Randomized First order Computability

Dimiter Skordev
Sofia Univ., Fac. of Math. and Informatics
Sofia, Bulgaria
skordev@fmi.uni-sofia.bg
http://www.fmi.uni-sofia.bg/fmi/logic/skordev/

July 25, 2005

Dedicated to Y. N. Moschovakis

A *randomized computation* is a computation that makes use of some random number generators. Mathematically, a random number generator can be regarded as an infinite sequence of real numbers $p_0, p_1, p_2, \ldots$ such that $0 \leq p_k < 1$ for all k in $\mathbb{N} = \{0, 1, 2, \ldots\}$, and

$$\sum_{k=0}^{\infty} p_k = 1$$

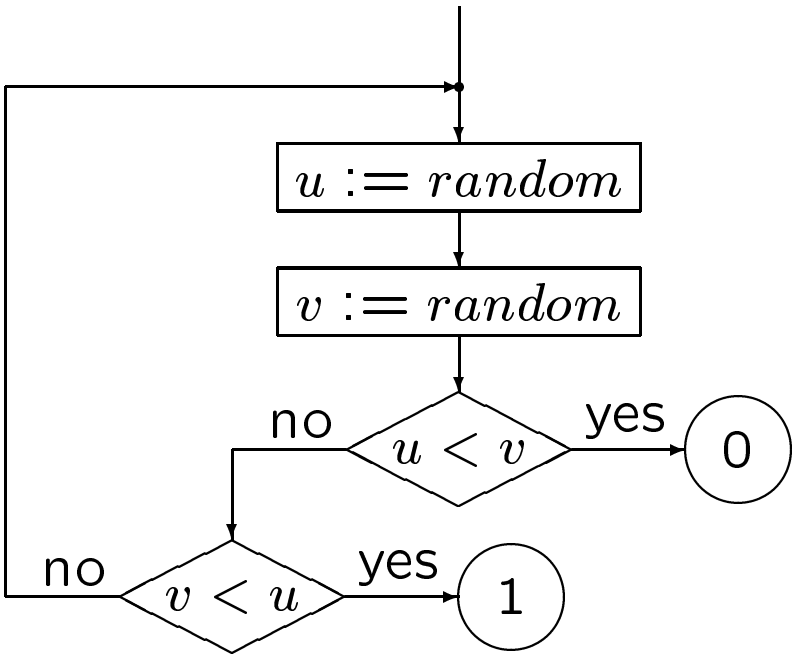
(p_k being the probability of generating the number k). Examples:

$$\frac{1}{2}, \frac{1}{2}, 0, 0, 0, \ldots$$
$$0, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, 0, 0, 0, \ldots$$
$$\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \ldots$$

Using two different random number generators $p_0, p_1, p_2, \ldots$ and $q_0, q_1, q_2, \ldots$ can be replaced by using a generator $r_0, r_1, r_2, \ldots$ defined as follows: we take a computable bijection J of $\mathbb{N}^2$ onto $\mathbb{N}$ and set $r_{J(m,n)} = p_m q_n$ for all $m, n \in \mathbb{N}$.

A *probabilistic program* is a program for randomized computations. Without loss of generality, we may assume that only one random number generator is used by the program. The course of the computations according such a program usually depends not only on the input data (if any), but also on the numbers generated by the random number generator at certain stages of the computation. The output of a probabilistic program has, in general, some probabilistic distribution, and in the case of input data the distribution may depend on them.

Example (simulation à la von Neumann of an unbiased coin). Let an arbitrary random number generator be given, and let $random$ be the result produced by it. Then the output of the program on the next slide is 0 or 1, and each of these two numbers has the probability $\frac{1}{2}$ to occur.



In the case of a probabilistic program with input, the dependence of the output on the input can be represented by a random function. If D is a set (*the data set*) then a *random function* in D is a non-negative real-valued function θ such that $\mathrm{dom}(\theta) = D^2$ and

$$\sum_{y=0}^{\infty} \theta(x, y) \ \leq \ 1$$

for any x in D (the number $\theta(x, y)$ is the probability of returning y as output when x is given as input).

The random functions that correspond to probabilistic programs will be called *computable*. Of course their class may depend on the considered class of programs and on the random number generator used at their execution.

A way via probabilistic Turing machines is often used for the investigation of the class of the computable random functions. This can be done if D is $\mathbb{N}$ or some other set of constructive objects, and D is considered with the usual algorithmic operations on it. However, such an approach is not directly applicable in the case of abstract data structures.

In the present talk another approach will be indicated that has a close connection to works of Yiannis Moschovakis. It uses a further generalization of a part of the theory developed in his path-breaking paper "Abstract first order computability" (published in 1969).

Our generalization works with certain kinds of function-like objects instead of functions, namely with elements of so-called *iterative combinatory spaces*. The idea about using such kinds of objects arose in 1974 under the influence of the unprejudiced way of operating with multiple-valued functions in the above-mentioned paper.

The generalization was presented in a series of papers (the first one published in 1976), as well as in two books – the first one in Russian and the second one in English (they appeared in 1980 and in 1992, respectively).

Besides the notions of prime and search computability introduced in Moschovakis' paper, computability notions for many other cases are captured by that generalization, and some of the cases are of a probabilistic nature.

One probabilistic example was examined more thoroughly, namely the case of randomized computability in the set $\mathbb{N}$ with the random number generator $\frac{1}{2}, \frac{1}{2}, 0, 0, 0, \ldots$. The function-like objects considered in that particular case are random functions in $\mathbb{N}$, and the computable ones among them turned out to be exactly those random functions θ for which the set

$$\left\{ (m, n, x, y) \in \mathbb{N}^4 \;\middle|\; \frac{m}{n+1} < \theta(x, y) \right\}$$

is recursively enumerable.

One can see from this result that any random number generator which is a computable sequence of computable real numbers can be simulated by some probabilistic program using the random number generator $\frac{1}{2}, \frac{1}{2}, 0, 0, 0, \ldots$ or even any other one.

The idea of the generalization can be explained by giving a characterization of the Moskovakis' prime computable functions in the functional programming style propounded by Backus in his 1978 lecture.

Let B be a set, $o \notin B$, $B^\circ = B \cup \{o\}$. Assuming ordered pairs $\langle x, y\rangle$ are defined so that no element of B° is an ordered pair, a set $B^\star$ (*the Moschovakis extension* of B) is defined by induction:
(i) if $z \in B^\circ$ then $z \in B^\star$;
(ii) if $x \in B^\star$ and $y \in B^\star$, then $\langle x, y\rangle \in B^\star$.

Let $I, L, R, O : B^\star \to B^\star$ be defined as follows: $I(z) = z$, $O(z) = o$ for all $z \in B^\star$, $L(\langle x, y\rangle) = x$, $R(\langle x, y\rangle) = y$ for all $x, y \in B^\star$, $L(o) = R(o) = o$, $L(z) = R(z) = \langle o, o\rangle$ for all $z \in B$.

Let $\mathcal{F}_{\mathsf{p}}$ be the set of all partial one-argument functions in $B^\star$. For any two functions φ and ψ from $\mathcal{F}_{\mathsf{p}}$ we define their *composition* $\varphi\psi$ and their *combination* (φ, ψ) as $\lambda z.\varphi(\psi(z))$ and $\lambda z.\langle\varphi(z), \psi(z)\rangle$, respectively.

For any χ, φ, ψ from $\mathcal{F}_{\mathsf{p}}$ we define the function $(\chi \to \varphi, \psi)$ (the *branching to φ and ψ controlled by χ*) as follows: $(\chi \to \varphi, \psi)(x) = y$ iff

$$\text{either } x \in H_1 \text{ and } \varphi(x) = y,$$
$$\text{or } x \in H_2 \text{ and } \psi(x) = y,$$

where $H_1 = \chi^{-1}(B^\star \setminus B^\circ)$, $H_2 = \chi^{-1}(B^\circ)$.

If χ and σ are functions from $\mathcal{F}_{\mathsf{p}}$ then the *iteration of σ controlled by χ* is, by definition, the least solution θ of the equation

$$\theta = (\chi \to \theta\sigma, I),$$

where $\mathcal{F}_{\mathsf{p}}$ is considered with the usual partial ordering. A more explicit description of this θ reads as follows: $\theta(x) = y$ iff there are a non-negative integer k and a sequence $(z_0, z_1, \ldots, z_k)$ of elements of $B^\star$ such that $z_0 = x$, $z_k = y \in H_2$, $z_i \in H_1$ and $z_{i+1} = \sigma(z_i)$ for $i = 0, 1, \ldots, k-1$, where H_1 and H_2 are the same as in the definition of branching. This description shows the intuitive meaning of iteration.

Characterization of the prime computable functions. Let A be any subset of $B^\star$, and $\varphi, \psi_1, \ldots, \psi_l$ be some functions from $\mathcal{F}_\mathsf{p}$. Then $\varphi \in PC(A, \psi_1, \ldots, \psi_l)$ iff φ can be obtained by means of composition, combination, branching and iteration from the functions I, L, R, O, $\psi_1, \ldots, \psi_l$ and some constant total functions from $\mathcal{F}_\mathsf{p}$ whose values belong to A.

Let us now consider random functions in $B^\star$ instead of ordinary partial functions. We shall embed the set $\mathcal{F}_\mathsf{p}$ into the set $\mathcal{F}_\mathsf{r}$ of the random functions in $B^\star$ by replacing any function from $\mathcal{F}_\mathsf{p}$ with the characteristic function of its graph (for example, we shall admit that $I(x, y) = 1$ if $y = x$, and $I(x, y) = 0$ otherwise). Under this embedding the partial ordering of random functions as real-valued functions extends in an intuitively acceptable way the partial ordering in $\mathcal{F}_\mathsf{p}$. The operations of composition, combination and branching also have intuitively acceptable extensions in $\mathcal{F}_\mathsf{r}$ (the extensions are shown on the next slide).

$$\varphi\psi = \lambda xy.\sum_{t\in B^\star}\varphi(t,y)\psi(x,t),$$

$$(\varphi,\psi)(x,y) = \begin{cases} \varphi(x,u)\psi(x,v) & \text{if } y=\langle u,v\rangle,\\ 0 & \text{if } y\in B^\circ,\end{cases}$$

$$(\chi\to\varphi,\psi) = \lambda xy.\,H_1(x)\varphi(x,y)+H_2(x)\psi(x,y),$$

where

$$H_1(x)=\sum_{t\in B^\star\setminus B^\circ}\chi(x,t),\qquad H_2(x)=\sum_{t\in B^\circ}\chi(x,t).$$

For any χ and σ in $\mathcal{F}_{\mathsf{r}}$ the equation

$$\theta=(\chi\to\theta\sigma,I)$$

again has a least solution, and it will be called again the *iteration of φ controlled by χ*. The iteration operation defined in this way is an extension of the iteration operation defined before in $\mathcal{F}_{\mathsf{p}}$. Again a more explicit description of the iteration showing its intuitive acceptability can be written (this description is on the next slide).

If θ is the iteration of σ controlled by χ then $\theta = \iota_0 + \iota_1 + \iota_2 + \cdots$, where

$$\iota_k(x,y) = H_2(y) \sum_{\bar{z} \in S_{k,x,y}} \prod_{i=0}^{k-1} H_1(z_i)\sigma(z_i, z_{i+1}),$$

H_1 and H_2 are the same as in the definition of branching, and $S_{k,x,y}$ is the set of all sequences $\bar{z} = (z_0, z_1, \ldots, z_k)$ of elements of $B^\star$ such that $z_0 = x$ and $z_k = y$.

The introduced operations on random functions in $B^\star$ can be used to define a notion of relative computability for such functions. Suppose Ψ is some subset of $\mathcal{F}_\mathsf{r}$. An element φ of $\mathcal{F}_\mathsf{r}$ will be called *computable in* Ψ if φ can be obtained by means of composition, combination, branching and iteration from some elements of the set $\{I, L, R, O\} \cup \Psi$. The intuitive acceptability of this notion can be seen from the intuitive acceptability of the operations mentioned in its definition.

An arguments in favour of the generality and the convenience of the introduced notion of computability: it is an instance of the computability notion of the theory of iterative combinatory spaces, and in that theory, for example, the first recursion theorem and a normal form theorem hold.

The natural numbers will have the representation in $B^\star$ chosen by Moschovakis, namely the number 0 will bs identified with the element o, and $k+1$ will be identified with $\langle k, o\rangle$. A random number generator $p_0, p_1, p_2, \ldots$ will be represented by the random function π such that $\pi(x, y) = p_y$ if y is a natural number, and $\pi(x, y) = 0$ otherwise (clearly π does not depend on its first argument).

Finite sequences of elements of $B^\star$ (in particular finite sequences of natural numbers) will be encoded as follows: the empty sequence ε has code o, and whenever a sequence has code u, then the new sequence obtained by appending t as a last term has code $\langle u, t\rangle$.

If we are interested in random functions in $B^\star$ that are computable with using a random number generator $p_0, p_1, p_2, \ldots$ then it is appropriate to consider a probabilistic analog of the functions from $\mathcal{F}_{\mathsf{p}}$ belonging to $PC(A, \psi_1, \ldots, \psi_l)$, namely the elements of $\mathcal{F}_{\mathsf{r}}$ that are computable in the set consisting of $\psi_1, \ldots, \psi_l$, of the total constant functions from $\mathcal{F}_{\mathsf{p}}$ with values in A and of the random function representing the generator. We shall call these elements of $\mathcal{F}_{\mathsf{r}}$ *probabilistically computable from* A *in* $\psi_1, \ldots, \psi_l$ *with using the generator* $p_0, p_1, p_2, \ldots$

No straightforward analog of the result concerning randomized computability in $\mathbb{N}$ holds for these random functions without some additional assumption. However, they will be characterized in another way, namely in terms of prime computability of the function value on the base of the value of the argument and certain finite sequences of natural numbers.

Let δ be a mapping of $(B^\star)^2$ into the power set of the set of all finite sequences of natural numbers, and let δ have the property that, whenever a sequence from $\delta(x, y_1)$ is a beginning of some sequence from $\delta(x, y_2)$, then $y_1 = y_2$ and the two sequences coincide. We shall call any such mapping δ a *variants diagram*. A way of constructing certain variants diagrams will be described on the next slide.

Let τ be a function from $\mathcal{F}_{\mathsf{p}}$, and x be an element of $B^\star$. A finite sequence of natural numbers will be called *admissible for x according to τ* if the pair $\langle x, u\rangle$, where u is the code of the sequence, belongs to $\tau^{-1}(B^\circ)$, and any pair $\langle x, v\rangle$, where v is the code of some proper beginning of the sequence, belongs to $\tau^{-1}(B^\star \setminus B^\circ)$. Clearly no proper beginning of a sequence admisible for x according to τ can be also admissible for x according to τ.

Let ρ be also a function from $\mathcal{F}_{\mathsf{p}}$, and y be also an element of $B^\star$. Then a finite sequence of natural numbers will be called *admissible for returning y on x according to τ and ρ* if this sequence is admiissible for x according to τ, and the equality $\rho(\langle x, u\rangle) = y$, where u is the code of the sequence, holds. We shall denote by $Adm_{\tau,\rho}(x, y)$ the set of all such finite sequences. It is easy to see that the mapping $\lambda xy.\, Adm_{\tau,\rho}(x, y)$ is a variants diagram.

Suppose now that a random number generator $p_0, p_1, p_2, \ldots$ is given. The *probability* $P(\bar{k})$ of a finite sequence $\bar{k} = (k_0, k_1, \ldots, k_{m-1})$ of natural numbers will be defined by setting

$$P(\bar{k}) = p_{k_0} p_{k_1} \cdots p_{k_{m-1}}.$$

Then for any variants diagram δ a corresponding random function in $B^\star$ can be constructed, namely

$$\lambda xy. \sum_{\bar{k} \in \delta(x,y)} P(\bar{k}).$$

In particular, if for any x and y in $B^\star$ we set

$$\theta_{\tau,\rho}(x,y) = \sum_{\bar{k} \in Adm_{\tau,\rho}(x,y)} P(\bar{k})$$

then $\theta_{\tau,\rho}$ will be a random function in $B^\star$.

Theorem. A random function in $B^\star$ is probabilistically computable from A in $\psi_1, \ldots, \psi_l$ with using the generator $p_0, p_1, p_2, \ldots$ iff it can be represented in the form $\theta_{\tau,\rho}$ with τ and ρ belonging to $\mathcal{F}_{\mathsf{p}} \cap PC(A, \psi_1, \ldots, \psi_l)$.

Proof of the "if"-part. Let π be the random function representing the generator $p_0, p_1, p_2, \ldots$, and ι be the iteration of the random function (I, π) controlled by τ. Then $\theta_{\tau,\rho} = \rho\iota(I, O)$, hence $\theta_{\tau,\rho}$ is probabilistically computable from A in $\psi_1, \ldots, \psi_l$ with using the given generator, whenever $\tau, \rho \in \mathcal{F}_{\mathsf{p}} \cap PC(A, \psi_1, \ldots, \psi_l)$.

The proof of the "only if"-part is by induction that follows the construction of the random functions in $B^\star$ that are probabilistically computable from A in $\psi_1, \ldots, \psi_l$ with using the given generator. If θ is some of the functions $I, L, R, O, \psi_1, \ldots, \psi_l$ or some total constant function with value in A then $\theta = \theta_{\tau,\rho}$ with $\tau = R$, $\rho = \theta L$. If θ is the random function representing the generator $p_0, p_1, p_2, \ldots$, then $\theta = \theta_{\tau,\rho}$ with $\tau = (R \to LR, (R, R))$, $\rho = R^2$. The inductive steps are by constructing representations in the needed form for the composition, the combination, the branching and the iteration of random functions in $B^\star$ on the base of their representations in this form.

Remark. If p_0 and p_1 are the only non-zero terms of the generator $p_0, p_1, p_2, \ldots$ then one can use a smaller set than $Adm_{\tau,\rho}(x, y)$ in the definition of $\theta_{\tau,\rho}$. Namely the set of the sequences in $Adm_{\tau,\rho}(x, y)$ consisting only of 0's and 1's is sufficient in this case, since any other sequence in $Adm_{\tau,\rho}(x, y)$ will have probability 0. Of course, in the particular case of $p_0 = p_1 = \frac{1}{2}$ the probability of any m-term sequence of 0's and 1's will be 2^{-m}.

We can construct an iterative combinatory space, where the variant diagrams are the function-like objects.

The general definition of combinatory space.
A combinatory space is a 9-tuple

$$(\mathcal{F}, I, \mathcal{C}, \Pi, L, R, \Sigma, T, F),$$

where $\mathcal{F}$ is a partially ordered semigroup, I is its identity, $\mathcal{C} \subseteq \mathcal{F}$, $\Pi : \mathcal{F}^2 \to \mathcal{F}$, $\Sigma : \mathcal{F}^3 \to \mathcal{F}$, $L, R, T, F \in \mathcal{F}$, and the following conditions are identically satisfied, when $\varphi, \psi, \theta, \chi$ range over $\mathcal{F}$, and c, d range over $\mathcal{C}$:

$$\forall c(\varphi c \geq \psi c) \;\Rightarrow\; \varphi \geq \psi,$$
$$\Pi(c,d) \in \mathcal{C},\; L\,\Pi(c,d) = c,\; R\,\Pi(c,d) = d,$$
$$\Pi(\varphi,\psi)c = \Pi(\varphi c, \psi c),$$
$$\Pi(I,\psi c)\theta = \Pi(\theta,\psi c),\; \Pi(c,I)\theta = \Pi(c,\theta),$$
$$T \neq F,\; Tc \in \mathcal{C},\; Fc \in \mathcal{C},$$
$$\Sigma(T,\varphi,\psi) = \varphi,\; \Sigma(F,\varphi,\psi) = \psi,$$
$$\theta\,\Sigma(\chi,\varphi,\psi) = \Sigma(\chi,\theta\varphi,\theta\psi),$$
$$\Sigma(\chi,\varphi,\psi)c = \Sigma(\chi c, \varphi c, \psi c),$$
$$\Sigma(I,\varphi c,\psi c)\theta = \Sigma(\theta,\varphi c,\psi c),$$
$$\varphi \geq \psi,\; \theta \geq \chi \;\Rightarrow\; \Sigma(I,\varphi,\theta) \geq \Sigma(I,\psi,\chi).$$

The definition implies that multrplication, Π and Σ are monotonically increasing operations.

Example (the combinatory space of the variants diagrams). Let $\mathcal{F}$ consist of all variants diagrams. Let $\varphi \geq \psi$ mean that $\varphi(x,y) \supseteq \psi(x,y)$ for all $x, y \in B^\star$, and let

$$\varphi\psi = \lambda xy. \bigcup_{t \in B^\star} \varphi(t,y)\psi(x,t).$$

The set $\mathcal{F}_\mathsf{p}$ is embedded into $\mathcal{F}$ by replacing any function from $\mathcal{F}_\mathsf{p}$ with the mapping θ such that $\theta(x,y) = \{\varepsilon\}$ if y is the function value at x, and $\theta(x,y) = \emptyset$ otherwise. In particular, the functions I, L, R will be regarded as elements of $\mathcal{F}$. Let the set $\mathcal{C}$ consist of all total constant functions from $\mathcal{F}_\mathsf{p}$. We set also

$$\Pi(\varphi,\psi)(x,y) = \begin{cases} \varphi(x,u)\psi(x,v) & \text{if } y = \langle u,v\rangle, \\ \emptyset & \text{if } y \in B^\circ, \end{cases}$$

$$T = \Pi(O,O), \quad F = O,$$

$$\Sigma(\chi,\varphi,\psi) = \lambda xy.\, H_1(x)\varphi(x,y) \cup H_2(x)\psi(x,y),$$

where

$$H_1(x) = \bigcup_{t \in B^\star \setminus B^\circ} \chi(x,t), \quad H_2(x) = \bigcup_{t \in B^\circ} \chi(x,t).$$

If in a combinatory space the equation

$$\theta = \Sigma(\chi, \theta\sigma, I)$$

has a least solution θ for any σ and χ, and this solution has certain additional nice properties, then the combinatory space is called *iterative*, and the least solution in question is called *the iteration of σ controlled by χ*.

A sufficient condition for iterativeness of a combinatory space is the existence in its semigroup of a zero element that is its least element, the existence of a least upper bound of any monotonically increasing sequence, and the continuity of the three operations from the definition with respect to such least upper bounds. This condition is satisfied for the combinatory spaces explicitly or implicitly mentioned in the present talk.

In the combinatory space of the variants diagrams, if θ is the iteration of σ controlled by χ then $\theta = \iota_0 \cup \iota_1 \cup \iota_2 \cup \ldots$, where

$$\iota_k(x,y) = H_2(y) \bigcup_{\bar{z} \in S_{k,x,y}} \prod_{i=0}^{k-1} H_1(z_i)\sigma(z_i, z_{i+1}),$$

H_1 and H_2 are the same as in the definition of Σ, and $S_{k,x,y}$ is the same set as in the expression for the iteration in $\mathcal{F}_{\mathsf{r}}$.

Suppose again that a random number generator $p_0, p_1, p_2, \ldots$ is given. Then we can consider a homomorphism of the combinatory space of the variants diagrams into the combinatory space of the random functions in $B^\star$, namely the mapping that transforms any variants diagram into the random function corresponding to it. This homomorphism preserves also iterations.

Also a homomorphism of the combinatory space of the random functions in $B^\star$ into the combinatory space of the partial multiple-valued mappings of $B^\star$ into $B^\star$ deserves attention. It can be obtained by considering for any random function θ in $B^\star$ the corresponding mapping

$$\lambda x.\{y \mid \theta(x,y) > 0\}.$$

This homomorphism also preserves iterations.

Let the random number generator $p_0, p_1, p_2, \ldots$ be such that the set $\{k \mid p_k > 0\}$ is recursively enumerable. Then the random functions in $B^\star$ that are probabilistically computable from A in $\psi_1, \ldots, \psi_l$ with using this generator go through the above-mentioned homomorphism into partial multiple-valued functions belonging to $PC(A, \psi_1, \ldots, \psi_l, \lambda x.\mathbb{N})$. (The last kind of relative computability of multiple-valued functions is, roughly speaking, equivalent to the Friedman-Shepherdson one.)

References

• Backus, J.: Can programming be liberated from the von Neumann style? A functional style and its algebra of programs. *Comm. of the ACM* **21** (1978) 613–641

• Moschovakis, Y. N.: Abstract first order computability. I. *Trans. Amer. Math. Soc.* **138** (1969) 427–464

• Skordev, D.: Recursion theory on iterative combinatory spaces. *Bull. Acad. Polon. Sci., Sér. Sci. Math. Astr. Phys.* **24** (1976) 23–31

• Skordev, D.: Computability in Combinatory Spaces. Kluwer Academic Publishers, Dordrecht/Boston/London, 1992

• von Neumann, J.: Various techniques used in connection with random digits. Notes by G. E. Forsythe. *National Bureau of Standards. Applied Math. Series* 12:36–38, 1951. Reprinted in von Neumann's Collected Works, vol. 5, Pergamon Press, 1963, 768–770