Electronic Notes in Computer Science 1 (1998)

Domain Theory in Learning Processes

A. Edalat [1]

Department of Computing
Imperial College of Science, Technology and Medicine
180 Queen's Gate
London SW7 2BZ UK

Abstract

We present applications of domain theory in stochastic learning automata and in neural nets. We show that a basic probabilistic algorithm, the so-called linear reward-penalty scheme, for the binary-state stochastic learning automata can be modelled by the dynamics of an iterated function system on a probabilistic power domain and we compute the expected value of any continuous function in the learning process. We then consider a general class of, so-called forgetful, neural nets in which pattern learning takes place by a local iterative scheme, and we present a domain-theoretic framework for the distribution of synaptic couplings in these networks using the action of an iterated function system on a probabilistic power domain. We then obtain algorithms to compute the decay of the embedding strength of the stored patterns.

1 Introduction

The probabilistic power domain was introduced in [21] and developed in [20,14] for studying probabilistic computation, in order to provide semantics for probabilistic programming languages. In [3,5], a general framework was established for application of the probabilistic power domain in computation beyond semantics. It was shown that the probabilistic power domain of the upper space of any compact metric space provides a domain-theoretic framework for measure and integration theory on the metric space. The basic idea is that any bounded measure on the metric space can be obtained as the least upper bound of an increasing chain of linear combinations of point measures, or valuations, on the upper space. This leads to a theory of generalised Riemann integration. These results have provided new domain-theoretic and constructive tech-

[1] This work has been supported by EPSRC project "Constructive Approximation Using Domain Theory".

niques and new algorithms for computation in several fields of mathematics and physics [7,4].

The present paper deals with applications in learning algorithms. We will first consider stochastic learning automata, which have been studied since 1950's and can be regarded as the precursors of neural nets. We show that a basic probabilistic algorithm, the so-called linear reward-penalty scheme, for the binary-state stochastic learning automata [19,16] can be modelled by the dynamics of an iterated function system on a probabilistic power domain. As a result, we obtain a precise characterisation of the distribution of the states at any stage of the learning process. This enables us to compute the expected value of any continuous function in the learning process. We will then consider a general class of, so-called forgetful, neural nets in which pattern learning takes place by a local iterative scheme [10,2]. Such nets have been proposed in order to overcome the phenomenon of catastrophic forgetting in the well-known Hopfield model. We present a domain-theoretic framework for the distribution of synaptic couplings in forgetful nets. This framework also uses the action of an iterated function system on a probabilistic power domain and it enables us to analyse the local behaviour of the distribution of the synaptic couplings. We then obtain algorithms to compute the decay of the embedding strength of the stored patterns, a quantity which is used in computing the storage capacity of the network.

2 Domain-theoretic Framework

In this section, we establish the domain-theoretic framework for measure and integration theory that we need in this paper by reviewing the work in [14,3,5,6] and by presenting some new results.

Let X be a compact metric space, and let $(\mathbf{U}X, \supseteq)$ be its upper space, i.e. the non-empty compact subsets of X ordered by reverse inclusion, which is a bounded complete ω-continuous dcpo (directed complete partial order) with bottom X. The Scott topology on $\mathbf{U}X$ has basic open sets

$$\Box a = \{C \in \mathbf{U}X \mid C \subseteq a\}$$

for any open set $a \subseteq X$. The singleton map $s : X \to \mathbf{U}X$ embeds X onto the set of maximal elements of $\mathbf{U}X$. Similarly, the set of normalised measures (or probability measures or probability distributions) on X can be embedded into the set of normalised measures on $\mathbf{U}X$. This is done using the normalised probabilistic power domain, which we now aim to define.

A *valuation* on a topological space Y is a map $\nu : \Omega(Y) \to [0, \infty)$ which satisfies:

(i) $\nu(a) + \nu(b) = \nu(a \cup b) + \nu(a \cap b)$,
(ii) $\nu(\emptyset) = 0$, and
(iii) $a \subseteq b \Rightarrow \nu(a) \leq \nu(b)$.

A *normalised* valuation ν further satisfies $\nu(Y) = 1$. A *Scott-continuous* valuation is a valuation such that whenever $A \subseteq \Omega(Y)$ is a directed set (wrt $\subseteq$) of open sets of Y, then

$$\nu(\bigcup_{O \in A} O) = \sup_{O \in A} \nu(O).$$

For any $b \in Y$, the *point valuation* based at b is the valuation $\delta_b : \Omega(Y) \to [0, \infty)$ defined by

$$\delta_b(O) = \begin{cases} 1 \text{ if } b \in O \\ 0 \text{ otherwise.} \end{cases}$$

Any finite linear combination $\sum_{i=1}^{n} r_i \delta_{b_i}$ of point valuations δ_{b_i} with constant coefficients $r_i > 0$, $(1 \leq i \leq n)$, is a Scott-continuous valuation on Y; we call it a *simple valuation*.

The *normalised probabilistic power domain*, $\mathbf{P}^1 Y$, of a topological space Y consists of the set of normalised Scott-continuous valuations ν on Y and is ordered as follows: $\mu \sqsubseteq \nu$ iff for all open sets O of Y, $\mu(O) \leq \nu(O)$. The partial order $(\mathbf{P}^1 Y, \sqsubseteq)$ is a dcpo in which the lub of a directed set $\langle \mu_i \rangle_{i \in I}$ is given by $\bigsqcup_i \mu_i = \nu$ where for any open subset $O \in \Omega(Y)$ we have

$$\nu(O) = \sup_{i \in I} \mu_i(O).$$

If D is an ω-continuous dcpo with bottom $\perp$, then $\mathbf{P}^1 D$ is also an ω-continuous dcpo with bottom $\delta_\perp$ and has a basis consisting of simple valuations [5]. Furthermore, any $\mu \in \mathbf{P}^1 D$ extends uniquely to a Borel measure on D. For convenience, we denote this unique extension by μ as well.

For two simple valuations

$$\mu_1 = \sum_{b \in B} r_b \delta_b \qquad \mu_2 = \sum_{c \in C} s_c \delta_c$$

in $\mathbf{P}^1 Y$, where B, C are finite subsets of D, we have by the *Splitting Lemma* [14,5]: $\mu_1 \sqsubseteq \mu_2$ iff, for all $b \in B$ and all $c \in C$, there exists a nonnegative number $t_{b,c}$ such that

$$\sum_{c \in C} t_{b,c} = r_b \qquad \sum_{b \in B} t_{b,c} = s_c \tag{1}$$

and $t_{b,c} \neq 0$ implies $b \sqsubseteq c$. We can consider any $b \in B$ as a source with mass r_b, any $c \in C$ as a sink with mass s_c, and the number t_{bc} as the flow of mass from b to c. Then, the above property can be regarded as conservation of total mass.

Since $(\mathbf{U}X, \supseteq)$ is an ω-continuous dcpo with bottom X, $\mathbf{P}^1\mathbf{U}X$ is an ω-continuous dcpo with bottom δ_X. Any Borel subset $B \subseteq X$ induces, via the singleton map s, a Borel subset $s(B) \subseteq \mathbf{U}X$. A valuation $\mu \in \mathbf{P}^1\mathbf{U}X$ is said to be *supported* in $s(X)$ if its unique extension to a Borel measure on $\mathbf{U}X$ satisfies $\mu(s(X)) = 1$. We have $\mu \in \mathbf{P}^1\mathbf{U}X$ is supported in $s(X)$ iff

$$\mu(s(a)) = \mu(\Box a) \tag{2}$$

for all open subsets $a \subseteq X$. If μ is supported in $s(X)$, then the *support* of μ is the set of points $y \in s(X)$ such that $\mu(O) > 0$ for any Scott-neighbourhood $O \subseteq \mathbf{U}X$ of y. Any element of $\mathbf{P}^1\mathbf{U}X$ which is supported in $s(X)$ is a maximal element of $\mathbf{P}^1\mathbf{U}X$ [3, Proposition 5.18]; we denote the set of all valuations which are supported in $s(X)$ by $\mathbf{S}^1X$. We can identify $\mathbf{S}^1X$ with the set $\mathbf{M}^1X$ of normalised Borel measures on X as follows. Let

$$\begin{aligned} e : \mathbf{M}^1X &\to \mathbf{S}^1X \\ \mu &\mapsto \mu \circ s^{-1} \end{aligned}$$

and

$$\begin{aligned} j : \mathbf{S}^1X &\to \mathbf{M}^1X \\ \nu &\mapsto \nu \circ s. \end{aligned}$$

Theorem 2.1 *[3, Theorem 5.21] The maps e and j are well-defined and induce an isomorphism between $\mathbf{S}^1X$ and $\mathbf{M}^1X$.*

For $\mu \in \mathbf{M}^1X$ and an open subset $a \subseteq X$, Equation (2) implies

$$\mu(a) = e(\mu)(s(a)) = e(\mu)(\Box a). \tag{3}$$

Furthermore, it follows from Theorem 2.1 that there exists an increasing chain of simple valuations $\langle \nu_i \rangle_{i \geq 0}$ with

$$\mu \circ s^{-1} = \bigsqcup_{i \geq 0} \nu_i. \tag{4}$$

In other words, any probability measure on X is the least upper bound of a chain of simple valuations on $\mathbf{U}X$. (For convenience, we write μ instead of $\mu \circ s^{-1}$.)

There is also the converse problem: Suppose $\mathcal{A} \subseteq \mathbf{P}^1\mathbf{U}X$ is a directed set of simple valuations, and we would like to know when the lub $\bigsqcup \mathcal{A}$ of $\mathcal{A}$ determines a Borel measure on X; in other words, under what conditions we have $\bigsqcup \mathcal{A} \in \mathbf{S}^1X$. The following proposition gives the necessary and sufficient condition for this. We will denote the diameter of any compact subset $c \subseteq X$ by $|c|$.

Proposition 2.2 *The following conditions are equivalent.*

(i) $\bigsqcup \mathcal{A} \in \mathbf{S}^1X$.
(ii) *For all* $\alpha > 0$ *and all* $\beta > 0$, *there exists* $\sum_{a\in A} r_a\delta_a \in \mathcal{A}$ *with*

$$\sum_{|a|\geq\beta} r_a < \alpha.$$

Given a poset D, a sequence $\langle A_i\rangle_{i\geq 0}$ of finite subsets $A_i \subseteq D$ $(i \geq 0)$ is a *tower* in D if $A_i \sqsubseteq_{EM} A_{i+1}$ for all $i \geq 0$, where $sqsubseteq_{EM}$ is the Egli-Milner ordering. Suppose now we have an increasing chain $\langle \nu_i\rangle_{i\geq 0}$ of simple valuations

$$\nu_i = \sum_{b\in B_i} r_{b,i}\delta_b$$

in $\mathbf{P}^1\mathbf{U}X$. Using Equation (1), it is easy to see that $\langle B_i\rangle_{i\geq 0}$ is a tower in $\mathbf{U}X$; we call it the *associated tower* of the chain $\langle \nu_i\rangle_{i\geq 0}$. Clearly, the lub in $\mathbf{U}X$ of any chain in the associated tower is an element of $\mathbf{U}X$, which is to say that the intersection of a shrinking sequence of non-empty compact subsets of X is a non-empty compact subset. Using Proposition 2.2 and König's Lemma, we can prove the following useful result.

Proposition 2.3 *Suppose* $\langle \nu_i\rangle_{i\geq 0} \in \mathbf{P}^1\mathbf{U}X$ *is an increasing chain of simple valuations. If the lub of any chain in the associated tower in a singleton subset of* X, *then the lub of the chain* $\langle \nu_i\rangle_{i\geq 0}$ *is in* $\mathbf{S}^1X$.

An *IFS with probabilities*,

$$\{X; f_1, \ldots, f_N; p_1, \ldots, p_N\},$$

is given by a finite number of continuous maps $f_i : X \to X$ $(1 \leq i \leq N)$ on a compact metric space X, such that each f_i is assigned a probability weight p_i with $0 < p_i < 1$ and $\sum_{i=1}^N p_i = 1$. The IFS is said to be *hyperbolic* if the maps f_i are all contracting. An hyperbolic IFS with probabilities gives rise to a unique invariant Borel measure on X [1].

It was shown in [3, Theorem 6.2], that this unique invariant measure is the fixed point of the *transition operator*

$$\begin{aligned} T : \mathbf{P}^1\mathbf{U}X &\to \mathbf{P}^1\mathbf{U}X \\ \mu &\mapsto T(\mu) \end{aligned}$$

defined by

$$T(\mu)(O) = \sum_{i=1}^{N} p_i \mu(f_i^{-1}(O)).$$

This fixed point can be written as $\bigsqcup_{n\geq 0} \mu_n$ where $\mu_0 = \delta_X$ and for $n \geq 1$,

$$\mu_n = T^n \delta_X = \sum_{i_1,i_2,\ldots,i_n=1}^{N} p_{i_1} p_{i_2} \ldots p_{i_n} \delta_{f_{i_1} f_{i_2} \cdots f_{i_n} X}.$$

Therefore, the unique invariant measure of the hyperbolic IFS with probabilities can be expressed as the lub of a canonical ω-chain of simple valuations in $\mathbf{P}^1\mathbf{U}X$. The invariant measure is obtained in a natural way. We start with the valuation δ_X, which contains no information, and then, at each stage of iteration, we obtain more information about the invariant measure. The chain of valuations gives rise to a tree, called the IFS tree with transitional probabilities as in Figure 1, whose nth level represents the nth valuation in the chain. The edges from the nodes on the nth level to those on the $n+1$st are labelled by probabilities p_i, indicating the ratios by which mass is distributed from each node to its children (cf. the Splitting Lemma, Equation (1)).

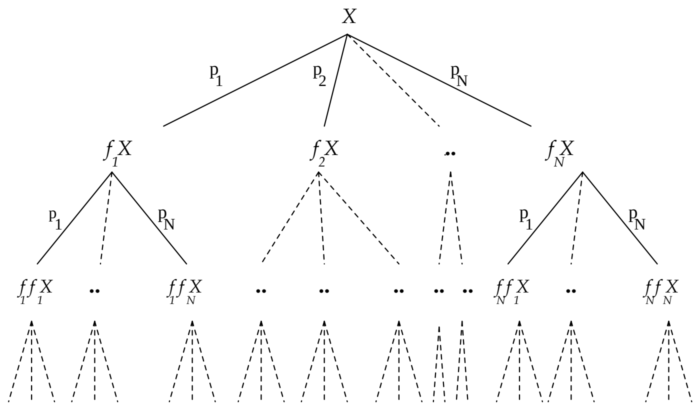

Figure 1. The IFS tree with transitional probabilities.

What can we say if the IFS is not hyperbolic? Using Proposition 2.3, we can deduce the following.

Proposition 2.4 *An IFS with probabilities has a unique invariant measure if* *the lub of every branch of the associated IFS tree is a singleton set.*

The above domain-theoretic framework also provides a finite algorithm for generating on the digitised screen the invariant measure of an IFS with probabilities satisfying the condition of Proposition 2.4. Since the lub of any branch is a singleton subset, every branch in the digitised screen terminates in a leaf which is of pixel size. The total weight assigned to each pixel z is the sum

$$\sum_{f_{i_1} f_{i_2} \cdots f_{i_n} X \subseteq z} p_{i_1} p_{i_2} \ldots p_{i_n}$$

of all weights $p_{i_1} p_{i_2} \ldots p_{i_n}$ of the leaves $f_{i_1} f_{i_2} \ldots f_{i_n} X$ which lie in z. This algorithm generalises that in [11] which holds for hyperbolic IFSs.

The generalised Riemann integral, developed in [5], gives a simple formula to compute the expected value of any continuous function with respect to the invariant measure of an IFS with probabilities. First assume that we have an IFS which satisfies the condition of Proposition 2.4 and let $g : X \to R$ be a real-valued function continuous almost everywhere with respect to the invariant measure μ^*. Fix a point $x \in X$. Then, we have the generalised Riemann sum for $T^n \delta_X$ given by

$$S(g, T^n \delta_X) = \sum_{i_1, \ldots, i_n = 1}^{N} p_{i_1} \ldots p_{i_n} g(f_{i_1} \ldots f_{i_n} x) \tag{5}$$

and it follows that

$$S(g, T^n \delta_X) \to \int g \, d\mu^* \tag{6}$$

as $n \to \infty$. This result refines that in [11] which holds for an hyperbolic IFS and continuous functions. If the IFS is hyperbolic and g satisfies a Lipschitz condition, then we can obtain a finite algorithm to calculate the integral to any given accuracy as follows [6]. Suppose there exist $k > 0$ and $c > 0$ such that

$$|g(x) - g(y)| \leq c(d(x, y))^k$$

for all $x, y \in X$. When $k = 1$, we call c a *Lipschitz constant* for g. Let $\theta > 0$ be given. Then

$$|S(g, T^n \delta_X) - \int g d\mu^*| \leq \theta \tag{7}$$

for $n = \lceil \log((\theta/c)^{1/k}/|X|)/\log s \rceil$, where s is the contractivity of the hyperbolic IFS and $\lceil a \rceil$ is the least integer greater than or equal to a.

3 Stochastic Learning Automata

Stochastic learning automata [15,19,16] are learning systems which are the precursors of modern neural nets. We will first give a brief review. A learning automaton operates in an unknown probabilistic environment with which it is connected by a feedback loop. Each action by the automaton receives a simple response from the environment which indicates its success or failure. This response is taken as input to the automaton which will then reevaluate its action. The automaton has to learn which actions have the highest probability of success. Learning takes place if the automaton tends to increase its chance of success.

More formally, a stochastic learning automata consists of the following.

(i) It has a finite set A of *actions*, which for convenience we denote by

$$A = \{1, 2, \ldots, r\}.$$

(ii) The *state*, $\mathbf{s}$, of the automaton at each step of learning is given by a probability vector,

$$\mathbf{s} = (s_1, s_2, \ldots, s_r), \qquad \sum_{i=1}^{r} s_i = 1.$$

The set of states S is therefore the $r-1$ dimensional simplex in R^r which is endowed with the induced Euclidean metric, and is, hence, a compact metric space.

(iii) At each step of learning, the automaton takes action $i \in A$ with probability s_i, where

$$\mathbf{s} = (s_1, s_2, \ldots, s_r) \in S$$

is the current state.

(iv) The environment receives action $i \in A$ as input and responds by sending a simple success or failure signal $b \in \{0, 1\}$. Here 1 and 0 denote success and failure respectively. The environment is characterised by a set of success probabilities

$$\{\pi_1, \pi_2, \ldots, \pi_r\}$$

where π_i is the probability of a success signal (i.e. $b = 1$) for action i. The probability of a failure signal ($b = 0$) for action i is therefore $1 - \pi_i$. The probabilities π_i are not known *a priori* of course. However for simplicity we assume that they are time independent (the environment is said to be stationary) and that

$$\sum_{i=1}^{r} \pi_i = 1.$$

(v) If the automaton takes action $i \in A$ and the environment responds with the signal $b \in \{0,1\}$, then the pair $e = (i,b)$ is called an *event*. The set of events E is therefore $E = A \times \{0,1\}$. If the automaton is in the state

$$\mathbf{s} = (s_1, s_2, \ldots, s_r) \in S,$$

then the probability of the event $e = (i,b)$ is easily seen to be given by

$$p_e(\mathbf{s}) = s_i(\pi_i b + (1-\pi_i)(1-b)).$$

And we have

$$\sum_{e \in E} p_e(\mathbf{s}) = 1$$

for each $\mathbf{s} \in S$.

(vi) A learning algorithm for the automaton is a set of continuous maps

$$\{f_e : S \to S \mid e \in E\}.$$

If the automaton is in the state $\mathbf{s}$ and the event $e = (i,b) \in E$ takes place, then the automaton modifies its state by the mapping f_e, i.e. $\mathbf{s} \mapsto f_e(\mathbf{s})$ and the new state is therefore $f_e(\mathbf{s})$. This means that we have the following dynamical equation for updating the state $\mathbf{s}_n$ of automaton at the step n of learning when an event $e_n \in E$ occurs

$$\mathbf{s}_{n+1} = f_{e_n}(\mathbf{s}_n)$$

where f_{e_n} is selected with probability $p_{e_n}(\mathbf{s}_n)$.

A basic learning algorithm is the *linear reward-penalty scheme* which is defined as follows. For $e = (i,1)$,

$$\begin{aligned} (f_e(\mathbf{s}))_i &= s_i + \lambda(1-s_i) \\ (f_e(\mathbf{s}))_j &= (1-\lambda)s_j \qquad \text{if } j \neq i \end{aligned}$$

and for $e = (i,0)$,

$$\begin{aligned} (f_e(\mathbf{s}))_i &= (1-\lambda)s_i \\ (f_e(\mathbf{s}))_j &= (1-\lambda)s_j + \lambda/(r-1) \quad \text{if } j \neq i \end{aligned}$$

where $0 < \lambda < 1$. Therefore, if action i is taken and rewarded by the environment, then s_i increases and all other states decrease. For a punishment response, s_i is decreased and the other states are increased. The binary-state linear reward-penalty automaton has been extensively studied since the work of Karlin [15]. In this case,

$$A = \{1, 2\},$$

$$S = \{\mathbf{s} \mid s_1 + s_2 = 1\},$$

$$E = \{(1,1), (2,1), (1,0), (2,0)\},$$

and the mappings $f_e : S \to S$ and the probabilities $p_e(\mathbf{s})$, for each $e \in E$, are given by

$$\begin{aligned}
f_{(1,1)}(\mathbf{s}) &= (s_1(1-\lambda)+\lambda, s_2(1-\lambda)) & p_{(1,1)}(\mathbf{s}) &= s_1\pi_1 \\
f_{(2,1)}(\mathbf{s}) &= (s_1(1-\lambda), s_2+\lambda(1-s_2)) & p_{(2,1)}(\mathbf{s}) &= s_2\pi_2 \\
f_{(1,0)}(\mathbf{s}) &= (s_1(1-\lambda), s_2(1-\lambda)+\lambda) & p_{(1,0)}(\mathbf{s}) &= s_1\pi_2 \\
f_{(2,0)}(\mathbf{s}) &= (s_1(1-\lambda)+\lambda, s_2(1-\lambda)) & p_{(2,0)}(\mathbf{s}) &= s_2\pi_1.
\end{aligned}$$

We note that $f_{(1,1)} = f_{(2,0)}$, and $p_{(1,1)}(\mathbf{s})+p_{(2,0)}(\mathbf{s}) = (s_1+s_2)\pi_1 = \pi_1$. Similarly, $f_{(2,1)} = f_{(1,0)}$, with $p_{(2,1)}(\mathbf{s})+p_{(1,0)}(\mathbf{s}) = (s_1+s_2)\pi_2 = \pi_2$. Therefore, the dynamics of the automaton is reduced to that of two maps with state-independent transition probabilities. More precisely, let

$$\begin{aligned}
f_1 : [0,1] &\to [0,1] \\
s &\mapsto s(1-\lambda)+\lambda
\end{aligned}$$

and

$$\begin{aligned}
f_2 : [0,1] &\to [0,1] \\
s &\mapsto s(1-\lambda).
\end{aligned}$$

Then the dynamics of the state s_1 is governed by the hyperbolic IFS with probabilities $([0,1]; f_1, f_2; \pi_1, \pi_2)$, and that of s_2 by $([0,1]; f_1, f_2; \pi_2, \pi_1)$. Since the automaton is probabilistic we are really interested in the *probability distribution* of the two states $s_1 \in [0,1]$ and $s_2 \in [0,1]$ at each stage of learning.

Domain theory provides the proper conceptual and technical framework to model the process of learning. We apply here the domain-theoretic setup, described in Section 2, for the step by step generation of the invariant probability distribution of an hyperbolic IFS with probabilities. Initially, there is no learning, and the distribution of each of the states is given by $\delta_{[0,1]}$. Then, at each stage of iteration of the probabilistic algorithm, new learning is achieved. The evolution of the distribution of the state s_1 is captured by the mapping

$$\begin{array}{rcl} L_1 : \mathbf{P}^1\mathbf{U}[0,1] \to & & \mathbf{P}^1\mathbf{U}[0,1] \\ \\ \mu & \mapsto & \pi_1\mu \circ f_1^{-1} + \pi_2\mu \circ f_2^{-1} \end{array}$$

and that of s_2 by the mapping

$$\begin{array}{rcl} L_2 : \mathbf{P}^1\mathbf{U}[0,1] \to & & \mathbf{P}^1\mathbf{U}[0,1] \\ \\ \mu & \mapsto & \pi_2\mu \circ f_1^{-1} + \pi_1\mu \circ f_2^{-1} \end{array}$$

The distributions of the states are refined at each step of the learning process; technically, this is expressed by the information relation $L_i^n\delta_{[0,1]} \sqsubseteq L_i^{n+1}\delta_{[0,1]}$ for each $n \geq 0$ and $i = 1, 2$. We have therefore the following theorem.

Theorem 3.1 *At the nth step of learning of the binary state linear reward-penalty automaton, the distribution of the state s_i is given by*

$$L_i^n\delta_{[0,1]} = \sum_{i_1,\ldots,i_n=1,2} p_{i_1}\ldots p_{i_n}\delta_{f_{i_1}\ldots f_{i_n}[0,1]}$$

for $i = 1, 2$. The limiting distribution is $\mu_i^* = \bigsqcup_{n\geq 0} L_i^n\delta_{[0,1]}$.

The nature of the limiting distribution depends on the parameter λ. This can easily be analysed using the above framework.

Theorem 3.2 *For $0 < \lambda < 1/2$, the support of μ_i^* ($i = 1, 2$) is a Cantor set; whereas for $1/2 \leq \lambda < 1$ the support of μ_i^* ($i = 1, 2$) is the whole unit interval.*

The above result confirms that in [15], which uses classical measure theory. Our framework, however, allows us to obtain the expected value of any real valued almost everywhere continuous function, $g : S \to R$, of the states of the automaton. Since $s_1 + s_2 = 1$, any such function can be regarded either as a function, $g_1 : [0,1] \mapsto R$, of s_1 or alternatively as a function, $g_2 : [0,1] \mapsto R$, of s_2. Using Equation (5) with $x = 0$, say, we have the generalised Riemann sum

$$S(g_i, T^n\delta_{[0,1]}) = \sum_{i_1,\ldots,i_n=1,2} p_{i_1}\ldots p_{i_n} g(f_{i_1}\ldots f_{i_n}(0))$$

for $i = 1, 2$, and by Equation (6) we obtain:

Proposition 3.3 *For each $i = 1, 2$, we have,*

$$\int g\, d\mu_1^* d\mu_2^* = \int g_i\, d\mu_i^* = \lim_{n\to\infty} S(g_i, T^n\delta_{[0,1]}).$$

If g is Lipschitz, we obtain a finite algorithm to obtain the integral up to a given threshold of accuracy as in Equation (7).

4 Forgetful Neural Nets

Neural networks present models for computational behaviour which resemble those of the brain. In this section, we give a brief review in order to describe the class of so-called forgetful nets. The study of neural nets began with the work of McCulloch and Pitts in 1943, who introduced the notion of a formal neuron as a two-state threshold element, with the firing state 1 and the non-firing state -1, say. Neuron i receives input $J_{ij}S_j$ from the neuron j, where S_j is the state of neuron j and J_{ij} is the *(synaptic) coupling* from neuron j to neuron i. Then, in an assembly of N neurons, the dynamics of the states of neurons is as follows:

$$S_i \text{ becomes } \begin{cases} 1 & \text{if } \sum_{j=1}^{N} J_{ij}S_j > T_i \\ -1 & \text{if } \sum_{j=1}^{N} J_{ij}S_j \leq T_i \end{cases} \tag{8}$$

where T_i is the threshold of neuron i. McCulloch and Pitts showed that networks of such elements can implement any logical function. An important step was later taken by Hebb in 1949, who suggested that a concept is represented in the brain by the firing activity of cell assembly and that learning takes place by modification of the couplings J_{ij} between the neurons. A variety of models for associative memory, pattern recognition and various classification problems were then developed.

Neural network models are of two different types of architecture. One is the feedforward networks, where neurons in one layer receive inputs only from the preceding layer. The other type is the multiply connected network with a lot of feedback between its elements.

An important breakthrough came with the work of Hopfield [12] in multiply connected networks. He assumed that the couplings are symmetric, i.e. $J_{ij} = J_{ji}$, and that the states of neurons change one at a time. Then, as in the theory of Ising spin and spin glasses in statistical physics, an energy function can be defined,

$$E = -\frac{1}{2}\sum_{i,j=1}^{N} J_{ij}S_iS_j + \sum_{i=1}^{N} T_iS_i$$

which under the dynamic equations (8) cannot increase: If ΔS_i is the change in the state S_i, then the change in the energy is easily seen to be given by

$$\Delta E = -\Delta S_i (\sum_{j=1}^{N} J_{ij} S_j - T_i) \leq 0.$$

Therefore, the states of neurons flip and the energy decreases until a stable local minimum of the energy is reached. In order to capture the concept of learning and memory, Hopfield made the final assumption that the stable local minima, or the "wells" of the energy function, are associated with the stored key patterns. Each stored pattern $\mathbf{X}$ is characterised by specifying its component state $X_i = \pm 1$ for each neuron i. Figure 2 depicts a schematic picture of an energy function with three stable local minima representing three stored patterns $\mathbf{X}^1$, $\mathbf{X}^2$ and $\mathbf{X}^3$.

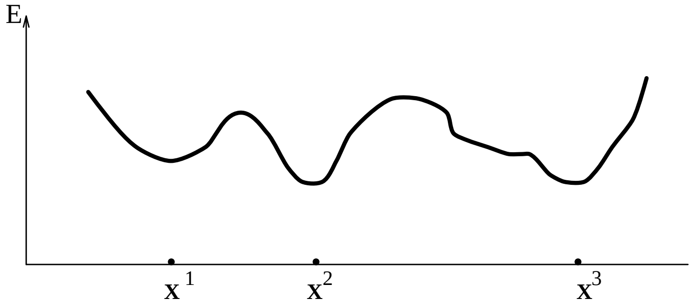

Figure 2. An energy landscape with three local stable minima representing three stored patterns.

In order that the stored key patterns become the stable local minima of E, the thresholds T_i and the couplings J_{ij} have to be suitably determined. Hopfield put $T_i = 0$ and $J_{ii} = 0$ for all $i = 1, \ldots, N$, and, for $i \neq j$, used a Hebbian prescription

$$J_{ij} = \frac{1}{N} \sum_{m=1}^{M} X_i^m X_j^m$$

where $\mathbf{X}^m$, $m = 1, \ldots M$, are the patterns to be stored. It can then be easily shown that $\mathbf{X}^m$ is near a stable local minimum of E. This implies that if we start with a configuration of neuron states which is close to a stored pattern $\mathbf{X}^m$, then the network follows a "down the hill" dynamics and ends up with the stable pattern $\mathbf{X}^m$. We can, therefore, say that the pattern $\mathbf{X}^m$ has been embedded in the memory of the net.

The Hopfield breakthrough was an important landmark in the theory and architecture of neural nets. However, the Hopfield model suffered from a serious shortcoming, namely that of *catastrophic forgetting*. As long as the important parameter $\alpha = M/N$, where M is the number of stored patterns and N is the number of neurons in the net, remains less than the critical value $\alpha_c \approx 0.14$, a Hopfield network works quite well and practically all stored patterns are remembered. But for $\alpha > \alpha_c$ the network goes into a state of total confusion and only a negligible amount of patterns is remembered. Technically, in this model, the couplings J_{ij} become arbitrarily large in magnitude. This is obviously very unsatisfactory. Any memory which has been well constructed should not go into a state of total confusion when overloaded; instead old patterns should be forgotten in order to leave room for new inputs.

The task, therefore, is to design networks which forget when overloaded. An attractive method to construct such *forgetful neural nets* is to model learning through a local, *iterative* procedure, which makes sure that the couplings J_{ij} remain bounded. This was proposed first by Hopfield himself [13]. If $J_{ij}^{(m)}$ represents the stored information up to and including pattern $\mathbf{X}^m$, then one stipulates that

$$J_{ij}^{(m)} = \frac{1}{N}\phi(\epsilon_N X_i^m X_j^m + N J_{ij}^{(m-1)}) \tag{9}$$

where $\phi : R \to R$ is a suitable continuous function, $J_{ij}^{(0)} = 0$ and $\mathbf{X}^m$, $m = 1, \ldots, M$, are the patterns to be stored. Different choices for ϕ give us different models. The original Hopfield model [12] can, in fact, be recovered by putting

$$\phi(x) = x \tag{10}$$

and $\epsilon_N = 1$. In [13], Hopfield chose

$$\phi(x) = \begin{cases} x & \text{if } |x| \leq 1 \\ \frac{x}{|x|} & \text{if } |x| > 1 \end{cases} \tag{11}$$

and $\epsilon_N = k\sqrt{N}$ (k a constant). Nadal *et al* [18] and Mézard *et al* [17] have examined the so-called *marginalist* case

$$\phi(x) = \lambda_N x \tag{12}$$

with $\lambda_N = 1 - kN^{-1}$ and $\epsilon_N = 1$. Hemmen *et al* [10] have considered the so-called *smooth* case when ϕ is an odd function ($\phi(-x) = -\phi(x)$), strictly concave ($\phi''(x) < 0$) for $x > 0$ and $\phi'(0) = 1$. The prototype of this general class is given by taking

$$\phi(x) = \tanh(x) \tag{13}$$

with $\epsilon_N = kN^{-1}$. For the sake of exposition, it is convenient to work with this prototype of the smooth model; all our results can be extended to the general case. Note that all the other models, including the marginalist (12), can be considered, in whole or in part and up to a scalar factor, as a limit of the smooth case, as $\phi''(x) \to 0$.

These different models give rise to different distributions of the couplings and to different storage capacities.

5 Probabilistic power domain and forgetful nets

The iterative equation (9) can be used to study the distribution of the couplings J_{ij}. This was first investigated by Behn *et al.* [2].

Consider equation (9) for a fixed bond $j \to i$, and assume that the key patterns $\mathbf{X}^m$ are independent, random and uncorrelated. This means that the variables X_i^m, $m = 1, \ldots, M$ are independent identically distributed random variables which take values ± 1 with equal probabilities $1/2$. Similarly, for X_j^m, $m = 1, \ldots, M$. It follows that the variables $h_m = X_i^m X_j^m$, for X_i^m, $m = 1, \ldots, M$, are also independent identically distributed random variables which take values ± 1 with equal probabilities $1/2$. Putting $x_m = NJ_{ij}^{(m)}$ and, for convenience, $\epsilon = \epsilon_N$, equation (9) reduces to the stochastic equation

$$x_{m+1} = \phi(x_m + \epsilon h_m), \tag{14}$$

with $x_0 = 0$. The choices of (10) and (11) give rise to simple distributions for x_m. In fact, it can be easily seen that (10) leads to a random walk on the set of integers, and (11) leads to a finite-state Markov partition.

Behn *et al.* studied the dynamics of the stochastic equation (14) in the marginalist (12) and smooth (13) cases. Let $\phi_\pm : R \to R$ be given by $\phi_\pm(x) = \phi(x \pm \epsilon)$. Then, in both cases of (12) and (13), ϕ_+ and ϕ_- each have a unique fixed point $x_\pm$ and the dynamics is confined in $[x_-, x_+]$. Using the Perron-Frobenius operator [22] and computer simulation, they inferred that the stochastic equation (14) has a limiting distribution as $M \to \infty$, which has a multi-fractal structure.

Using domain theory, we now present a general framework for the stochastic equation (14), by studying the dynamics of the IFS with probabilities $([x_-, x_+]; \phi_+, \phi_-; 1/2, 1/2)$ on the domain $\mathbf{P}^1\mathbf{U}[x_-, x_+]$. For the marginalist and smooth models, this domain theoretic setting provides a simple constructive analysis to show the existence and uniqueness of the limiting distribution and to examine its properties. Furthermore, it enables us to compute the expected value of physical quantities with respect to this distribution. We can in particular compute the decay of the embedding strength of the stored patterns. This quantity is used in computing the storage capacity of the network.

Note that the IFS with probabilities

$$([x_-, x_+]; \phi_+, \phi_-; 1/2, 1/2)$$

captures the stochastic equation (14) *without* the specification of the initial value $x_0 = 0$. We show first that in both the marginalist and smooth case the IFS has a unique invariant measure, and we will deduce some of their properties. We will then show that these invariant measures are in fact the limiting distributions of the stochastic equation (14). For convenience, we put $I = [x_-, x_+]$.

In the case of (12), we have

$$\phi_\pm(x) = \lambda_N(x \pm 1), \qquad\qquad x_\pm = \pm\lambda/(1-\lambda),$$

and the IFS is hyperbolic with contractivity $\lambda_N = 1 - kN^{-1}$. Furthermore, for $\lambda_N < 1/2$, the two sets $\phi_+ I$ and $\phi_- I$ have empty intersection and hence, as in Theorem 3.2, the support of the invariant measure is a Cantor set; whereas, for $\lambda_N \geq 1/2$, the union of the above two sets is I and, therefore, the support is this whole interval.

In the smooth case of (13), $\phi_\pm(x) = \tanh(x \pm \epsilon)$ as depicted in Figure 3.

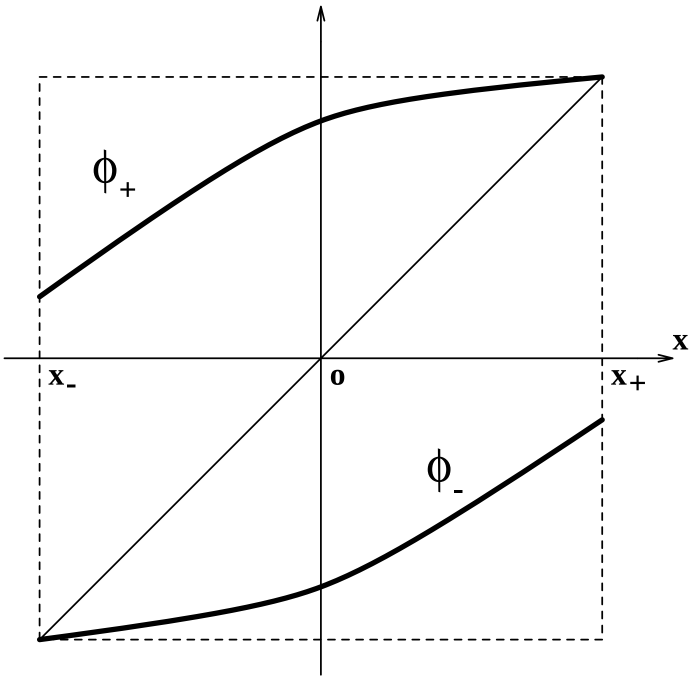

Figure 3. The graphs of $\phi_\pm = \tanh(x \pm \epsilon)$ with $\epsilon > \epsilon_0 \approx 0.957$.

For $\epsilon > \epsilon_0$ where $\epsilon_0 \approx 0.957$ is the root of the equation $\epsilon = \tanh 2\epsilon$, the IFS is hyperbolic (since $\phi'_\pm(x) < 1$ for all $x \in I$) and satisfies $\phi_+ I \cap \phi_- I = \emptyset$. Therefore, for $\epsilon > \epsilon_0$, the support of the invariant measure is a Cantor set. For $0 < \epsilon < \epsilon_0$, the IFS is not hyperbolic (since each $\phi'_\pm$ takes value 1 when $\phi_\pm$ vanishes), but it can be shown to satisfy the condition of Proposition 2.4; therefore, a unique invariant measure exists; furthermore, $\phi_+ I \cup \phi_- I = I$, which implies that the support of the invariant measure is the whole interval.

Note that in all cases the invariant measure μ^* is given constructively by the unique fixed point of the transition operator

$$\begin{array}{rcl} T : \mathbf{P}^1\mathbf{U}I \to & & \mathbf{P}^1\mathbf{U}I \\ \mu & \mapsto & \frac{1}{2}\mu \circ \phi_-^{-1} + \frac{1}{2}\mu \circ \phi_+^{-1}. \end{array}$$

In other words, $\mu^* = \bigsqcup_{n \geq 0} T^n \delta_I$ where, for $n \geq 1$,

$$T^n \delta_I = \frac{1}{2^n} \sum_{i_1, i_2, \ldots, i_n = \pm} \delta_{\phi_{i_1}\phi_{i_2}\ldots\phi_{i_n} I}.$$

Using Elton's Theorem [8], we can now prove the following.

Theorem 5.1 *The limiting distribution of the couplings of the marginalist and smooth models are given by the invariant measures of the corresponding IFSs with probabilities, the unique fixed points of the transition operators on the domain* $\mathbf{P}^1\mathbf{U}I$.

As an interesting application, one can use the above result to investigate the local behaviour of the limiting distributions, which is of basic theoretical as well as practical interest [2]. In fact, for any open subset $a \subseteq I$, we have

$$\mu^*(a) = \sup_{n \geq 0} (T^n \delta_I)(\Box a) = \sup_{n \geq 0} \sum_{\phi_{i_1}\phi_{i_2}\ldots\phi_{i_n} I \subseteq a} \frac{1}{2^n}.$$

In this paper, however, we will use Equations (6) and (7) to compute the expected value of any physical quantity with respect to the distribution of the couplings. An important quantity is the decay rate of each stored pattern as new patterns are learned; this is used to compute the storage capacity of the net [2] and is defined as follows. If patterns $\mathbf{X}^m$, $m = 1, \ldots, M$, have been learned and encoded in the couplings $J_{ij} = J_{ij}^{(M)}$ $(i, j = 1, \ldots, N)$ of a net, then the *embedding strength* [9] of pattern $\mathbf{X}^m$ is given by

$$e_m = \frac{1}{N} \sum_{i,j=1}^{N} J_{ij} X_i^m X_j^m.$$

In forgetful nets the embedding strength of a pattern depends on its *storage ancestry*. That is, if $n = M - m$, then $e_m = e_{M-n} = e(n)$. The decay rate of $e(n)$ with n is exponentially fast and is given by, $e(n) \sim \exp(\gamma n)$ as $n \to \infty$ [2], where γ is the *average Lyapunov exponent* of the stochastic mapping (14), and is given by,

$$\gamma = \int \frac{1}{2}(\ln \phi'_+(x) + \ln \phi'_-(x))\, d\mu^*(x).$$

For the marginalist model, one immediately finds that $\gamma = \ln \lambda_N$ [2]. For the smooth case,

$$\gamma = -\int_I \ln \cosh(x+\epsilon)\cosh(x-\epsilon)\, d\mu^*(x),$$

which can be accurately computed by the generalised Riemann integral as compared with the approximation obtained in [2]. First note that the function $g : [0,1] \to R$ with

$$g(x) = -\ln \cosh(x+\epsilon)\cosh(x-\epsilon)$$

is an analytic function in I with

$$|g(x) - g(y)| \leq c|x-y|$$

for $c = \max_{x_- \leq x \leq x_+} |g'(x)|$. For $\epsilon > \epsilon_0$, we know that the associated IFS is hyperbolic, and therefore Equation (7), with $k = 1$ and $s = 1/2$, gives us a formula to compute the integral up to any threshold $\theta > 0$.

For $0 < \epsilon \leq \epsilon_0$, when the associated IFS is not hyperbolic, we have an algorithm to compute the integral up to $\theta > 0$ accuracy, although we have not yet analysed its complexity. We will describe the algorithm and show its soundness. Since $\mu^* = \bigsqcup_{n \geq 0} T^n \delta_I \in \mathbf{S}^1 I$, by Proposition 2.2, with $\alpha = \theta/2$ and $\beta = \theta/2c$, there exists $n \geq 0$ such that

$$\frac{k}{2^n} < \theta/2 \tag{15}$$

where k is the number of lists $\langle i_1, i_2, \ldots, i_n \rangle$ (where $i_1, \ldots, i_n = \pm$) of length n such that the diameter of the set $\phi_{i_1}\phi_{i_2}\ldots\phi_{i_n} I$ is at least $\theta/2c$.

The generalised Riemann sum

$$S(g, T^n\delta_I) = \frac{1}{2^n} \sum_{i_1,\ldots,i_n = \pm} g(\phi_{i_1}\phi_{i_2}\ldots\phi_{i_n} 0)$$

and the integral are both sandwiched between the corresponding lower and upper sums S^ℓ and S^u and we have

$$S^u - S^\ell =$$

$$\frac{1}{2^n}\sum_{i_1,\ldots,i_n=\pm}(\sup g(\phi_{i_1}\ldots\phi_{i_n}I) - \inf g(\phi_{i_1}\ldots\phi_{i_n}I)) =$$

$$\frac{1}{2^n}\sum_1(\sup g(\phi_{i_1}\ldots\phi_{i_n}I) - \inf g(\phi_{i_1}\ldots\phi_{i_n}I))+$$

$$\frac{1}{2^n}\sum_2(\sup g(\phi_{i_1}\ldots\phi_{i_n}I) - \inf g(\phi_{i_1}\ldots\phi_{i_n}I))$$

where the first summation is over those subsets whose diameters are less than $\theta/2c$ and the second summation is over the rest of the subsets. By Equation (15), the first term is $< \theta/2$, since $0 < g(x) \leq 1$ for all $x \in R$. By the Lipschitz condition, the second term is also $< \theta/2$, since the sum of the coefficients $1/2^n$ is at most 1. Hence,

$$|S(g, T^n\delta_I) - \int g\, d\mu^*| < \theta.$$

The algorithm, therefore, finds the least $n \geq 0$ such that the nth level of the IFS tree, equivalently $T^n\delta_I$, satisfies Equation (15). Note that since ϕ_- and ϕ_+ are both monotone, the diameter of $\phi_{i_1}\ldots\phi_{i_n}I$ is simply

$$\phi_{i_1}\ldots\phi_{i_n}(x_+) - \phi_{i_1}\ldots\phi_{i_n}(x_-).$$

The algorithm then computes $S(g, T^n\delta_I)$.

Acknowledgement

I would like to thank U. Behn for inviting me to the Physics Department at Leipzig University in the Summer of 1994 and for a number of informative discussions on forgetful neural nets.

References

[1] M. F. Barnsley. *Fractals Everywhere.* Academic Press, second edition, 1993.

[2] U. Behn, J. L. van Hemmen, R. Kühn, A. Lange, and V. A. Zagrebnov. Multifractality in forgetful memories. *Physica D*, 68:401–415, 1993.

[3] A. Edalat. Dynamical systems, measures and fractals via domain theory (Extended abstract). In G. L. Burn, S. J. Gay, and M. D. Ryan, editors, *Theory and Formal Methods 1993*. Springer-Verlag, 1993. Full paper to appear in *Information and Computation*.

[4] A. Edalat. Domain of computation of a random field in statistical physics. In Hankin *et al*, editor, *Proceedings of the Second Imperial College, Department of Computing, Theory and Formal Methods Workshop*. 1994.

[5] A. Edalat. Domain theory and integration (Extended abstract). In *Logic in Computer Science*. IEEE Computer Society Press, 1994. Ninth Annual IEEE Symposium, July 3-7, 1994, Paris, France. Full paper to appear in *Theoretical Computer Science*.

[6] A. Edalat. Power domains and iterated function systems. Technical Report Doc 94/13, Department of Computing, Imperial College, 1994. Submitted to *Information and Computation*.

[7] A. Edalat. Domain theory in stochastic processes. In *Proceedings of Logic in Computer Science, Tenth Annual IEEE Symposium, 26-19 June, 1995, San Diego*. IEEE Computer Society Press, 1995. To appear.

[8] J. Elton. An ergodic theorem for iterated maps. *Journal of Ergodic Theory and Dynamical Systems*, 7:481–487, 1987.

[9] J. L. van Hemmen, G. Keller, A. Huber, and R. Kühn. Nonlinear neural networks. *Journal of Statistical Physics*, 50(1/2):231–293, 1988.

[10] J. L. van Hemmen, G. Keller, and R. Kühn. Forgetful memories. *Europhysics Letters*, 5(7):663–668, 1988.

[11] D. Hepting, P. Prusinkiewicz, and D. Saupe. Rendering methods for iterated function systems. In *Proceedings IFIP Fractals 90*. 1991.

[12] J. J. Hopfield. In *Proceedings of the National Academy of Science, USA*, volume 79, page 2554. 1982.

[13] J. J. Hopfield. Neurons with graded response have collective computational properties like those of two-state neurons. In *Proceedings of the National Academy of Science, USA*, volume 81, pages 3088–3092. 1984.

[14] C. Jones and G. Plotkin. A Probabilistic Powerdomain of Evaluations. In *Logic in Computer Science*, pages 186–195. IEEE Computer Society Press, 1989.

[15] S. Karlin. Some random walks arising in learning models. *Pacific Journal of Mathematics*, 3:725–756, 1953.

[16] S. Lakshmivarahan. *Learning algorithms theory and applications*. Springer, 1981.

[17] M. Mézard, J. P. Nadal, and G. Toulouse. Solvable models of working memories. *J. Physique*, 47:1457–1462, 1986.

[18] J. P. Nadal, G. Toulouse, J. P. Changeux, and S. Dehaene. Networks of formal neurons and memory palimpsests. *Europhysics letters*, 1(10):535–542, 1986.

[19] M. F. Norman. *Markov processes and learning models.* Academic Press, 1972.

[20] G. D. Plotkin. A powerdomain for countable non-determinism. In M. Nielsen and E. M. Schmidt, editors, *Automata, Languages and programming*, pages 412–428, Berlin, 1982. EATCS, Springer-Verlag. Lecture Notes in Computer Science Vol. 140.

[21] N. Saheb-Djahromi. Cpo's of measures for non-determinism. *Theoretical Computer Science*, 12(1):19–37, 1980.

[22] A. N. Shiryayev. *Probability.* Graduate texts in mathematics. Springer, 1984.