CDMTCS Research Report Series

On Computable Theoretic Properties of Structures and Their Cartesian Products

Bakhadyr Khoussainov
Department of Computer Science
University of Auckland

CDMTCS-125
March 2000

Centre for Discrete Mathematics and
Theoretical Computer Science

On Computable Theoretic Properties of Structures and Their Cartesian Products

Bakhadyr Khoussainov*

Abstract: In this paper we show that for any set $X \subset \omega$ there exists a structure $\mathcal{A}$ that has no presentation computable in X such that $\mathcal{A}^2$ has a computable presentation. We also show that there exists a structure $\mathcal{A}$ with infinitely many computable isomorphism types such that $\mathcal{A}^2$ has exactly one computable isomorphism type.

Keywords: Computable structure, Computable isomorphism.

1 Introduction, Basic Notions and Results

The Cartesian product operation occurs in many areas of mathematics, in algebra, model theory, topology, etc. The operation usually preserves many properties of the underlying structures from a certain class and is a source of constructions of new structures from given ones. For example, the Cartesian product of any two structures from a given variety (e.g. from the variety of groups, rings, etc.) is a structure from the same variety. In general, it is of a particular interest to find the relationship between structures and their Cartesian products. This is usually done by studying whether a certain property is shared by both a given class of structures and their Cartesian products. In this paper we study the relationship between computability–theoretic properties of structures and their cartesian products. The goal of this paper is to answer the following two questions:

Question 1 *How complicated, from the computability theory point of view, can a structure $\mathcal{A}$ be if $\mathcal{A}^2$ has a computable presentation? In particular, can $\mathcal{A}$ have a computable presentation if $\mathcal{A}^2$ has one?*

*Computer Science Department, The School of Mathematical and Information Sciences, The University of Auckland, Auckland, New Zealand. E-mail: bmk@cs.auckland.ac.nz

Question 2 *What is the relationship between the computable isomorphism types of $\mathcal{A}$ and $\mathcal{A}^2$? In particular, if $\mathcal{A}^2$ is computably categorical is then $\mathcal{A}$ computably categorical?*

Theorem 1 and Theorem 2, formulated at the end of the introduction, give answers to both questions, respectively. Their proofs are given in Section 2 and Section 3.

We now present some basic notions from the theory of computable structures. Fix a computable language with the sign for equality.

Definition 1 *A structure of the language is* **computable** *if its domain and the atomic diagram are computable sets.*

For example, any structure of a finite language, whose domain is ω and whose predicates and operations are computable, is a computable structure. If a structure $\mathcal{B}$ is isomorphic to a computable structure $\mathcal{A}$ then $\mathcal{B}$ is called **computably presentable**. Then any isomorphism from $\mathcal{B}$ into a computable structure is called a **computable presentation** of $\mathcal{B}$. Of course, structures with uncountable domains are not computably presentable, as well as there are countable structures without computable presentations.

A natural way to identify two computable structures is to say that they are computably isomorphic. Here is a formal definition.

Definition 2 *Computable structures $\mathcal{A}$ and $\mathcal{B}$ are* **computably isomorphic** *if there exists a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.*

This definition allows one to introduce the notion of computable isomorphism types of structures similar to how classical algebra or model theory introduces the isomorphism types of structures. Thus, if $\mathcal{A}$ and $\mathcal{B}$ are computably isomorphic then we say that they have the same **computable isomorphism type**.

Definition 3 *The* **dimension** *of a computable structure $\mathcal{A}$ is the number of its computable isomorphism types. $\mathcal{A}$ is* **computably categorical** *if its dimenion is* 1*, i.e. every computable structure $\mathcal{B}$ isomorphic to $\mathcal{A}$ is computably isomorphic to $\mathcal{A}$.*

Rational numbers with the natural order, atomless Boolean algebras, finitely generated computable algebras are all examples of computably categorical structures.

Now we recall that $\mathcal{A}^2$ is the structure whose domain is the set of all pairs (a, b) from the domain of $\mathcal{A}$ such that for any n-ary predicate P of the language, P holds true on any given n–tuple of pairs of elements $((a_1, b_1), \ldots, (a_n, b_n))$ if and only if P holds on the n-tuples $(a_1, \ldots, a_n)$ and $(b_1, \ldots, b_n)$ in structure $\mathcal{A}$. The language operations are defined in a similar manner. If now $\mathcal{A}$ is a computable structure then clearly $\mathcal{A}^2$ is also computable. The corollary of Theorem 1 gives an example where the converse does not hold. On the other hand, if $\mathcal{A}$ is computably categorical then $\mathcal{A}^2$ may or may not be computably categorical. For example, if $\mathcal{A}$ is a structure of pure equality symbol then clearly $\mathcal{A}$ is computably categorical and so is $\mathcal{A}^2$. However, the structure (ω, S), where S is the successor function on ω, is clearly computably categorical, and it can be proved that the Cartesian product of the structure (ω, S) to itself is not computably categorical. Our Theorem 2 gives an example of a noncomputably categorical structure $\mathcal{A}$ such that $\mathcal{A}^2$ is computably categorical thus answering the second question asked above.

In some cases it can be proved that computatbility of $\mathcal{A}^2$ implies computable presentablity of $\mathcal{A}$. We say that $\mathcal{B}$ is **existentially definable** in $\mathcal{A}$ if there are existential formulas $D(\bar{x}, \bar{a})$, $E(\bar{x}, \bar{y}, \bar{a})$, and quantifier free formulas $\Phi_i(\bar{x}_1, \ldots, \bar{x}_{n_i}, \bar{a})$, where $\bar{a}$ is tuple in $\mathcal{A}$, and $\bar{y}$, $\bar{z}$, and all $\bar{x}_i$ have the length as $\bar{x}$, such that the following properties hold:

1. The formula $D(\bar{x}, \bar{a})$ defines the domain D of all tuples in $\mathcal{A}$ that make the formula $D(\bar{x}, \bar{a})$ true,

2. $E(\bar{y}, \bar{z}, \bar{a})$ defines an equavalence relation on D,

3. Uniformly on i, each $\Phi_i(\bar{x}_1, \ldots, \bar{x}_{n_i}, \bar{a})$ defines the predicate (or the graph of the operation) P_i of the language of $\mathcal{A}$ so that P_i is well–defined on the E–equivalence classes (for a binary relation P_i for example this means that if $E(\bar{y}_1, \bar{z}_1, \bar{a})$ and $E(\bar{y}_2, \bar{z}_2, \bar{a})$ are true then $P(\bar{y}_1, \bar{y}_2)$ is true if and only if $P_i(\bar{z}_1, \bar{z}_2)$ is true),

4. The structure, whose domain is the E–equivalence classes and whose predicates and operations are P_i's defined in the previous item, is isomorphic to $\mathcal{B}$.

It is not hard to see that the following proposition is true:

Proposition 1 *Assume that $\mathcal{B}$ is* **existentially definable** *in $\mathcal{A}$ by formulas $D(\bar{x}, \bar{a})$, $E(\bar{x}, \bar{y}, \bar{a})$, $\Phi_i(\bar{x}_1, \ldots, \bar{x}_{n_i}, \bar{a})$ such that $E(\bar{x}, \bar{y})$ defines a computable relation in $\mathcal{A}$. Then $\mathcal{B}$ is a computably presentable structure.* □

Now we give several examples to which the proposition above can be applied. The first example is due to Denis Hirschfeldt [6].

Corollary 1.1 *Let $\mathcal{R}$ be a commutative ring with a unity element* 1*. Then R^2 has a computable presentation if and only if R has a computable presentation.*

Proof. Consider the formula $x = ax$, where $a = (1, 0)$ and 1 is the unity element of the ring. $E(x, y)$ is the formula $ax = ay$. Note that if $E(x, y)$ holds in $\mathcal{R}^2$ then the left coordinates of x and y coincide. Clearly $E(x, y)$ defines an equivalence relation in the domain of the ring R^2. If $\mathcal{R}^2$ is computable then so is the equivalence relation. It is not hard to see that the operations $+$ and $\times$ of the ring R^2 define well–behaved operations on the E–equivalence classes. By the proposition above, R has a computable presentation.□

Similar to the proof of the above result one can show the following

Corollary 1.2 *A Boolean algebra $\mathcal{A}$ has a computable presentation if and only if $\mathcal{A}^2$ has a computable presentation.* □

The next corollary shows that each structure can be expanded in a natural way so that the computability of $\mathcal{A}$ and the Cartesian product of the expanded structure are equivalent.

Corollary 1.3 *Let $\mathcal{A}$ be a structure of a pure functional language with equality. Expand the structure $\mathcal{A}$ to the structure $\mathcal{A}'$ by adding the binary function Eq so that $Eq(a, b) = a_0$ if $a \neq b$ and $Eq(a, b) = a_1$ if $a = b$, where a_0 and a_1 are two distinct elements in $\mathcal{A}$. If $(A')^2$ has a computable presentation then so does $\mathcal{A}$.*

Proof. Let $(\mathcal{A}')^2$ be computable. Define in this structure the following relation $E(x, y)$:

$$Eq(x, y) = (a_1, a_0) \bigvee E(x, y) = (a_1, a_1).$$

Note that if $E(x, y)$ holds in $(\mathcal{A}')^2$ then the left coordinates of x and y coincide. It is not hard to see that $E(x, y)$ defines an equivalence relation in the domain of the structure $(\mathcal{A}')^2$. If $(\mathcal{A}')^2$ is computable then so is the equivalence relation.

One can check that the operations in $(\mathcal{A}')^2$ are well-behaved on the equivalence classes. Hence by the proposition, the structure $\mathcal{A}'$ has a computable presentation. $\square$

Here are now the formulations of our next results whose proofs will be given in the next two sections. The theorem below and the corollary answer the first question.

Theorem 1 *Let X be a subset of ω. Then there exists a structure $\mathcal{A}$ such that the following properties hold:*

1. *X is Turing equivalent to the set of all existential sentences true in $\mathcal{A}$.*
2. *$\mathcal{A}^2$ has a computable presentation.*

Corollary 1.4 *For every $X \subset \omega$ there exists a structure $\mathcal{A}$ which does not have a presentation computable in X such that $\mathcal{A}^2$ has a computable presentation.*

Proof. Take the double jump X'' of X. By the theorem above there exists a structure $\mathcal{A}$ for which the set of all existential sentences true in $\mathcal{A}$ is Turing equivalent to X'' and $\mathcal{A}^2$ has a computable presentation. The structure $\mathcal{A}$ can not have a presentation computable in X. Otherwise, the set of all existential sentences true in $\mathcal{A}$ would be computable in X'. This would be a contradiction. $\square$

Corollary 1.5 *There exists a structure $\mathcal{A}$ without computable presentations such that $\mathcal{A}^2$ has a computable presentation.*$\square$

Proof. In the proof of the previous corollary take X so that the Turing degree of X is not a c.e. degree. $\square$

The theorem below answers the second question.

Theorem 2 *There exists a computable structure $\mathcal{A}$ with ω computable isomorphism types such that the structure $\mathcal{A}^2$ is computably categorical, that is, $\mathcal{A}^2$ has exactly one computable isomorphism type.*

Finally, we refer the reader interested in the subject of the theory of computable structures to the Handbook of Recursive Mathematics [5] and the Handbook of Computability Theory [2]

2 Proof of Theorem 1

The structure $\mathcal{A}$ will be a graph consisting of the union of finite graphs called **cycles**. We always assume that N is the set of natural numbers without 0. An n–**cycle**, $n \in N$, is a directed graph isomorphic to the following graph:

$$(\{0, 1, 2, \ldots, n-1\}, E),$$

where the edge relation E holds the following pairs only: $(0, 1)$, $(1, 2)$, $(2, 3)$, …, $(n-1, 0)$, (i, a). n is the length of the cycle. For every subset $B \subset N$, we define the graph $\mathcal{G}(B)$ that has the following properties:

1. For each $n \in B$ there are countably many n–cycles in $\mathcal{G}(B)$.
2. $\mathcal{G}(B)$ is a disjoint union of all cycles that belong to $\mathcal{G}(B)$.
3. If x and y are in $\mathcal{G}(B)$ that belong to distinct cycles then there is no edge between x and y.
4. If $\mathcal{G}(B)$ contains an n–cycle then $n \in B$.

It is not hard to see that all the four properties above define the graph $\mathcal{G}(B)$ up to isomorphism.

Consider the following set C:

$$C = \{pq|\ p,\ q \text{ are distinct prime numbers greater than } 2\}.$$

Clearly C is a computable set. Let $c_0, c_1, c_2, \ldots$ be a computable list of all elements from C without repetition. Take a set $X \subset N$. For the set X we want to construct a structure $\mathcal{A}$ whose existence is stated in the theorem.

Let X_1 be the set $\{c_i | i \in X\}$. Clearly X and X_1 have the same Turing degree. Let $\mathcal{A}$ be the structure isomorphic to the graph $\mathcal{G}(N \setminus X_1)$. Now the theorem follows from the following three claims that prove that the graph $\mathcal{A}$ is a desired structure.

Claim 2.1 *The graph $\mathcal{A}$ has a presentation computable in X.*

Proof. Indeed, let us list all elements not in X_1 without repetition:

$$n_0, n_1, n_2, \ldots$$

Uniformly on i construct the graph $\mathcal{G}(\{n_i\})$ such that the domains of $G(\{n_i\})$ and $\mathcal{G}(\{n_j\})$, for all $i \neq j$, have no elements in common. Hence the union of all $G(\{n_i\})$ is a presentation of $\mathcal{A}$. By the construction, the presentation is clearly computable in X. This proves the claim.□

Claim 2.2 *The set X is Turing equivalent to the set $T_{\exists}(\mathcal{A})$ of all existential sentences true in $\mathcal{A}$.*

Proof. Indeed take a sentence ϕ_n, $n \in N$, that states that there are n distinct elements that form an n-cycle. Then $n \in X$ if and only if ϕ_{c_n} does not belong to $T_{\exists}(\mathcal{A})$. This proves the claim.□

Claim 2.3 *The Cartesian product of $\mathcal{A}$ to itself, that is $\mathcal{A}^2$, is isomorphic to $\mathcal{G}(N)$, and hence is computably presentable.*

Proof. Consider the structure $\mathcal{A}^2$. It is not hard to see that $\mathcal{A}^2$ has the following properties:

1. Every element in $\mathcal{A}^2$ belongs to a cycle.
2. No two elements of $\mathcal{A}^2$ that belong to distinct cycles are connected via an edge.

Hence $\mathcal{A}^2$ is a disjoint union of cycles. If $n \notin X_1$, then $\mathcal{G}$ has infinitely many cycles of length n because the Cartesian product of a 1–cycle and an n–cycle is again an n–cycle, and we know that $\mathcal{A}$ has infinitely many n–cycles and 1–cycles. Assume that $n \in X_1$. Then $n = pq$ for some prime $p, q \in \omega \setminus C$. Now note that the Cartesian product of a p–cycle and a q–cycle is a pq–cycle. Since $\mathcal{A}$ has infinitely many p–cycles and q–cycles, $\mathcal{G}$ has infinitely many pq–cycles. Therefore $\mathcal{A}^2$ is isomorphic to $\mathcal{G}(N)$. Clearly, $\mathcal{G}$ has a computable presentation. The claim, and hence the theorem are proved. □

3 Proof of Theorem 2

The structure $\mathcal{A}$ will be a graph expanded with an equivalence relation E on the nodes of the graph. In constructing $\mathcal{A}$ we follow the notions and notations introduced in the proof of Theorem 1. The structure $\mathcal{A}$ will be isomorphic to an infinite disjoint union of graphs of the type $\mathcal{G}(B_i)$, $B_i \subset N$, $i \in \omega$, that is, $\mathcal{A}$ will be of the form:

$$\mathcal{G}(B_0) + \mathcal{G}(B_1) + \mathcal{G}(B_2) + \mathcal{G}(B_3) + \ldots,$$

where $+$ represents the union operation of disjoint graphs. We call the graphs $\mathcal{G}(B_i)$ E**-components**. These components will be the E–equivalence classes. Each E–component, by the definition of $\mathcal{G}(B_i)$, consists of cycles. Consider the set C:

$$C = \{pq|\ p,\ q \text{ are distinct prime numbers greater than } 2\}.$$

We construct structures $\mathcal{A}_i$, $i \in \omega$, that are pairwise isomorphic but not computably isomorphic. The desired structure $\mathcal{A}$ will then be the structure $\mathcal{A}_1$. Let $\phi_0, \phi_1, \phi_2, \ldots$ be an enumeration of all computable partial functions on ω. We construct the structures $\mathcal{A}_i$ by stages and will satisfy the following requirements:

$R_{i,j,k}:$ ϕ_k is not an isomorphism from $\mathcal{A}_i$ into $\mathcal{A}_j$, $i \neq j$, $i, j, k \in \omega$.

A strategy to satisfy the requirement $R_{i,j,k}$ is the following. The construction picks up two isomorphic cycles $c^{(i)}_{i,j,k}$ in $\mathcal{A}_i$ and $c^{(j)}_{i,jk}$ in $\mathcal{A}_j$ that belong to isomorphic E–components and waits for ϕ_k to be defined on $c^{(i)}_{i,j,k}$. These cycles are called **witnesses** for $R_{i,j,k}$. Once picked the witnesses will never be changed. Let $\mathcal{G}(B^{(i)}_{i,j,k})$, $\mathcal{G}(B^{(j)}_{i,j,k})$ be the E–components that contain the cycles $c^{(i)}_{i,j,k}$ and $c^{(j)}_{i,j,k}$, respectively. The components in each $\mathcal{A}_{l,s}$ are also $E_{l,s}$–equivalence classses, $l \leq s$. If $\phi_{k,s}$ maps $c^{(i)}_{i,j,k}$ isomorphically onto $c^{(j)}_{i,j,k}$ at some stage s then we say that $R_{i,j,k}$ **requires attention**. When $R_{i,j,k}$ requires attention the strategy **attacks** $R_{i,j,k}$ by making $\mathcal{G}(B^{(i)}_{i,j,k})$ and $\mathcal{G}(B^{(j)}_{i,j,k})$ not isomorphic, and constructs new isomorphic copies of $\mathcal{G}(B^{(i)}_{i,j,k})$, $\mathcal{G}(B^{(j)}_{i,j,k})$ in $\mathcal{A}_j$ and $\mathcal{A}_i$, respectively. This change is passed then to all other $\mathcal{A}_t$'s which are being constructed. More formally we proceed as follows.

Stage $s+1$. At stage s, the construction has constructed isomorphic structures $\mathcal{A}_{m,s}$, $m \leq s$, such that each $\mathcal{A}_{m,s}$ is of the form:

$$\mathcal{G}(B_{m,0,s}) + \mathcal{G}(B_{m,1,s}) + \mathcal{G}(B_{m,2,s}) + \mathcal{G}(B_{m,3,s}) + \ldots + \mathcal{G}(B_{m,t_s,s}),$$

where each $\mathcal{G}(B_{m,l,s})$ is an equivalence class (E–component) of the equivalence relation $E_{m,s}$. We assume that the witnesses $c^{(i)}_{i,j,k}$ and $c^{(j)}_{i,j,k}$, $i, j, k \leq s$ are all in the structures constructed so far and witnesses for distinct requirements have no entries in common. Take the minimal $R_{i,j,k}$, $i, j, k \leq s$, that requires attention. If it exists then attack it as follows:

1. Extend the E–component $\mathcal{G}(B^{(i)}_{i,j,k})$ by putting countably many n-cycles into it, where n is the first element t in C for which $\mathcal{G}(B^{(i)}_{i,j,k})$

has no a t–cycle. Prohibit putting an n–cycle into the E–component $\mathcal{G}(B_{i,j,k}^{(j)})$. Call n **prohibited number** for the E–component $\mathcal{G}(B_{i,j,k}^{(j)})$.

2. Extend $\mathcal{G}(B_{i,j,k}^{(j)})$ by putting countably many r-cycles into it, where r is the first element t in C for which $\mathcal{G}(B_{i,j,k}^{(j)})$ has no a t–cycle and $t \neq n$. Prohibit putting an r–cycle into $\mathcal{G}(B_{i,j,k}^{(i)})$. Call r **prohibited number** for the E–component $\mathcal{G}(B_{i,j,k}^{(i)})$.

3. Construct new isomorphic copies of $\mathcal{G}(B_{i,j,k}^{(i)})$, $\mathcal{G}(B_{i,j,k}^{(j)})$ in $\mathcal{A}_{j,s}$ and $\mathcal{A}_{i,s}$, respectively. These new added E–components now have the same prohibited numbers as their isomorphic copies. These are also new $E_{i,s}$ and $E_{j,s}$–equivalence classes. Pass this change to every other $\mathcal{A}_{m,s}$ so that all are still isomorphic, $m \leq s$.

4. Extend each E–component in all $\mathcal{A}_{m,s}$, $m \leq s$, by putting into it infinitely many l-cycles, where l is the least unused and unprohibited number for the component.

5. Extend now each $\mathcal{A}_{m,s}$ to $\mathcal{A}_{m,s+1}$, $l \leq s+1$, so that $c_{i,j,k}^{(j)}$, $i, j, k \leq s+1$ all appear in these structures, and make all these structures isomorphic.

If no $R_{i,j,k}$ requires attention then extend now each $\mathcal{A}_{l,s}$ to $\mathcal{A}_{l,s+1}$, $l \leq s+1$, so that $c_{i,j,k}^{(j)}$, $i, j, k \leq s+1$ all appear in these structures and make all these structures isomorphic. This ends the stage $s+1$ of the construction. Set

$$\mathcal{A}_i = \bigcup_s \mathcal{A}_{i,s}.$$

Now we list several properties of these structures in the next series of claims.

Claim 3.1 *Each E–component in every $\mathcal{A}_i$ is either isomorphic to $\mathcal{G}(N)$ or $G(N \setminus \{m\})$ for some $m \in C$.*

Proof. Take any E–component in $\mathcal{A}_i$. If at no stage the construction prohibits a number for the component then the component is isomorphic to $\mathcal{G}(N)$. Otherwise, by the construction the E–component will have only one prohibited number. By the construction the number must belong to C. Moreover, by the construction, no other number is prohibited for the component. This proves the claim. □

Claim 3.2 *Each $R_{i,j,k}$ is satisfied.*

Proof. We first note that if $R_{i,j,k}$ is attacked then, by the construction, $R_{i,j,k}$ is satisfied. So assume that ϕ_k is an isomorphism between $\mathcal{A}_i$ and $\mathcal{A}_j$. Then at some stage s, $\phi_{k,s}(c^{(i)}_{i,j,k}) = c^{(j)}_{i,j,k}$. Hence ϕ_k must be attacked at some stage $s' \geq s$. This is a contradication. The claim is proved. □

Claim 3.3 *The structures $\mathcal{A}_i$ and $\mathcal{A}_j$ are isomorphic but not computably isomorphic.*

Proof. From the previous claim we see that the structures $\mathcal{A}_i$ and $\mathcal{A}_j$ are not computably isomorphic. The construction guarantees that the structures are isomorphic. □

Let $\mathcal{A}$ be the structure isomorphic to $\mathcal{A}_1$. Thus, we see that $\mathcal{A}$ has infinitely many computable isomorphism types. Consider the Cartesian product $\mathcal{A}^2$.

Claim 3.4 *The structure $\mathcal{A}^2$ is isomorphic to the disjoint union of structures of the type $\mathcal{G}(N)$ which are also E-equivalence classes in $\mathcal{A}^2$.*

Proof. Let $\mathcal{G}(B_1)$ and $\mathcal{G}(\mathcal{B}_2)$ be E–components of $\mathcal{A}$. Then one can check that $\mathcal{G}(B_1)\times\mathcal{G}(B_2)$ is isomorphic $\mathcal{G}(N)$ using the same reasoning as in Claim 2.3. Morover, since $\mathcal{G}(B_1)$ and $\mathcal{G}(\mathcal{B}_2)$ are equivalence classes $\mathcal{G}(B_1) \times \mathcal{G}(B_2)$ is an equivalence class in the Cartesian product. Now note that in $\mathcal{A}^2$ there are infinitely many components. Each of the components is an equivalnece class and is isomorphic to $\mathcal{G}(N)$.

Claim 3.5 *The structure $\mathcal{A}^2$ is computably categorical.*

Proof. Take any two computable presentations of $\mathcal{A}^2$. Effectively map in a one–to–one manner all the equivalence classes in one presentation onto the equivalence classes in the other presentation. Then the map can be effectively extended, as can be seen from the previous claim, to an isomorphism between the two presentations. This proves the claim, and hence the theorem. □

4 Conclusion and Acknowledgement

We conclude the paper with the following two hypothesis. The first hypothesis belongs to Goncharov and is stated in [3].

Hypothesis 1 *If a structure $\mathcal{A}$ is computably categorical then the dimension of $\mathcal{A}^2$ is either* 1 *or* ω.

We think that the next hypothesis can be proved by modifying constructions of noncomputably categorical structures of finite dimension from [4] [1].

Hypothesis 2 *For every natural number $n \geq 1$ there exists a computable structure $\mathcal{A}$ such that the dimension of $\mathcal{A}^2$ is n.*

The paper was written while the author was visiting the mathematics and computer science departments at the University of Chicago during the winter of 1999. The author thanks Professor Robert Soare for the invitation and many helpful discussions on computability theory and the theory of computable structures.

References

[1] P. Cholak, S. Goncharov, B. Khoussainov, R. Shore. Computably Categorical Structures and Expansions by Constants, to appear in *Journal of Symbolic Logic*, 1999.

[2] E. Griffor. ed., Handbook of Computability Theory, Studies in Logic and Foundations of Mathematics Series, North-Holland, 1999.

[3] Yu. Ershov and S. S. Goncharov, eds., *Logic Notebook: Problems in Mathematical Logic*, Novosibirsk University, 1986.

[4] B. Khoussainov, R. Shore. Computable isomorphisms, Degree Spectra of Relations, and Scott Families,*Annals of Pure and Applied Logic* 93, 1998, p.153-193.

[5] Yu.Ershov, S. Goncharov, A.Nerode, J.Remmel eds., Marek V. assoc.ed. Handbook of Recursive Mathematics, Volume 1-2. North-Holland, 1999.

[6] D. Hirschfeldt. Personal Communications.

[7] R. Soare. Recursively Enumerable Sets and Degrees. Springer–Verlag, New York, 1987.