Deterministic Operators for BSS RAM's I

(Extended abstract)

Christine Gaßner
University of Greifswald, Germany
gassnerc@uni-greifswald.de

Abstract

We deal with some questions arising from the comparison of different models for computation over algebraic structures. One of these models is the machine-oriented BSS RAM model. Here, we extend the BSS RAM's in order to compare the BSS RAM model with a machine-independent model of abstract computation developed by Moschovakis and models in recursion theory. We introduce a nondeterministic operator similar to an operator considered by Moschovakis and deterministic operators derived from this operator. Two of the deterministic operators used as oracles allow to compute measures of sets semi-decidable or decidable by BSS RAM's and the limits of these sets, respectively. Here, the focus is on structures over which it is possible to simulate BSS RAM's with measure operators by BSS RAM's equipped with a limit operator and by strong Type-2 BSS RAM's, respectively.

1 Introduction

We present some recent advances in the investigation of abstract computation related to operators for BSS RAM's. The here considered models of computation are based on the concept and design of the well-known random access machines. Besides BSS RAM's over several mathematical structures we consider Type-2 RAM's and introduce Type-2 BSS RAM's. These machines are generalizations of the uniform BSS model of computation over the real numbers introduced in [Blum et al. '89], the analytic machines introduced by Günter Hotz, Gero Vierke, and Börn Schieffer in [Hotz et al. '95] in accordance with ideas presented in [Hotz, '94] (cp. [Gärtner '08]), the Type-2 Turing machines studied based on books such as [Weihrauch '00] in the field of computability and complexity in analysis, and infinite Turing machines considered, for instance, by Philipp Schlicht (for the definition see [Carl & Schlicht '16]) and other researchers, respectively. Since a systematic comparison of these models plays an important role in understanding questions about computability and for answering them, we want to study the typical features of the BSS RAM model for some structures, as well as to compare it, for several structures, with other models of computation.

Here we would also like to follow up on the discussion about the strength of operators introduced — in analogy with operators considered by Stephen Cole Kleene and Yiannis Nicholas Moschovakis — for our BSS RAM's, similar new operators, and the relationship between BSS RAM's with these operators and other RAM's.

For this purpose, we give the most important definitions in Section 2. In Section 3 we give the definitions of a nondeterministic and deterministic operators and characterize halting problems for the classes of nondeterministic and deterministic BSS RAM's by the introduced operators. In Section 4, we consider several classes of structures and discuss the possibilities to compute the measures of semi-decidable and decidable sets and the integrals of computable functions by Type-2 BSS RAM's.

2 BSS RAM's and Type-2 RAM's

Let $\mathcal{A} = (U; (c_i)_{i\in I}; f_1, \ldots, f_{n_1}; r_1, \ldots, r_{n_2})$ be a first-order structure containing at least two different constants $c_0, c_1 \in U$ $(0, 1 \in I,\ c_0 \neq c_1)$. Any infinite dimensional *BSS RAM over* $\mathcal{A}$ or $\mathcal{A}$*-machine* $\mathcal{M}$ is equipped with an infinite number of Z-registers $Z_1, Z_2, \ldots$ for the elements of the universe U and a finite number of index registers $I_1, \ldots, I_{k_\mathcal{M}}$ for the addresses of the Z-registers. They can execute each single application of a function f_i and copy single elements by indirect addressing from one Z-register into another Z-register in a fixed time unit and we assume that each relation r_i can also be evaluated, for given values in the Z-registers, in one step. The allowed types of instructions are presented in Figure 1.

We will distinguish between the ordinary BSS RAM and finite and infinite Type-2 RAM's. The ordinary BSS RAM can process inputs $(x_1, \ldots, x_n)$ of any length. As defined in [Gaßner '09], at the beginning, besides assigning the input $(x_1, \ldots, x_n) \in U^\infty =_{\mathrm{df}} \bigcup_{i=1}^{\infty} U^i$ to Z-registers, an index register of such a BSS RAM obtains the length n by an input procedure (cp. Figure 2). The *nondeterministic* machines are additionally able to guess an arbitrary finite number m of arbitrary elements $y_1, \ldots, y_m \in U$ which are provided by an input and guessing procedure (cp. Figure 3) for the further work. The output is done by an output procedure where the length of the output is dependent on the value of a certain index register (cp. Figure 4).

The computed functions are determined by the input-output behavior of the BSS RAM's. Infinite dimensional deterministic BSS RAM's compute partial functions of the type described by $f :\subseteq U^\infty \to U^\infty$. Infinite dimensional nondeterministic BSS RAM's compute partial functions of the type

Computation instructions:
$\ell\colon\ Z_j := f_i(Z_{j_1}, \ldots, Z_{j_{m_i}})$ (e.g. $\ell\colon\ Z_j := Z_{j_1} + Z_{j_2}$)
$\ell\colon\ Z_j := c_i$
Branching instructions:
$\ell\colon$ if $r_i(Z_{j_1}, \ldots, Z_{j_{k_i}})$ then goto ℓ_1 else goto ℓ_2
Copy instructions:
$\ell\colon\ Z_{I_j} := Z_{I_k}$
Index instructions:
$\ell\colon\ I_j := 1$
$\ell\colon\ I_j := I_j + 1$
$\ell\colon$ if $I_j = I_k$ then goto ℓ_1 else goto ℓ_2
Stop instruction:
$\ell\colon$ stop.

Figure 1: BSS RAM's and the allowed instructions

described by $f :\subseteq U^\infty \to \mathcal{P}(U^\infty)$.

Note, that it is also possible to use finite dimensional BSS RAM's for computing — deterministically or nondeterministically — functions $f :\subseteq U^\infty \to U^m$ and $f :\subseteq U^\infty \to \mathcal{P}(U^m)$, respectively. Type-2 RAM's work with additional read-only input and write-only output tapes. This means that there are an additional infinite sequence of input registers and an additional infinite sequence of output registers. While executing the program, stepwise reading inputs from the input tape and writing outputs on the output tape are possible via two additional types of instructions, called read instruction and print instruction, respectively. We distinguish (∞,ω)-, (ω,ω)-, and (ω,∞)-RAM's over $\mathcal{A}$. Finite Type-2 RAM's provide output values in U^∞ after a finite number of steps. We are also interested in computable enumerations over U and over U^∞, the resulting sequences in U^ω given after an infinite number of steps, and their limits. Therefore, we also consider infinite Type-2 BSS RAM's working with a procedure for inputting finite or infinite sequences (where only in the first case the length of the input is assigned to an index register and in any case a second index register gets an information about the input type 1 or 2), an instruction for reading values given on an input tape, an instruction for writing values on an output (or print) tape, an additional output index register I_0 (for determining the length of a possible limit), and a forward-looking procedure for outputting sequences in U^ω (and maybe the value of I_0). For a given

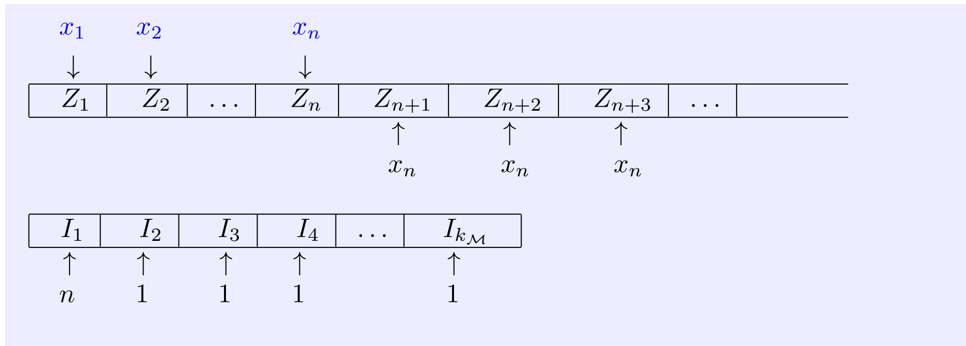

Figure 2: The input of $x_1, \ldots, x_n$) by an input procedure of any deterministic BSS RAM

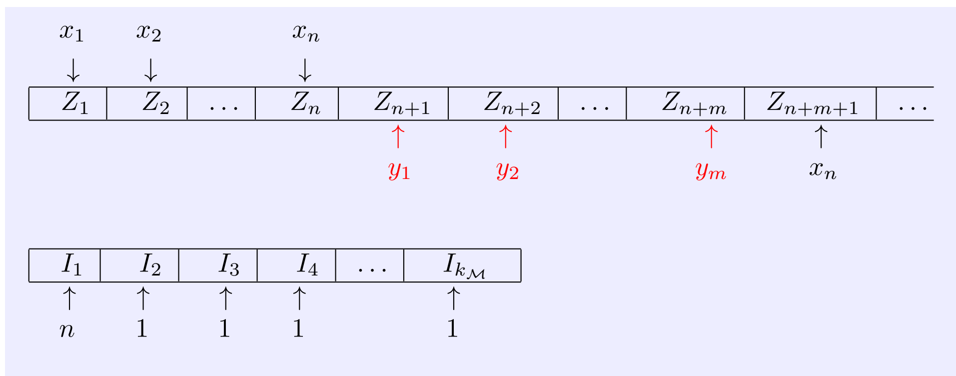

Figure 3: Guessing by an input-guessing procedure of nondeterministic machines

sequence of metrics $(d_m)_{m\in\mathbb{N}^+}$ with $d_m : (U^m)^2 \to \mathbb{R}_+$[1] we consider also infinite-limit Type-2 BSS RAM's providing only the limit in U^∞ for each sequence $(y_{1+kc(I_0)}, \ldots, y_{(k+1)c(I_0)})_{k\in\mathbb{N}^+}$ in $(U^{c(I_0)})^\omega$ if the sequence $y_1, y_2, \ldots$ is written on the print tape after copying the current value $m = c(I_1)$ of the index register I_1 into the register I_0 and the limit exists with respect to $d_{c(I_0)}$. If $\mathcal{A}$ is the ordered field of reals and the considered metrics d_m are the Euclidean metrics, then the infinite-limit Type-2 BSS RAM's with the input space U^∞ can provide limits that can also be computed — by definition — by analytic machines (with the same constants). Analogously to the strong analytic machines considered in [Gärtner & Ziegler '11][2], we will

[1] $\mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geq 0\}$

[2] By [Gärtner & Ziegler '11], a strong analytic machine computes $\vec{y} \in \mathbb{R}^\infty$ if it produces

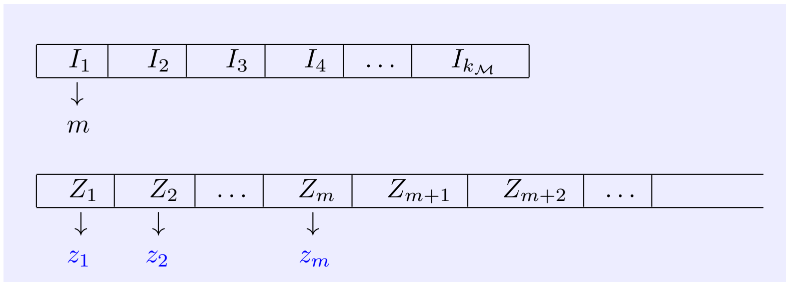

Figure 4: Output procedure of any BSS RAM

also consider the strong infinite-limit Type-2 BSS RAM's and the strong infinite Type-2 BSS RAM's where we get an output only if the sequence $y_1, y_2, \ldots$ — written on the output tape after writing m into I_0 — satisfies $d_m((y_{1+kc(I_0)}, \ldots, y_{(k+1)c(I_0)}), \vec{y}) \leq \frac{1}{2^k}$ for some $\vec{y} \in U^m$ and all $k \in \mathbb{N}^+$ such that the considered sequence $(y_{1+km}, \ldots, y_{(k+1)m})_{k\in\mathbb{N}^+}$ in $(U^m)^\omega$ converges rapidly to the limit $\vec{y} \in U^m$.

Similar to the ν-operator given in [Moschovakis '69] and Kleene's μ-operator, we introduced several *Moschovakis operators*, a ν-operator for nondeterministic ν-oracle machines, further operators for partially ordered structures such as $\nu_{\min}$, $\nu_{\inf}, \ldots$, operators $\nu_{\lim}$ defined by metrics on subsets of U^∞, and measure operators defined by measures on σ-algebras over the universe. We will present details and first results of comparing some oracle BSS RAM's with Type-2 RAM's. In the following, $\mathcal{A}$ is a structure with constants c_0 and c_1 ($c_0 \neq c_1$) such that $\{c_0\}$ and $\{c_1\}$ are decidable and, for any problem P, the *characteristic function* $\chi_P : U^\infty \to \{c_0, c_1\}$ is defined by $\chi_P(\vec{x}) = c_0$ if $\vec{x} \in P$ and $\chi_P(\vec{x}) = c_1$ if $\vec{x} \notin P$.

3 A nondeterministic operator, deterministic operators, and halting problems

ν-oracle BSS RAM's and nondeterministic ν-oracle instructions. For the structure $\mathcal{A}$ and any function $f :\subseteq U^\infty \to U^\infty$ computable by a usual BSS RAM over $\mathcal{A}$, we allow that *ν-oracle BSS RAM's* can — in addition to the instruction types of the usual BSS RAM's — execute a *$\nu[f]$-oracle instruction*. More precisely, for

$\nu[f](x_1, \ldots, x_n)$

---

a sequence $(\vec{y}^{(n)})_{n\in\mathbb{N}}$ with $\dim(\vec{y}^{(n)}) = \dim(\vec{y})$ and $||\vec{y}^{(n)} - \vec{y}|| \leq \frac{1}{2^n}$ for all $n \in \mathbb{N}$.

$$=_{\mathrm{df}} \{y_1 \in U \mid (\exists (y_2, \ldots, y_m) \in U^\infty)(f(x_1, \ldots, x_n, \underbrace{y_1, y_2, \ldots, y_m}_{\vec{y} \in U^\infty}) = c_0)\}$$

we allow the labeled $\nu[f]$-*oracle instruction* — of the form

$$\ell : \; Z_j := \nu[f](Z_1, \ldots, Z_{I_1}).$$

In this way, by $\nu[f] :\subseteq U^\infty \to \mathcal{P}(U)$, the register Z_j obtains the value $y_1 \in \nu[f](z_1, \ldots, z_n)$ if the content of I_1 is n, the contents of $Z_1, \ldots, Z_n$ are $z_1, \ldots, z_n$, and $y_1 \in U$ is some value such that there are an m and $y_2, \ldots, y_m \in U$ satisfying $f(z_1, \ldots, z_n, y_1, \ldots, y_m) = c_0$, and otherwise Z_j does not obtain a value and the machine again goes to the label ℓ (cp. Figure 5, [Gaßner & Valencia '15], and [Gaßner '16/B]).

NONDETERMINISTIC!		$\ell : \; \underset{\substack{\downarrow \\ y_1}}{Z_j} := \nu[f](\overset{\substack{z_1 \\ \downarrow}}{Z_1}, \ldots, \overset{\substack{z_n \\ \downarrow}}{Z_{I_1}})$
$\nu[f](z_1, \ldots, z_n) \neq \emptyset$	$\Rightarrow$	Z_j contains some $y_1 \in \nu[f](z_1, \ldots, z_n)$.
$\nu[f](z_1, \ldots, z_n) = \emptyset$	$\Rightarrow$	No stop (the machine loops forever).

Figure 5: Oracle instruction with ν-operator

This new type of instructions allows to compute *multiple-valued functions* from U^∞ to the power set of U^∞. For computation over $\mathcal{A}$, we say that a problem $P \subseteq U^\infty$ is ν-*semi-decidable* by a ν-oracle BSS RAM $\mathcal{M}$ (using $\nu[f]$ applied on some BSS RAM computable f) when $\mathcal{M}$ computes a multiple-valued function g so that $c_0 \in g(\vec{x})$ iff $\vec{x} \in P$.

Enumerability. A set $P \subseteq U^\infty$ is *enumerable* (*over* $\mathcal{A}$) if there is an $\mathcal{A}$-computable surjective function $f : \{c_1\}^\infty \to P$. Hence, for a structure with $\mathbb{N} \subseteq U$, the constant $c_1 = 0$, the computable successor function s with $s(x) = x + 1$ for $x \in \mathbb{N}$, and a relation $=$ decidable for non-negative integers in $\mathbb{N}$, $P \subseteq U^\infty$ is enumerable over $\mathcal{A}$ if and only if there is an $\mathcal{A}$-computable surjective function $f : \mathbb{N} \to P$ since we can compute $n \mapsto (0, \ldots, 0) \in \mathbb{N}^{n+1}$ for all $n \in \mathbb{N}$ and $(0, \ldots, 0) \in \mathbb{N}^n \mapsto (n-1)$ for all $n \in \mathbb{N}$. Note that, for computations over such a structure, there is an (∞, ω)-machine computing $f :\subseteq U^\infty \to U^\omega$ if and only if a function $g :\subseteq U^\infty \to U$ satisfying[3] $f(\vec{x}) = (g(n \,.\, \vec{x}))_{n \in \mathbb{N}}$ can be computed by a BSS RAM.

[3] $(\vec{x} \,.\, \vec{y}) = (x_1, \ldots, x_n, y_1, \ldots, y_m)$ if $\vec{x} = (x_1, \ldots, x_n)$ and $\vec{y} = (y_1, \ldots, y_m)$.

BSS RAM's and deterministic oracle instructions. For any structure $\mathcal{A}$ with a partial order r_1 on U, let us consider oracle machines such as $\nu_{\min}$-*oracle BSS RAM's* defined in [Gaßner '15/B]. By the labeled $\nu_{\min}[f]$-*oracle instruction* ℓ : $Z_j := \nu_{\min}[f](Z_1, \ldots, Z_{I_1})$ the register Z_j obtains the value y_1 if $Z_j := \nu[f](Z_1, \ldots, Z_{I_1})$ provides at least one value for Z_j, the set of these possible values has a minimum, and y_1 is this minimum. In all other settings, Z_j does not obtain a value and the machine again goes to the label ℓ.

In a similar way, we can define operators $\nu_{\inf}, \nu_{\max}, \ldots$ for $\nu_{\inf}$-, $\nu_{\max}$-, and further oracles. Note, that Pedro F. Valencia Vizcano gave talks in Greifswald where he presented, for the computation over the field of reals, the ideas for the simulation of the $\nu_{\max}[f]$-oracle instruction by a $\nu_{\sup}$-oracle BSS RAM. If $\nu[f](\vec{x})$ provides a set of values for structures with a metric, then an operator $\nu_{\lim}$ provides the limit (the only accumulation point) of this set if it is applied on f and $\vec{x}$ and the limit exists. Note, that the corresponding operators for providing minimal elements or accumulation points are suitable to extend BSS RAM's by nondeterministic oracle instructions.

In order to define several measure operators as in [Gaßner '16/A], let $\vec{\nu}[f](\vec{x}) = \{\vec{y} \in U^\infty \mid f(\vec{x} \,.\, \vec{y}) = c_0\}$ and let $\vec{\nu}^{(k)}[f]$ be a projection of $\vec{\nu}$ on U^k. In more details, let $\vec{\nu}^{(k)}[f](\vec{x}) =_{\mathrm{df}} \{\vec{w} \in U^k \mid (\exists \vec{y} \in U^\infty)(f(\vec{x} \,.\, \vec{w} \,.\, \vec{y}) = c_0)\}$ for $k \geq 1$. By the way, this means that $\vec{\nu}^{(1)}[f](\vec{x}) = \nu[f](\vec{x})$. Then, for any structure $\mathcal{A}$ with $\mathbb{N} \subseteq U$, we can consider the σ-algebra $\mathfrak{A} = \mathcal{P}(U^\infty)$ and, thus, the $\vec{\nu}_{\mathrm{numb}}[f]$-*oracle instruction* $Z_j := \vec{\nu}_{\mathrm{numb}}[f](Z_1, \ldots, Z_{I_1})$ where the measure operator $\vec{\nu}_{\mathrm{numb}} : \subseteq U^\infty \to \mathbb{N}$ is defined by means of the counting measure such that $\vec{\nu}_{\mathrm{numb}}[f](\vec{x}) =_{\mathrm{df}} |A|$ if $A = \vec{\nu}[f](\vec{x})$ and $|A| < \infty$ and $\vec{\nu}_{\mathrm{numb}}[f](\vec{x})$ is not defined otherwise. The corresponding $\nu_{\mathrm{numb}}[f]$-*oracle instruction* refers to $\nu[f](\vec{x})$.

For structures of the reals, we will consider the deterministic oracle instruction $Z_j := \vec{\nu}^{(k)}_{\mathrm{Lebes}}[f](Z_1, \ldots, Z_{I_1})$ with the *Lebesgue measure operator* $\vec{\nu}^{(k)}_{\mathrm{Lebes}}$ defined by $\vec{\nu}^{(k)}_{\mathrm{Lebes}}[f](\vec{x}) =_{\mathrm{df}} \lambda^k(\vec{\nu}^{(k)}[f](\vec{x}))$ and the Lebesgue measure λ^k for Borel or Lebesgue sets given in the form of sets $\vec{\nu}^{(k)}[f](\vec{x}) \subseteq \mathbb{R}^k$ (where $\vec{\nu}^{(k)}_{\mathrm{Lebes}}[f](\vec{x})$ is not defined if $\vec{\nu}^{(k)}[f](\vec{x})$ is not in the considered σ-algebra), and the corresponding instruction determined by the *uniform Lebesgue measure operator* $\vec{\nu}_{\mathrm{Lebes}}$ given by $\vec{\nu}_{\mathrm{Lebes}}[f](\vec{x}) =_{\mathrm{df}} \sum_k \lambda^k(\vec{\nu}^{(k)}[f](\vec{x}))$.

For any σ-algebra $\mathfrak{A} \subseteq \mathcal{P}(U^n)$ and any measure κ on $\mathfrak{A}$, we denote the corresponding measure operator also by $\vec{\nu}^{\mathfrak{A}}_\kappa$ and $\vec{\nu}_\kappa$, respectively.

Halting problems and operators. For the class $\mathsf{M}_{\mathcal{A}}$ of all deterministic BSS RAM's over $\mathcal{A}$ and for the class $\mathsf{M}^{\mathrm{N}}_{\mathcal{A}}$ of all nondeterministic BSS RAM's over $\mathcal{A}$ we will consider the halting problems $\mathbb{H}_{\mathcal{A}}$ and $\mathbb{H}_{\mathcal{A}}^{\mathrm{N}}$ given by

$$\mathbb{H}_{\mathcal{A}}^{[\mathrm{N}]} = \{(\vec{x}\,.\,code(\mathcal{M})) \mid \vec{x} \in U^{\infty} \ \&\ \mathcal{M} \in \mathsf{M}_{\mathcal{A}}^{[\mathrm{N}]} \ \&\ \mathcal{M}(\vec{x}) \downarrow\}$$

where, in the nondeterministic setting, $\mathcal{M}(\vec{x}) \downarrow^{[\leq k]}$ means that $\mathcal{M}$ halts on $\vec{x}$ for some guesses $y_1, \ldots, y_m$ [after at most k steps].[4] Analogously to the relationship between deterministic and nondeterministic Turing machines, there holds $\mathbb{H}_{\mathcal{A}}^{\mathrm{N}} \equiv_{\mathcal{A}} \mathbb{H}_{\mathcal{A}}^{\mathrm{PROJ}}$ for

$$\begin{aligned} &\mathbb{H}_{\mathcal{A}}^{\mathrm{PROJ}} \\ &= \bigcup_{n=1}^{\infty} \{(\vec{x}\,.\,code(\mathcal{M})) \mid \vec{x} \in U^{n} \ \&\ \mathcal{M} \in \mathsf{M}_{\mathcal{A}} \ \&\ (\exists \vec{y} \in U^{\infty})(\mathcal{M}(n\,.\,\vec{x}\,.\,\vec{y}) \downarrow)\}, \end{aligned}$$

and we have $\mathbb{H}_{\mathcal{A}}^{\mathrm{N}} \equiv_{\mathcal{A}} \mathbb{H}_{\mathcal{A}}^{\mathrm{EXI}}$ for

$$\mathbb{H}_{\mathcal{A}}^{\mathrm{EXI}} = \{(\vec{x}\,.\,code(\mathcal{M})) \mid \vec{x} \in U^{\infty} \ \&\ \mathcal{M} \in \mathsf{M}_{\mathcal{A}} \ \&\ (\exists \vec{y} \in U^{\infty})(\mathcal{M}(\vec{x}\,.\,\vec{y}) \downarrow)\}.$$

A consequence is that a problem is ν-semi-decidable by a ν-oracle BSS RAM over $\mathcal{A}$ if and only if there is a nondeterministic BSS RAM over $\mathcal{A}$ that recognizes this problem (cp. Figure 6).

$$\begin{array}{llll} & \quad x_1 \ \cdots \ x_n & & \quad x_1 \ \cdots \ x_n \quad y_1 \\ & \quad \downarrow \qquad \downarrow & & \quad \downarrow \qquad \downarrow \qquad \downarrow \\ Z_j := \nu[f](Z_1, \ldots, Z_{I_1}); \ \ldots & & Z_j := \nu[f](Z_1, \ldots, Z_{I_1-1}, Z_{I_1}) & \ldots \\ \downarrow & & \downarrow & \\ y_1 & & y_2 & \\ \end{array}$$

$$\Rightarrow f(x_1, \ldots, x_n, y_1, \ldots, y_m) = c_0$$

Figure 6: Guessing by the ν-operator of a ν-oracle machine

Theorem 1 *For the computation over $\mathcal{A}$, the halting problem $\mathbb{H}_{\mathcal{A}}$ and consequently all semi-decidable sets are decidable by a BSS RAM using one of the following operators*

- $\nu_{\min}$ *provided that there is a partial order $<$ on U with $c_0 < c_1$,*

[4]Let $code(\mathcal{M}) \in \{c_1\} \times \{c_1\} \times U \times \{c_0\} \times U \times \{c_0\} \times \cdots$ be the code of the program (which is a string) where any symbol of a constant is encoded by itself and the other single symbols of the program are encoded by k elements in $\{c_0, c_1\}$.

- ν_{inf} *provided that there is a partial order* $<$ *on* U *with* $c_0 < c_1$,
- ν_{lim} *and an enumerable infinite set* $\{c_i^{(0)} \mid i \in \mathbb{N}\}$ *with* $\lim_{i\to\infty} c_i^{(0)} = c_0$ *with respect to a metric on* U.
- ν_{numb} *provided that* $\mathbb{N} \subseteq U$ *and* $x = 2$ *is decidable.*
- $\vec{\nu}_{\mathrm{Lebes}}$ *provided that* $\lambda^{[1]}(V) > 0$ *for some* $V \subseteq \mathbb{R}_+ \subseteq U$ *and* $c_1 = 0$.

If, for all nondeterministic $\mathcal{A}$*-machines* $\mathcal{M}$, $\mathcal{M}(\vec{x}) \downarrow^{\leq k}$ *is decidable over* $\mathcal{A}$, *then the same holds for* $\mathbb{H}_{\mathcal{A}}^{\mathbb{N}}$.

4 The computation of measures and integrals by Type-2 BSS RAM's

The simulation of the $\vec{\nu}_{\mathrm{Lebes}}$-operator by the ν_{lim}-operator. If we want to compute, for the set $A = \{\vec{y} \in \mathbb{R}^\infty \mid f(\vec{x}\,.\,\vec{y}) = 1\}$ defined by a computable totally defined function f, the sum of the Lebesgue measures for all $A \cap U^n$ $(n \geq 1)$ by a ν_{lim}-operator over the ordered ring of the reals, then we can use that f is computable by a machine where any test $Z_i \geq Z_j$ is replaced by a test $Z_i - Z_j > 0$, followed by the tests $Z_i - Z_j = 0$ (considered in case that $Z_i - Z_j > 0$ is not satisfied) and $Z_i - Z_j < 0$ (considered in case that $Z_i - Z_j > 0$ and $Z_i - Z_j = 0$ are not satisfied). This means that, in this setting where all test functions are continuous, any new computation path can be traversed only by inputs in an open set or only by zeros of polynomials. Thus, it is sufficient to compute limits of values given by the Lebesgue measure of open sets since the Lebesgue measure of the zero sets vanishes for all non-constant polynomial functions. Consequently, this observation allows to compute the Lebesgue measure sum $\vec{\nu}_{\mathrm{Lebes}}[f](\vec{x})$ by ν_{lim}. We want to make a first step to generalize this statement possible for the ordered ring to structures with continuous basic functions. We know that the set of closed cubes $[(a_1, \ldots, a_n), (b_1, \ldots, b_n)] =_{\mathrm{df}} [a_1, b_1] \times \cdots \times [a_n, b_n]$ with $a_i < b_i$ for all $i \leq n$ is sufficient to generate the σ-algebra $\mathscr{B}(\mathbb{R}^n)$ of Borel sets since any open set is the (σ-)union of closed cubes with disjoint interiors defined as follows. For all $l \in \mathbb{N}$, let

$$\mathcal{W}_l = \{[\frac{i_1}{2^l}, \frac{i_1+1}{2^l}] \times \cdots \times [\frac{i_n}{2^l}, \frac{i_n+1}{2^l}] \mid i_1, \ldots, i_n \in \mathbb{Z}\}.$$

Then, any open set $S \subseteq \mathbb{R}^n$ can be described by

$$S = \bigcup_{l=0}^{\infty} \bigcup_{W \in \mathcal{W}_l^S} W = \lim_{m \to \infty} \bigcup_{l=0}^{m} \bigcup_{W \in \mathcal{W}_l^S} W$$

for

- $\mathcal{W}_0^S = \{W \in \mathcal{W}_0 \mid W \subseteq S\}$,
- $\mathcal{W}_l^S = \{W \in \mathcal{W}_l \mid W \subseteq S \;\&\; W^\circ \not\subset \bigcup_{j=0}^{l-1} \bigcup_{W_j \in \mathcal{W}_j^S} W_j\}$ $(l \geq 1)$[5]

(cp. [Hackenbroch '87], p. 38). (For the background see also [Elstrodt '02].) In the following part we will use the latter property. Therefore, let the class $\mathsf{struc}_{(\mathrm{I})}$ be the class of structures $(\mathbb{R}; (c_i)_{i \in I}; f_1, \ldots, f_{n_1}; \geq)$ satisfying the following requirements.

1. For any $i \leq n_1$, the basic function $f_i : \mathbb{R}^{m_i} \to \mathbb{R}$ is totally defined and continuous.

2. $f_1(x, y) = x + y$ and $f_2(x, y) = x - y$ for all reals x and y.

3. $c_0 = 1$ and $c_1 = 0$.

4. The values $c = \frac{1}{b}$ and $c^2, c^3, \ldots$ are, for some $b \in \{2, 3, \ldots\}$, effectively enumerable by a BSS RAM over $\mathcal{A}$.

5. The zero set $S_h \subseteq \mathbb{R}^n$ of any function $h : \mathbb{R}^n \to \mathbb{R}$ resulting from composing some operations $f_i : \mathbb{R}^{m_i} \to \mathbb{R}$ and constant functions $f_{(c_i)} : \mathbb{R}^n \to \{c_i\}$ has the Lebesgue measure 0 if the set is not the whole domain $\mathbb{R}^n$.

6. It is decidable whether the composition of functions $f_i : \mathbb{R}^{m_i} \to \mathbb{R}$ and $f_{(c_i)} : \mathbb{R}^n \to \{c_i\}$ provides the trivial constant function $f_{(0)} : \mathbb{R}^n \to \{0\}$. For any code code($\mathcal{M}$) of a machine $\mathcal{M}$ with a straight-line program (a program without branching instructions) we want to get the answer for the function computed by $\mathcal{M}$.

7. There is a BSS RAM $\mathcal{N}$ over $\mathcal{A}$ that decides for any function $h : \mathbb{R}^n \to \mathbb{R}$ resulting from the composition of some $f_i : \mathbb{R}^{m_i} \to \mathbb{R}$ and $f_{(c_i)} : \mathbb{R}^n \to \{c_i\}$ and for any open cube $](\frac{i_1}{b^l}, \ldots, \frac{i_n}{b^l}), (\frac{i_1+1}{b^l}, \ldots, \frac{i_n+1}{b^l})[$ whether the intersection of the zero set S_h and the cube is the empty set.

[5] $W^\circ = \bigcup\{G \subset W \mid G \text{ open}\}$

More precisely, we assume that the set $\{((i_1,\ldots,i_n,l)\,.\,code(\mathcal{M}_h)) \mid (\forall\vec{x}\in\mathbb{R}^n)(h(\vec{x})=\mathcal{M}_h(\vec{x}))\ \&\ ](\frac{i_1}{b^l},\ldots,\frac{i_n}{b^l}),(\frac{i_1+1}{b^l},\ldots,\frac{i_n+1}{b^l})[\cap S_h=\emptyset\}$ is decidable by $\mathcal{N}$ over $\mathcal{A}$. [6]

Theorem 2 *If $\mathcal{A}$ is in $\mathsf{struc}_{(\mathrm{I})}$ and $f:\mathbb{R}^\infty\to\mathbb{R}^\infty$ is any totally defined function computable over $\mathcal{A}$, then*
$\vec{\nu}_{\mathrm{Lebes}}[f](\vec{x})$ is computable
— by some BSS RAM with ν_{lim}-oracle and
— by an infinite-limit Type-2 BSS RAM
over $\mathcal{A}$.

A consequence is that the Lebesgue measure sum of every decidable set is computable by an infinite-limit Type-2 BSS RAM. If we define $g(1\,.\,\vec{x})=1$ if $f(\vec{x})=1$ for all $\vec{x}\in\mathbb{R}^\infty$ and $g(1\,.\,\vec{x})$ is not defined otherwise, then $\vec{\nu}_{\mathrm{Lebes}}[g](1)$ provides the Lebesgue measure sum of a semi-decidable set.

Theorem 3 *If $\mathcal{A}$ is in $\mathsf{struc}_{(\mathrm{I})}$ and $f:\subseteq R^\infty\to U^\infty$ is any function computable over $\mathcal{A}$, then*
$\vec{\nu}_{\mathrm{Lebes}}[f](\vec{x})$ is computable
— by some BSS RAM with ν_{lim}-oracle and
— by an infinite-limit Type-2 BSS RAM
over $\mathcal{A}$.

This means also that the Lebesgue measure sum of a semi-decidable set is computable by an infinite-limit Type-2 BSS RAM.

For functions $f:[0,1]^n\to U^\infty$, Theorem 2 can be improved as follows.

Theorem 4 *If $\mathcal{A}$ is in $\mathsf{struc}_{(\mathrm{I})}$ and $f:[0,1]^n\to U^\infty$ is any totally defined function computable over $\mathcal{A}$, then $\vec{\nu}_{\mathrm{Lebes}}[f](\vec{x})$ is computable by a strong infinite-limit Type-2 BSS RAM over $\mathcal{A}$.*

A consequence is that the Lebesgue measure sum of every decidable set $A\subseteq[0,1]^n$ is computable by a strong infinite-limit Type-2 BSS RAM if $\mathcal{A}$ is in $\mathsf{struc}_{(\mathrm{I})}$.

κ-Integration and the operators ν_κ and ν_{lim}. In the following, let $\mathsf{struc}_{(\mathrm{II})}$ be the class of structures $\mathcal{A}=(U;(c_i)_{i\in I};f_1,\ldots,f_{n_1};r_1,\ldots,r_{n_2})$ with the following properties.

[6]For $h:\mathbb{R}^n\to\mathbb{R}$, $h(\vec{x})=\mathcal{M}(\vec{x})$ means that $h(\vec{x})$ results from the input-output behavior of $\mathcal{M}$. If $\mathcal{M}$ computes f, we have $h(\vec{x})=f(\vec{x})$ for the considered $\vec{x}\in\mathbb{R}^n$.

1. $\mathbb{R}_+ \subseteq \{c_i \mid i \in I\}$.
2. $c = \frac{1}{2}$ and $c^2, c^3, \ldots$ are computable over $\mathcal{A}$.
3. The function $a : (\mathbb{R}_+)^2 \to \mathbb{R}_+$ with $a(x,y) = x + y$ for all $x, y \in \mathbb{R}_+$ is computable over $\mathcal{A}$.
4. The function $m : \{c, c^2, c^3, \ldots\} \times \mathbb{R}_+ \to \mathbb{R}_+$ with $m(x,y) = (x \cdot y)$ for $x \in \{c, c^2, c^3, \ldots\}$ and $y \in \mathbb{R}_+$ is computable over $\mathcal{A}$.
5. The relation $\leq$ restricted to $(\mathbb{R}_+)^2$ is decidable.

For any $\mathcal{A} \in \mathsf{struc}_{(\mathrm{II})}$ and any σ-algebra $\mathfrak{A} \subseteq \mathcal{P}(U^p)$, we consider functions $h : U^p \to \{a_0^h, \ldots, a_{i_h}^h\}$ with $\{a_0^h, \ldots, a_{i_h}^h\} \subseteq \mathbb{R}_+$. Any such function h is given by[7] $h(\vec{x}) = \sum_{i=0}^{i_h} a_i \mathbf{1}_{h^{-1}(\{a_i\})}(\vec{x})$. If h is computable over $\mathcal{A}$, then the sets $h^{-1}(\{a_i\}) \subseteq U^\infty$ are decidable over $\mathcal{A}$. h is called $\mathfrak{A}$-$\mathcal{A}$-*simple* if it is computable over $\mathcal{A}$ and $\mathfrak{A}$-$\mathscr{B}(\mathbb{R})$ measurable. Thus, we get the following for finite integrals in $\mathbb{R}_+$.

Theorem 5 *For any $\mathcal{A}$ in $\mathsf{struc}_{(\mathrm{II})}$, any measure κ over a σ-algebra $\mathfrak{A} \subseteq \mathcal{P}(U^p)$, and any $\mathfrak{A}$-$\mathcal{A}$-simple function h, the κ-integral of h over U^p is computable by a BSS RAM using the $\nu_\kappa^{\mathfrak{A}}$-operator if the κ-integral of h over U^p is in $\mathbb{R}_+$.*

Corollary 1 *For any $\mathcal{A}$ in $\mathsf{struc}_{(\mathrm{I})} \cap \mathsf{struc}_{(\mathrm{II})}$, the λ-integral of any $\mathscr{B}(\mathbb{R}^p)$-$\mathcal{A}$-simple function $h : \mathbb{R}^p \to \mathbb{R}_+$ is computable over $\mathcal{A}$*
— by a BSS RAM using the $\vec{\nu}_{\mathrm{Lebes}}^{[p]}$-operator,
— by some BSS RAM with ν_{lim}-oracle, and
— by a strong infinite-limit Type-2 BSS RAM
if the λ-integral of h over $\mathbb{R}^p$ is in $\mathbb{R}_+$.

Theorem 6 *For any $\mathcal{A}$ in $\mathsf{struc}_{(\mathrm{II})}$, any measure κ over a σ-algebra $\mathfrak{A} \subseteq \mathcal{P}(U^p)$, and any non-negative $\mathfrak{A}$-$\mathscr{B}(\mathbb{R})$ measurable function $f : U^p \to \mathbb{R}_+$ totally defined and computable by a BSS RAM over $\mathcal{A}$, the κ-integral of f over U^p is computable by an infinite-limit Type-2 BSS RAM using the ν_κ-operator if the integral is in $\mathbb{R}^+$.*

We want to use that, for any $\mathfrak{A}$-$\mathscr{B}(\mathbb{R})$ measurable $f : U^p \to \mathbb{R}$, there are two non-negative $\mathfrak{A}$-$\mathscr{B}(\mathbb{R})$ measurable functions $f^+, f^- : U^p \to \mathbb{R}_+$

[7] The so-called indicator function $\mathbf{1}_S$ is the characteristic function of S.

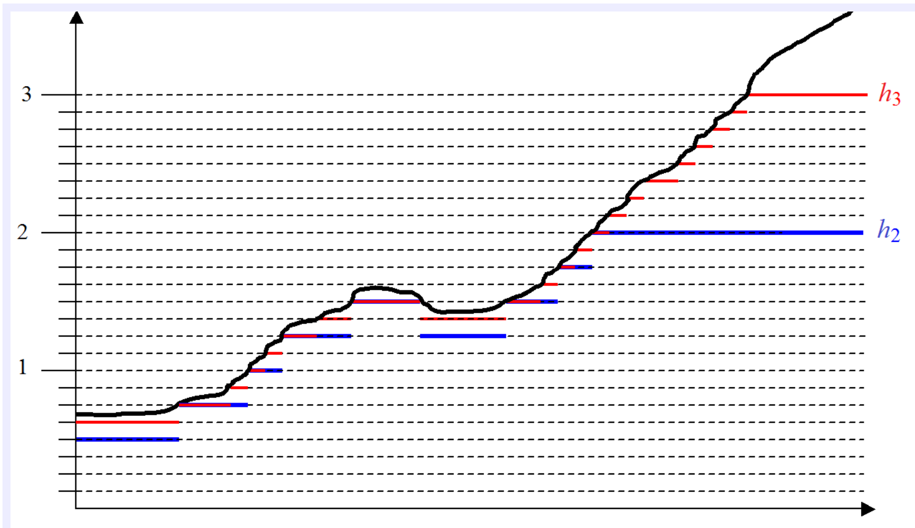

Figure 7: The simple functions h_n

given by $f^+(\vec{x}) =_{\mathrm{df}} \max\{f(\vec{x}), 0\}$ and $f^-(\vec{x}) =_{\mathrm{df}} \max\{-f(\vec{x}), 0\}$ such that we have $f = f^+ - f^-$ and consequently

$$\int_{U^p} f d\kappa = \int_{U^p} f^+ d\kappa - \int_{U^p} f^- d\kappa$$

if one of the integrals has a value in $\mathbb{R}$. Let $\mathsf{struc}_{(\mathrm{III})}$ be the class of structures in $\mathsf{struc}_{(\mathrm{II})}$ with $\mathbb{R} \subseteq U$ such that the function $s : \mathbb{R}^2 \to \mathbb{R}$ with $s(x, y) = x - y$ for all $x, y \in \mathbb{R}$ is computable and the relation $\leq$ restricted to $\mathbb{R}^2$ is decidable.

Theorem 7 *For any $\mathcal{A}$ in $\mathsf{struc}_{(\mathrm{III})}$, any measure κ over a σ-algebra $\mathfrak{A} \subseteq \mathcal{P}(\mathbb{R}^p)$, and any $\mathfrak{A}$-$\mathscr{B}(\mathbb{R})$ measurable function $f : U^p \to \mathbb{R}$ totally defined and computable by a BSS RAM over $\mathcal{A}$, the κ-integral of f over U^p is computable by an BSS RAM using the $\nu_{\lim}$-operator applied on functions computable by means of the ν_κ-operator if the κ-integral of f^+ over U^p and the κ-integral of f^- over U^p are in $\mathbb{R}_+$.*

Theorem 8 *For any $\mathcal{A}$ in $\mathsf{struc}_{(\mathrm{III})}$, any measure κ over a σ-algebra $\mathfrak{A} \subseteq \mathcal{P}(\mathbb{R}^p)$, and any $\mathfrak{A}$-$\mathscr{B}(\mathbb{R})$ measurable function $f : [0, 1]^p \to [0, 1]$ totally defined and computable by a BSS RAM over $\mathcal{A}$, the κ-integral of f over $[0, 1]^p$ is computable by a strong infinite-limit BSS RAM using the ν_κ-operator.*

Theorem 9 *For any $\mathcal{A}$ in $\mathsf{struc}_{(\mathrm{I})} \cap \mathsf{struc}_{(\mathrm{II})}$ and any function $f : \mathbb{R}^p \to \mathbb{R}_+$ totally defined and computable by a BSS RAM over $\mathcal{A}$, the $\lambda^{[p]}$-integral of f over $\mathbb{R}^p$ is computable by an infinite-limit Type-2 BSS RAM.*

Theorem 10 *For any $\mathcal{A}$ in $\mathsf{struc}_{(\mathrm{I})} \cap \mathsf{struc}_{(\mathrm{II})}$ and any function $f : [0,1]^p \to [0,1]$ totally defined and computable by a BSS RAM over $\mathcal{A}$, the κ-integral of f over $[0,1]^p$ is computable by a strong infinite-limit BSS RAM.*

5 Conclusion and Acknowledgment

Since $\mathbb{R}^{\leq} = (\mathbb{R}; \mathbb{R}; +, -, \cdot; \leq) \in \mathsf{struc}_{(\mathrm{I})} \cap \mathsf{struc}_{(\mathrm{II})}$ holds, the presented theorems can be applied to the BSS model and, thus, they help to better understand the similarities and differences between the relationships of different oracle BSS RAM's and special Weihrauch reductions considered by Arno Pauly for Type-2 Turing machines (cp. [Neumann & Pauly '16]), investigations on the computation of integrals by Florian Steinberg (cp. [Steinberg '17]), and their consequences.

I would like to thank Volkmar Liebscher for helpful discussions about the "bad" properties of certain continuous functions. Moreover, I thank Arno Pauly for critical reading the definitions and his proposal to introduce also the strong Type-2 BSS RAM's in this paper.

References

[Balcázar et al. '88/90] Balcázar, J., J. Díaz und J. Gabarró: "Structural complexity I"; Springer-Verlag (1988/90).

[Blum et al. '98] Blum, L., F. Cucker, M. Shub und S. Smale: "Complexity and real computation"; Springer-Verlag (1998).

[Blum et al. '89] Blum, L., M. Shub, and S. Smale: "On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines"; Bulletin of the Amer. Math. Soc. 21 (1989), 1–46.

[Carl & Schlicht '16] Carl, M., and P. Schlicht: "Random reals and infinite time Turing machines" (2017), https://math-inf.uni-greifswald.de/fileadmin/uni-greifswald/ fakultaet/mnf/mathinf/gassner/arbeitstreffen_2016/texte/Slides-schlicht.pdf.

[Elstrodt '02] Elstrodt, J.: "Maß- und Integrationstheorie"; Springer, Berlin (2002).

[Gärtner '08] Gärtner, G.: "Analytische Maschinen und Berechenbarkeit analytischer Funktionen"; Dissertation, Saarbrücken (2008).

1–20.

[Gärtner & Ziegler '11] Gärtner, G., and M. Ziegler: "Real Analytic Machines and Degrees"; LMCS 7 (3:11) (2011)

[Gaßner '09] Gaßner, C.: "Oracles and relativizations of the P =? NP question for several structures"; JUCS vol. 15 no. 6 (2009), 1186–1205.

[Gaßner '15/A] Gaßner, C.: "Computation over algebraic structures and a classification of undecidable problems"; MSCS (2016), Available on CJO 2016 doi:10.1017/S0960129516000116.

[Gaßner & Valencia '15] Gaßner, C., and P. F. Valencia Vizcano: "Operators for BSS RAM's" in Martin Ziegler, Akitoshi Kawamura (eds.): Abstract Booklet Computability & Complexity in Analysis (CCA 2015), 24–26.

[Gaßner '15/B] Gaßner, C.: "Moschovakis Operators for BSS RAMs: Definitions and Motivation";

https://math-inf.uni-greifswald.de/fileadmin/uni-greifswald/fakultaet/mnf/mathinf/gassner/ShortenedVersionCCA.pdf

[Gaßner '16/A] Gaßner, C.: "BSS RAMs with Operators for Several Measure"; Arbeitstreffen "Computability and the BSS model" (2016).

https://math-inf.uni-greifswald.de/fileadmin/uni-greifswald/fakultaet/mnf/mathinf/gassner/arbeitstreffen_2016/texte/Slides-Gassner.pdf

[Gaßner '16/B] Gaßner, C.: "BSS RAMs with ν-Oracle and the Axiom of Choice"; Colloquium Logicum (2016).

http://www.math.uni-hamburg.de/spag/ml/CL2016/Slides/gassner.pdf

[Goode '94] Goode, J. B.: "Accessible telephone directories"; J. Symbolic Logic Volume 59 (1) (1994), 92–105.

[Hackenbroch '87] Hackenbroch, Wolfgang: "Integrationstheorie"; Teubner (1987).

[Hemmerling '98] Hemmerling, A.: "Computability of string functions over algebraic structures"; Mathematical Logic Quarterly 44 (1998), 1–44.

[Hotz, '94] Hotz, G.: "Über Berechenbarkeit fraktaler Strukturen", Abhandlungen der mathematisch-naturwissenschaftlichen Klasse der Akademie der Wissenschaften und der Literatur, 1994.

[Hotz et al. '95] Hotz, G., G. Vierke, and B. Schieffer: "Analytic machines"; ECCC 2:25 (1995).

[Kozen '06] Kozen, D. C.: "Theory of computation"; Springer (2006).

[Moschovakis '69] Moschovakis, Y. N.: "Abstract first order computability. I"; Transactions of the American Mathematical Society 138 (1969), 427–464. http://dx.doi.org/10.2307/1994926

[Neumann & Pauly '16] Eike Neumann, E., and A. Pauly: "A topological view on algebraic computations models", arXiv:1602.08004 (2016).

[Poizat '95] Poizat, B.: "Les Petits Cailloux"; Aléas (1995).

[Steinberg '17] Steinberg, F.: "Complexity theory for spaces of integrable functions. (2017), Available at https://arxiv.org/abs/1612.06419. To appear in Logical Methods in Computer Science.

[Weihrauch '00] Weihrauch, K.: "Computable analysis"; Springer (2000).