. *in general neither δb* ≤ *δ*X *nor the relation x* ∈ *U is* [*δb , ν*]*-r.e.*

X

X

Proof: [i](#_bookmark7). Obvious.

* + - * 1. Let *q* ∈ Σ*ω* be a(n encoded) list of all (*u, v*) such that *ν*(*u*)⊆*ν*(*v*). Then for some computable function *h* : ⊆Σ*ω* × Σ*ω* → Σ*ω*, *p* '→ *h*(*p, q*) translates *δb*

X

to *δ*X

* + - * 1. Omitted.

# Effective Metrization

In [[9](#_bookmark34)] Schr¨oder has constructed a computable metric for a given “computable” regular space. We will extend this space to a bigger one by adding a dense set of computable points. Since for this purpose we return to the details of Schr¨oder’s proof we outline it in this section using a slightly modified termi- nology.

A T0-space X = (*X, τ* ) is *regular*, if for every *x* ∈ *X* and every neighbour- hood *V* of *x* there exists a neighbourhood *U* of *x* such that *x* ∈ *U*¯ ⊆ *V* ([[4](#_bookmark32)]).

By a simple argument, every regular space is a a T2-space (Hausdorff space). A T0-space X = (*X, τ* ) is *normal*, if for every pair (*A, A*') of disjoint closed sets there is a pair (*O, O*') of disjoint open sets such that *A*⊆*O* and *A*'⊆*O*'. The following definition is equivalent to that from [[9](#_bookmark34)].

Definition 4.1 [computably regular space]

A *computably regular space* is a computable T0-space X = (*X, τ, β, ν*) for which there is a computable function t3 : ⊆ Σ∗ × Σ∗ → Σ*ω* such that for *R* := dom(t3),

1. (∀*v* ∈ dom(*ν*))*, ν*(*v*)=

(*u,v*)∈*R*

*ν*(*u*)*,* and

1. (∀(*u, v*) ∈ *R*) *ν*(*u*) ⊆ *ψ>*(t3 (*u, v*)) ⊆ *ν*(*v*)

Every computably regular space is regular. Every countably based regular

space is normal [[4](#_bookmark32)]. We prove a computable version.

Definition 4.2 [computably normal]

A *computably normal* space is a computable T0-space such that the multifunc- tion T4 : ⊆*τc* × *τc* ⇒ *τ* × *τ* defined by

dom(T4) := {(*A, B*) ∈ *τc* × *τc* | *A* ∩ *B* = ∅}*,*

(*OA, OB*) ∈ T4 (*A, B*): ⇐⇒ *A*⊆*OA* ∧ *B*⊆*OB* ∧ *OA* ∩ *OB* = ∅*,*

is (*ψ>, ψ>, θ<, θ<*)–computable.

Theorem 4.3 (computably regular implies computably normal)

*Every computably regular space is computably normal.*

In the metrization proof in [[9](#_bookmark34),[4](#_bookmark32)] a sequence of pseudometrics is defined each of which is constructed via a sequence of sets inducing contour levels on the space *X*.

Definition 4.4 (i) Let *r* : N → {*q* ∈ Q | 0 ≤ *q* ≤ 1} be a bijective standard numbering with *r*0 =0 and *r*1 =1

(ii) and define the set of *contour maps* on X by

CM := {(*Vi, Ci*)*i*∈N | *Vi* ∈ *τ, Ci* ∈ *τc, Vi* ⊆ *Ci* ⊆ *Vj* if *ri < rj,* for *i, j* ∈ N} *.*

For any sequence (*Vi, Ci*)*i*∈N ∈ CM, the boundary of each set *Vi* can be considered as a contour line of level *ri* ∈ Q. Since these lines do not cross, the sequence (*Vi, Ci*)*i*∈N ∈ CM can be considered as a contour map on the space X with a hollow *V*0 of level 0 and the top of a hill *C*1 of level 1. For a computably normal space, from *V*0 and *V*1 a contour map (*Vi, Ci*)*i*∈N ∈ CM can be computed.

Lemma 4.5 *Let* X = (*X, τ, β, ν*) *be computably normal space. Deﬁne* SEQ :

⊆*τc* × *τc* ⇒ (*τ* × *τc*)N *by*

dom(SEQ) := {(*A, B*) ∈ *τc* × *τc* | *A* ∩ *B* = ∅}*,*

(*Vi, Ci*)*i*∈N ∈ SEQ(*A, B*): ⇐⇒ (*Vi, Ci*)*i*∈N ∈ CM*, A* ⊆ *V*0 *and V*1 = *Bc.*

*Then* SEQ *is* (*ψ>, ψ>,* [*θ<, ψ>*]*ω*)*–computable.*

A contour map (*Vi, Ci*)*i*∈N ∈ CM can be “extended” to a continuous real function on X.

Lemma 4.6 *Let* X = (*X, τ, β, ν*) *be computably normal space and deﬁne*

SF : ⊆(*τ* × *τc*)N → C(*X,* [0*,* 1]) *by*

dom(SF) = CM*,*

SF((*V ,C* )

)(*x*)= ⎧⎨ inf{*rs* | *x* ∈ *Vs*} *if x* ∈ *V*1 *.*

*i i i*∈N

⎩

1 *otherwise*

*Then* SF *is* ([*θ<, ψ>*]*ω,* [*δ*X → *ρ*])*–computable,* SF((*Vi, Ci*)*i*∈N)[*V*0] = {0} *and*

SF((*Vi, Ci*)*i*∈N)[*X* \ *V*1]= {1}*.*

As a by-product we get a computable Urysohn theorem.

Corollary 4.7 (computable Urysohn)

*Let* X = (*X, τ, β, ν*) *be computably normal space. Then the multi-function* UR := SF ◦ SEQ *mapping every pair* (*A, B*) *of disjoint closed sets to all continuous functions f* : *X* → [0; 1] *such that f* [*A*] = 0 *and f* [*B*] = 1 *is* (*ψ>, ψ>,* [*δ*X → *ρ*])*—computable.*

For a definition of the representation [*δ*X → *ρ*] see [[13](#_bookmark37)]. Let X = (*X, τ, β, ν*) be a computably regular space and let *k* '→ (*uk, vk*) be a canonical numbering of the r.e. set *R* = dom(*t*3) from Definition [4.1](#_bookmark12).

Lemma 4.8 *There is a sequence* ((*V k, Ck*)*i*∈N)*k*∈N *of contour maps such that*

*i* *i*

1. *ν*(*uk*)⊆*V k and ν*(*vk*)= *V k,*

0 1

1. (*k, i*) '→ (*V k, Ck*) *is* (*ν*N*, ν*N*,* [*θ<, ψ>*])*–computable.*

*i* *i*

Proof: For each *k* compute some element

(*V k, Ck*)*i*∈N ∈ SEQ(*ψ>*(*t*3(*uk, vk*)*, ν*(*vk*)*c*)) *.*

*i* *i*

Notice that *U* '→ *Uc* is (*ν, ψ>*)–computable.

Lemma 4.9 *For the computable sequence of contour maps from Lemma* [*4.8*](#_bookmark13) *deﬁne sequences of functions* (*fk*)*k and* (*dk*)*k of real functions on X by*

1. *fk* := SF((*V k, Ck*)*i*∈N)*,*

*i* *i*

1. *dk*(*x, y*) := | *fk*(*x*) − *fk*(*y*) | *.*

*Then for each k, dk is a continuous (w.r.t. τ) pseudometric on X and*

(*k, x*) '→ *fk*(*x*) *is* (*ν*N*, δ*X*, ρ*)*–computable, and*

(*k, x, y*) '→ *dk*(*x, y*) *is* (*ν*N*, δ*X*, δ*X*, ρ*)*–computable.*

The proof is obvious. From the sequenc (*dk*)*k* of pseudometrics a com- putable metric can be defined.

Theorem 4.10 (effective metrization)

*The function d* : *X* × *X* → R *deﬁned by*

1. *d*(*x, y*) := Σ 2−*kdk*(*x, y*)

∞

*k*=0

*is a* (*δ*X*, δ*X*, ρ*)*–computable metric on X inducing the topology τ.*

# Computable Dense Subsets and Embedding to Com- putable Metric Spaces

A computable metric space [[12](#_bookmark38),[13](#_bookmark37)] is a tuple M = (*M, d, A, α*) such that (*M, d*) is a metric space and *α* : ⊆ Σ∗ → *A* is a notation with recursive domain of a (countable) dense subset *A*⊆*M* such that the distance on *A*, (*a, b*) '→ *d*(*a, b*), is (*α, α, ρ*)–computable. The canonical representation of a computable metric space is the Cauchy representaion defined by *δC*(*p*) = *x*, iff *p* = *ι*(*u*0)*ι*(*u*1)*ι*(*u*2) *...* such that (∀*i*)*d*(*x, α*(*ui*)) ≤ 2−*i*. In particular, the elments *a* ∈ *A* are *δC*–computable.

In the previous section for a computably regular space X = (*X, τ, β, ν*) we have constructed a (*δ*X*, δ*X*, ρ*)–computable metric (Definition [4.10](#_bookmark15)) which generates the topology *τ* . But this does not mean that we have found a *com- putable metric space* since we do not have a dense set of computable points where *computable* means *δ*X–computable such that the Cauchy representation is equivalent to *δ*X. In general, a computably regular T0-space X with com- putable metric *d* may have no *δ*X–computable points at all. But there is hope to find a dense set of computable points in its metric completion.

Example 5.1 Consider the computable T0-space X = (R*, τ*R*,* Int*, I*) (Defini- tion [3.1](#_bookmark3)) where *τ*R is the Euclidean topology and *I* is a standard notation of the set Int of open intervals with rational end-points. Let *δ*X be the “topological” standard representation of X (Definition [3.2](#_bookmark4)). On the other hand, for the real numbers there is the computable metric space M = (R*, d,* Q*, ν*Q) with Cauchy representation *ρ*. Since *δ*X ≡ *ρ*, *νQ* is a notation of a set of *δ*X–computable points.

Let X' = (R'*,τ* ' *,* Int'*, I*') be the space obtained from X by deleting all the *ρ*–computable points. Since *ν*(*u*)⊆*ν*(*v*) ⇐⇒ *ν*'(*u*)⊆*ν*'(*v*), X' is a computable T0-space. The embedding *x* '→ *x* from X' to X is (*δ*X' *, δ*X)–computable and its inverse is (*δ*X*, δ*X' )–computable. Therefore, *X*' cannot contain any *δ*X' –

R

computable point, since the computable embedding would map it to a *δ*X– computable, hence *ρ*–computable point.

There are various methods of metric completion. Embedding X into the (complete) Banach space *B*(*X*) of bounded functions from *X* into R (4.41 in [[8](#_bookmark33)]) requires a computability concept on *B*(*X*) which is a new problem. For effective completion via Cauchy sequences on a countable dense set (19.33 in [[8](#_bookmark33)], [[13](#_bookmark37)]) we need to know (computable) points, but we do not have such points. A third method can be derived from Cauchy’s intersection property: A metric space is complete, iff *n Sn* /= ∅ for *every* descending sequence of nonempty closed sets *Sn* such that dm(*Sn*) ≤ 2−*n*. A completion of our metric space (*X, d*) can be constructed by considering all (or sufficiently many) nested sequences of closed sets, introducing a pseudometric on these sequences and factorizing. It turns out that in our case we can compute *suﬃciently many* sequences in order to find a dense set of computable points in a “partial completion” of (*X, d*).

Notice, that so far our approach to computable topology was pointless. We have handled the notation *ν* of basic sets but have not considered any concrete points of the space X. Points occur only hypothetically. For example, if a function *f* is (*δ*X*, δ*X)–computable, then there is a Tpe-2 machine, which from a name *p* ∈ Σ*ω* of some point *x* computes some name of *f* (*x*). By our concept of realization, the machine works correctly for correct inputs but may behave arbitrarily otherwise. However, the machine is not able to check whether an input sequence is a correct name. It is the “user” who must “understand” the representation *δ*X and who is “responsible” himself for supplying the machine with correct names of points.

In this section we extend our computably regular space X = (*X, τ, β, ν*) in such a way that the extension contains a dense set of computable points with computable distances. We start from the computable sequence (*fk*)*k*∈N of contour functions from Lemma [4.9](#_bookmark14). In the following we consider implicitly *ν*N and some standard notation of the finite sequences of rational numbers. We begin with a technical lemma.

Lemma 5.2

*There is a computable function h* : (*v, p, m*) '→ (*u,* (*a*0*, b*0*,... , am, bm*)) *mapping every v* ∈ dom(*ν*)*, every p such that δ*X(*p*) ∈ *ν*(*v*) *and every m* ∈ N *to some word u* ∈ dom(*ν*) *and a sequence of rational numbers such that for all k* ≤ *m,*

1. *δ*X(*p*) ∈ *ν*(*u*)⊆*ν*(*u*)⊆*ν*(*v*) *,*
2. 0 *< bk* − *ak* ≤ 2−*m ,*
3. *ν*(*u*)⊆*f* −1[*ak*; *bk*] *.*

*k*

Proof: First, from *v* and *p* some *u*' can b e computed such that *δ*X(*p*) ∈ *ν*(*u*') and (*u*'*, v*) ∈ *R* (Definition [4.1](#_bookmark12)). Then *ν*(*u*')⊆*ν*(*v*). By Lemma [4.9](#_bookmark14), the function

1. (*m, a, b*) '→ *f* −1(*a*; *b*) is (*ν*N*, ν*Q*, ν*Q*, θ<*)–computable

*m*

(0 ≤ *a < b* ≤ 1). There are rational numbers *a*0*, b*0*,... , am, bm* such that (∀*k* ≤ *m*)0 *< bk* − *ak* ≤ 2−*m* and

*δ*X(*p*) ∈ *ν*(*u*') ∩ *f* −1(*a*0; *b*0) ∩ *...* ∩ *f* −1(*am*; *bm*) /= ∅ *.*

0 *m*

Applying exhaustive search, by Lemma [3.8](#_bookmark6) and ([21](#_bookmark17)) some word *u* and rational numbers *a*0*, b*0*,... , am, bm* can be computed such that (∀*k* ≤ *m*)0 *< bk* − *ak* ≤ 2−*m* and

*δ*X(*p*) ∈ *ν*(*u*)⊆*ν*(*u*') ∩ *f* −1(*a*0; *b*0) ∩ *...* ∩ *f* −1(*am*; *bm*) *.*

0 *m*

([18](#_bookmark17)-[20](#_bookmark17)) follow immediately.

The following similar lemma can be proved accordingly.

Lemma 5.3

*There is a computable function h*' : (*v, m*) '→ (*u,* (*a*0*, b*0*,... , am, bm*)) *mapping every v* ∈ dom(*ν*) *and every m* ∈ N *to some word u* ∈ dom(*ν*) *anda sequence of rational numbers such that for all k* ≤ *m,*

1. *ν*(*u*)⊆*ν*(*u*)⊆*ν*(*v*) *,*
2. 0 *< bk* − *ak* ≤ 2−*m ,*
3. *ν*(*u*)⊆*f* −1[*ak*; *bk*] *.*

*k*

By the following lemma in every base element *ν*(*v*) we can compute a nested sequence (*Ui*)*i*∈N of base elements and a sequence of real numbers (*yk*)*k*∈N such that for every pseudometric *dk*, the diameters dm*k*(*Ui*) and the distances *dk*(*Ui, yk*) converge to 0 rapidly.

Lemma 5.4

*There is a* (1IΣ∗ *,* [*νω, ρω*])*–computable function H* : *w* '→ ((*Ui*)*i*∈N*,* (*yk*)*k*∈N) *mapping each w* ∈ dom(*ν*) *to some sequence of base elements and some se- quence of real numbers such that for all i and for all k* ≤ *i,*

1. *U*0 = *ν*(*w*)*,* (∀*i*) *Ui*+1⊆*Ui ,*
2. dm*k*(*Ui*) ≤ 2−*i ,*
3. sup{|*fk*(*x*) − *yk*|| *x* ∈ *Ui*}≤ 2−*i.*

Proof: We compute a sequence (*ui*)*i* of (names of) base elements and for each

*k* a sequence ((*ak*; *bk*))*i*∈N of intervals with rational endpoints inductively as

*i i*

follows.

i = 0 : *u*0 := *w*, *ak* =0 and *bk* =1 for all *k*.

0 0

i> 0 : With the function *h*' from Lemma [5.3](#_bookmark18) define

(28()*ui,* (*a*0*, b*0*,... , ai, bi*)) := *h*'(*ui*−1*, i*)*,* and *ak* =0 and *bk* =1 for *k > i .*

*i i i i i* *i*

Define *Ui* := *ν*(*ui*). Then ([25](#_bookmark19)) follows from ([22](#_bookmark18)). By ([24](#_bookmark18)), *Ui*⊆*f* −1[*ak*; *bk*] for

*k i i*

*k k*

all *i, k*. Therefore by ([25](#_bookmark19)),

*i*≤*m*[*ai* ; *bi* ] /= ∅ for all *k, m*. By ([28](#_bookmark19)) and ([23](#_bookmark18)),

0 ≤ *bk* − *ak* ≤ 2−*i* for *k* ≤ *i*. Therefore, *yk* such that {*yk*} = [*ak*; *bk*] exists.

*i*

*i*

*i*∈N

*i*

*i*

([27](#_bookmark19)) follows from *yk* ∈ [*ak*; *bk*], 0 ≤ *bk* − *ak* ≤ 2−*i* for *k* ≤ *i* and *Ui*⊆*f* −1[*ak*; *bk*].

*i i i i k i i*

Since 0 ≤ *bk* − *ak* ≤ 2−*i* for *k* ≤ *i*, the sequence of intervals converges rapidly

*i* *i*

such that the real numbers *yk* can be computed.

([26](#_bookmark19)) follows from *Ui*⊆*f* −1[*ak*; *bk*] and 0 ≤ *bk* − *ak* ≤ 2−*i* for *k* ≤ *i*.

*k i i i* *i*

The following lemma can be proved accordingly by using Lemma [5.2](#_bookmark17) in- stead of Lemma [5.3](#_bookmark18)

Lemma 5.5

*There is a* (1IΣ*ω ,* [*νω*

*, ρω*])*–computable function H*'

: *p* '→ ((*Ui*)*i*∈N*,* (*yk*)*k*∈N)

*mapping each p* ∈ dom(*δ*X) *to some sequence of base elements and some se-*

*quence of real numbers such that for all i and for all k* ≤ *i,*

1. *U*0 = *ν*(*w*)*,* (∀*i*) *δ*X(*p*) ∈ *Ui*+1⊆*Ui ,*
2. dm*k*(*Ui*) ≤ 2−*i ,*
3. sup{|*fk*(*x*) − *yk*|| *x* ∈ *Ui*}≤ 2−*i.*

Now we extend our computably regular space X = (*X, τ, β, ν*) with com- putable metric *d* from Theorem [4.10](#_bookmark15) to some space containing the limits of the sequences computed above. It is a “partial” completion via some nested sequences of closed balls. For a metric *d* on *M* and *A, B*⊆*M* we define, as usual, *d*(*A, B*) := inf*a*∈*A,b*∈*B d*(*a, b*).

Definition 5.6 [partial completion]

* 1. Define a pseudometric space (*S, dS*) by

*S* := {(*Ui*)*i*∈N | *Ui* ∈ *β,* dm(*Ui*) *<* 2−*i, Ui*+1⊆*Ui* for all *i* ∈ N} *, dS*((*Ui*)*i*∈N*,* (*Vi*)*i*∈N) := sup *d*(*Ui, Vi*) *.*

*i*∈N

* 1. Let the metric space (*X*˜*, d*˜) be the factorization of the pseudometric space (*S, dS*) with respect to ∼, that is, (*X*˜*, d*˜)= (*S/*∼*, dS/*∼), where

(*Ui*)*i*∈N ∼ (*Vi*)*i*∈N ⇐⇒ sup *d*(*Ui, Vi*)= 0 *.*

*i*∈N

* 1. Let *τ*˜ be the topology generated by the metric *d*˜.

Lemma 5.7

*The spaces* (*S, dS*) *and* (*X*˜*, d*˜) *are well-deﬁned.*

Proof: Obviously, *d*(*s, s*') ≥ 0, *d*(*s, s*)= 0 and *d*(*s, s*')= *d*(*s*'*, s*) for all *s, s*' ∈

*S*. It remains to prove the tiangle inequality for (*Ui*)*i*∈N*,* (*Vi*)*i*∈N*,* (*Wi*)*i*∈N ∈ *S*. For all *i* ∈ N and all *x* ∈ *Ui, y*'*, y*'' ∈ *Vi,z* ∈ *Wi*,

*d*(*x, z*) ≤ *d*(*x, y*')+ *d*(*y*'*, y*'')+ *d*(*y*''*, z*)

≤ *d*(*x, y*')+ 2−*i* + *d*(*y*''*, z*) since dm(*Vi*) *<* 2−*i*. This implies

*d*(*Ui, Wi*)= inf

*x*∈*Ui*

*z*∈*Wi*

*d*(*x, z*) ≤ *d*(*x, y*')+ 2−*i* + *d*(*y*''*, z*) *.*

Since this is true for all *x, y*'*, y*''*, z*,

*d*(*Ui, Wi*) ≤ inf

*x*∈*Ui*

*y*'∈*Vi*

*d*(*x, y*')+ 2−*i* + inf

*y*''∈*Vi*

*z*∈*Wi*

*d*(*y*''*, z*)= *d*(*Ui, Vi*)+ 2−*i* + *d*(*Vi, Wi*) *.*

We obtain

*d*(*Ui, Wi*) ≤ sup *d*(*Uj, Vj*)+ 2−*i* + sup *d*(*Vj, Wj*)

*j j*

and therefore,

sup *d*(*Uj, Wj*)= sup(*d*(*Uj, Wj*) − 2−*j*) ≤ sup *d*(*Uj, Vj*)+ sup *d*(*Vj, Wj*) *,*

*j j j j*

hence

*dS*((*Ui*)*i*∈N*,* (*Wi*)*i*∈N) ≤ *dS*((*Ui*)*i*∈N*,* (*Vi*)*i*∈N)+ *dS*((*Vi*)*i*∈N*,* (*Wi*)*i*∈N) *.*

The metric space (*X*˜*, d*˜) is obtained from the pseudometric space by identify- ing points of distance 0.

We introduce a canonical representation for the space (*X*˜*, d*˜) as follows.

Definition 5.8

Define a representation *δ*˜ of *X*˜ by

dom(*δ*˜) := {*ι*(*u*0)*ι*(*u*1) *...* | ((*ν*(*u*0)*, ν*(*u*1)*,.. .*) ∈ *S*} *,*

*δ*˜(*ι*(*u*0)*ι*(*u*1) *.. .*) := (*ν*(*u*0)*, ν*(*u*1)*,.. .*)*/*∼ *.*

Since the set *S* in Definition [5.6](#_bookmark21) does not contain all descending sequences of closed sets (or balls) in (*X, d*), the space (*X*˜*, d*˜) may be a proper subspace

of the completion of (*X, d*) (up to isometric mappings). But the space (*X*˜*, d*˜) already has a computable dense sequence of computable points. Furthermore, the space (*X, d*) can be embedded isometrically into (*X*˜*, d*˜). The inverse of

this embedding is continuous. For computability we introduce some further axiom.

Definition 5.9 [embedding]

Define a function emb : *X* → *X*˜ by

emb(*x*) := (*Ui*)*i/*∼ such that (*Ui*)*i* ∈ *S* and (∀*i*)*x* ∈ *Ui .*

Theorem 5.10

1. *The embedding function* emb *is well-deﬁned, isometric and* (*δ*X*, δ*˜)*– computable.*
2. *Its inverse is* (*δ*˜*, δ*X)*-continuous.*

Proof:

Prop. 1: For each *p* ∈ dom(*δ*X) a sequence (*Vj*)*j* ∈ *S* can be computed such that (∀*j*)*δ*X(*p*) ∈ *Vj*.

Proof 1: Let (*Ui*)*i* be the sequence computed by *H*' from *p* (Lemma [5.5](#_bookmark20). Define *Vj* := *Uj*+2. By ([29](#_bookmark20)), (∀*j*)*δ*X(*p*) ∈ *Vj*. It remains to show dm(*Vj*) ≤ 2−*j*. Suppose *x, y* ∈ *Vj*. Since *x, y* ∈ *Ui*+2 by ([30](#_bookmark20)),

*d*(*x, y*)= Σ

*k*≤*j*+2

2−*kdk*(*x, y*)+ Σ

*k>j*+2

2−*kdk*(*x, y*)

≤ Σ

*k*≤*j*+2

2−*k*2−*j*−2 + Σ

*k>j*+2

2−*k*

≤ 2 · 2−*j*−2 + 2−*j*−2 *<* 2−*j .*

Therefore, dm(*Vj*) ≤ 2−*j*.

Prop. 2: If (*Vj*)*j,* (*Wj*)*j* ∈ *S*, (∀*j*)*x* ∈ *Vj* and (∀*j*)*y* ∈ *Wj*.

Then *dS*((*Vj*)*j,* (*Wj*)*j*)= *d*(*x, y*).

Proof 2: For all *j*, *d*(*Vj, Wj*) ≤ *d*(*x, y*) (since *x* ∈ *Vj* and *y* ∈ *Wj*), hence

*dS*((*Vj*)*j,* (*Wj*)*j*)= sup*j d*(*Vj, Wj*) ≤ *d*(*x, y*). Furthermore, for all *j*, all *xj* ∈ *Vj* and *yj* ∈ *Wj*,

*d*(*x, y*) ≤ *d*(*x, xj*)+ *d*(*xj, yj*)+ *d*(*yj, y*)

≤ 2−*j* + *d*(*xj, yj*)+ 2−*j*

since dm(*Vj*)*,* dm(*Wj*) ≤ 2−*j*. Since this is true for all *xj* ∈ *Vj* and *yj* ∈ *Wj*,

*d*(*x, y*) ≤ 2 · 2−*j* + *d*(*Vj.Wj*) ≤ 2−*j* + *dS*((*Vj*)*j,* (*Wj*)*j*) Since this is true for all *j*, *d*(*x, y*) ≤ *dS*((*Vj*)*j,* (*Wj*)*j*).

Let *δ*X(*p*) = *δ*X(*q*) and let (*Vj*)*j* ∈ *S* and (*Wj*)*j* ∈ *S* be the sequences computed from *p* and *q*, repectively, according to Prop. 1. Then by Prop. 2, *dS*((*Vj*)*j,* (*Wj*)*j* = *d*(*δ*X(*p*)*, δ*X(*q*)) = 0. Therefore, the function emb is well- defined and (*δ*X*, δ*˜)–computable.

Immediately from Prop. 2 we conclude Therefore, the embedding is isometric.

*d*˜(emb(*x*)*,* emb(*y*)) = *d*(*x, y*).

If *δ*˜(*p*)= emb(*x*) ∈ *X*˜, then *p* lists a neighbourhood base of *x*. Therefore the identity is a computable (*δ*˜*, δb* )-realization of emb−1 and emb−1 is (*δ*˜*, δ*X)-continuous by Lemma [3.10](#_bookmark10).

X

Possibly neither emb−1 is (*δ*˜*, δ*X)–computable nor the metric *d*˜ is (*δ*˜*, δ*˜*, ρ*)– computable on the set emb(*X*)⊆*X*˜. By the next theorem, however, there is

a notation *α* of a dense subset of *X*˜

computable on its range.

such that the distance

*d*˜ is (*α, α, ρ*)–

Theorem 5.11

*There is a notation α* : ⊆Σ∗ → *A with recursive domain of some dense set A*

*in* (*X*˜*, d*˜) *such that α* ≤ *δ*˜ *and d*˜ *is* (*α, α, ρ*)*–computable on A.*

Proof: We use the function *H* from Lemma [5.4](#_bookmark19). Let dom(*α*) := dom(*ν*). For each *w* ∈ dom(*ν*) let

1. *α*(*w*) := (*Uj*+2)*j/*∼ wherer (*Ui*)*i* = pr1*H*(*w*) *.*

As in the proof of Proposition 1 in the proof of Theorem [5.10](#_bookmark23), we can show (*Uj*+2)*j* ∈ *S*. Therefore, *α* is well-defined and *α* ≤ *δ*˜.

We show that range(*α*) is dense in *X*˜. Let *x*˜ ∈ *X*˜. Then *x*˜ = (*ν*(*wi*))*i/*∼

for some *w*0*, w*1*,...* ∈ dom(*ν*). For *k* ∈ N, *α*(*wk*)= (*Vi*)*i/*∼ for some sequence (*Vi*)*i* ∈ *S* such that *Vi*⊆*ν*(*wk*) for all *i*. Then

1. *d*˜((*ν*(*wi*))*i/*∼*,* (*Vi*)*i/*∼)= sup *d*(*ν*(*wi*)*, Vi*) ≤ dm(*ν*(*wk*))) ≤ 2−*k .*

*i*

We show computability of *d*˜ on *A*. For *H* from Lemma [5.4](#_bookmark19) and *u, v* ∈

dom(*ν*)= dom(*α*) let

*H*(*u*)= ((*Ui*)*i,* (*yj*)*j*)*,*

*H*(*v*)= ((*Vi*)*i,* (*zj*)*j*) *.*

For *y* ∈ *Ui* and *z* ∈ *Vi* (since in general | |*a* − *b*|− |*c* − *d*|| ≤ |*a* − *c*| + |*b* − *d*|), *d*(*y, z*) − Σ 2−*k*|*yk* − *zk*|

*k*≥0

= Σ 2−*kdk*(*y, z*)+ Σ 2−*kdk*(*y, z*) − Σ 2−*k*|*yk* − *zk*|− Σ 2−*k*|*yk* − *zk*|

*k*≤*i*

*k>i*

*k*≤*i*

*k>i*

≤ Σ 2−*k*|*fk*(*y*) − *fk*(*z*)|− Σ 2−*k*|*yk* − *zk*| + 2−*i*+1

*k*≤*i*

*k*≤*i*

≤ Σ 2−*k* |*fk*(*y*) − *fk*(*z*)|− |*yk* − *zk*| + 2−*i*+1

*k*≤*i*

≤ Σ 2−*k* |*fk*(*y*) − *yk*| + |*fk*(*z*) − *zk*| + 2−*i*+1

*k*≤*i*

≤ Σ 2−*k*(2−*i* + 2−*i*)+ 2−*i*+1 by ([27](#_bookmark19))

*k*≤*i*

≤ 6 · 2−*i*

For *s* := Σ*k*≥0 2−*k*|*yk* − *zk*|, *s* − 2−*i*+3 ≤ *d*(*y, z*) ≤ *s* + 2−*i*+3 and therefore,

*s* − 2−*i*+3 ≤ *d*(*Ui, Vi*) ≤ *s* + 2−*i*+3

since *d*(*Ui, Vi*) = inf{*d*(*y, z*) | *y* ∈ *Ui,z* ∈ *Vi*}. Since *d*(*Ui, Vi*) ≤ *d*(*Ui*+1*, Vi*+1), for all *i*,

We obtain

*s* − 2−*i*+3 ≤ sup *d*(*Uj, Vj*) ≤ *s* + 2−*i*+3 *.*

*j*

*s* = sup *d*(*Uj, Vj*)= *dS*((*Ui*+2)*i,* (*Vi*+2)*i*)= *d*˜(*α*(*u*)*, α*(*v*)) *.*

*j*

Since the sequences (*yi*)*i* and (*zi*)*i* of real numbers can be computed from *u* and *v* and since *yi, zi* ≤ 2 by ([27](#_bookmark19)), *s* = *k*≥0 2−*k*|*yk* − *zk*| = *d*˜(*α*(*u*)*, α*(*v*)) can be computed from *u* and *v*.

Σ

So far for a computably regular space X = (*X, τ, β, ν*) with standard rep- resentation *δ*X we have constructed a computable metric *d* generating *τ* and a metric space X˜ = (*X*˜*, d*˜) with representation *d*˜ such that

* 1. there is a (*δ*X*, δ*˜)–computable isometric embedding emb : *X* → *X*˜,
  2. emb−1 is (*δ*˜*, δ*X)-continuous,
  3. there is a notation *α* of a dense subset *A*⊆*X*˜

with recursive domain such

that *α* ≤ *δ*˜ and *d*˜ is (*α, α, ρ*)–computable on *A*.

For proving *δb*

X

≤*t δ*X, that is, continuous reducibility of *δb*

to *δ*X in

Lemma [3.10](#_bookmark10) we have used the set {(*u, v*) | *ν*(*u*)⊆*ν*(*v*)} as an “oracle”. In many applications there is some similar r.e. oracle for this purpose. In such a

X

X

case, *δb*

X

≡ *δ*X. Since emb−1 is (*δ*˜*, δb* )–computable (proof of Theorem [5.10](#_bookmark23).[ii](#_bookmark24))

the same oracles can be used to prove that emb−1 is (*δ*˜*, δ*X)-continuous or

–computable.

Notice that (*X*˜*, d*˜*, A, α*) is a computable metric space [[13](#_bookmark37)]. The following Theorem summarizes these results.

Theorem 5.12

*Let* X = (*X, τ, β, ν*) *be a computably regular space such that* {(*u, v*) | *ν*(*u*)⊆*ν*(*v*)} *is r.e. Then there is a* (*δ*X*, δ*˜)*–computable isometric embedding to the computable metric space* (*X*˜*, d*˜*, A, α*) *and the inverse of the embedding is* (*δ*˜*, δ*X)*–computable.*

Given a computable metric space we have to consider the associated Cauchy representation (see [[13](#_bookmark37)]).

Definition 5.13

Let *δC* be the Cauchy representation of the computable metric space (*X*˜*, d*˜*, A, α*).

The representations

*δ*˜ and *δC* induce two computability concepts on *X*˜.

They are related as follows.

Theorem 5.14

*δ*˜ ≤ *δC, δC* ≤*t δ*˜*.*

Proof: *δ*˜ ≤ *δC*: In the proof of Theorem [5.11](#_bookmark25) we have already proved that

*A* is dense. Suppose *δ*˜(*p*)= *x* Then by Definition [5.8](#_bookmark22), *p* = *ι*(*w*0)*ι*(*w*1) *...* such

that *x* = (*ν*(*w*0)*, ν*(*w*1)*,.. .*)*/*∼. By ([33](#_bookmark25)) for all *k*, *d*˜(*x, α*(*wk*)) ≤ 2−*k*, hence

*x* = *δC*(*ι*(*w*0)*ι*(*w*1) *.. .*). Therefore, the identity translates *δ*˜ to *δC*.

*δC* ≤*t δ*˜: (Omitted)

Corollary 5.15

*The distance d*˜ *on X*˜

*is* (*δ*˜*, δ*˜*, ρ*)*–computable.*

Proof: This is true since *d*˜ is (*δC, δC, ρ*)–computable and *δ*˜ ≤ *δC*.

# References

1. Brattka, V. and G. Presser, *Computability on subsets of metric spaces*, Theoretical Computer Science 305 (2003), pp. 43–76.

URL <msubsets.html>

1. Brattka, V. and K. Weihrauch, *Computability on subsets of Euclidean space I: Closed and compact subsets*, Theoretical Computer Science 219 (1999), pp. 65–93.

URL <subsets.html>

1. Collins, P., *Continuity and computability on reachable sets*, Theoretical Computer Science 341

(2005), pp. 162–195.

1. Engelking, R., “General Topology,” Sigma series in pure mathematics 6, Heldermann, Berlin, 1989.
2. Grubba, T. and K. Weihrauch, *A computable version of Dini’s theorem for topological spaces*, in: T. Grubba, P. Hertling, H. Tsuiki and K. Weihrauch, editors, *Computability and Complexity in Analysis*, Informatik Berichte 326 (2005), pp. 117–129, proccedings, Second International Conference, CCA 2005, Kyoto, Japan, August 25–29, 2005.
3. Grzegorczyk, A., *Computable functionals*, Fundamenta Mathematicae 42 (1955), pp. 168–202.
4. Lacombe, D., *Extension de la notion de fonction r´ecursive aux fonctions d’une ou plusieurs* *variables r´eelles I*, Comptes Rendus Acad´emie des Sciences Paris 240 (1955), pp. 2478–2480, th´eorie des fonctions.
5. Schechter, E., “Handbook of Analysis and Its Foundations,” Academic Press, San Diego, 1997.
6. Schr¨oder, M., *Effective metrization of regular spaces*, in: K.-I. Ko, A. Nerode, M. B. Pour- El, K. Weihrauch and J. Wiedermann, editors, *Computability and Complexity in Analysis*, Informatik Berichte 235 (1998), pp. 63–80, cCA Workshop, Brno, Czech Republic, August, 1998.
7. Smyth, M., *Topology*, in: S. Abramsky, D. Gabbay and T. Maibaum, editors, *Handbook of Logic in Computer Science, Volume 1* (1992), pp. 641–761.
8. Stoltenberg-Hansen, V. and J. V. Tucker, *Concrete models of computation for topological* *algebras*, Theoretical Computer Science 219 (1999), pp. 347–378.
9. Weihrauch, K., *Computability on computable metric spaces*, Theoretical Computer Science 113

(1993), pp. 191–210, fundamental Study.

1. Weihrauch, K., “Computable Analysis,” Springer, Berlin, 2000.
2. Weihrauch, K., *Multi-functions on multi-represented sets are closed under flowchart programming*, in: T. Grubba, P. Hertling, H. Tsuiki and K. Weihrauch, editors, *Computability* *and Complexity in Analysis*, Informatik Berichte 326 (2005), pp. 267–300, proccedings, Second International Conference, CCA 2005, Kyoto, Japan, August 25–29, 2005.
3. Yasugi, M., T. Mori and Y. Tsujii, *Effective properties of sets and functions in metric spaces with computability structure*, Theoretical Computer Science 219 (1999), pp. 467–486.
4. Zhong, N. and K. Weihrauch, *Computability theory of generalized functions*, Journal of the Association for Computing Machinery 50 (2003), pp. 469–505.
5. Zhou, Q., *Computable real-valued functions on recursive open and closed subsets of Euclidean space*, Mathematical Logic Quarterly 42 (1996), pp. 379–409.
6. Ziegler, M., *Computability on regular subsets of Euclidean space*, Mathematical Logic Quarterly

48 (2002), pp. 157–181.

1. Ziegler, M., *Computable operators on regular sets*, Mathematical Logic Quarterly 50 (2004),

pp. 392–404.