Learning Automata-Based Solutions to the Nonlinear Fractional Knapsack Problem With Applications to Optimal Resource Allocation

Ole-Christoffer Granmo, B. John Oommen, *Fellow, IEEE*, Svein Arild Myrer, and Morten Goodwin Olsen

***Abstract*—This paper considers the nonlinear fractional knapsack problem and demonstrates how its solution can be effectively applied to two resource allocation problems dealing with the World Wide Web. The novel solution involves a "team" of deterministic learning automata (LA). The first real-life problem relates to resource allocation in web monitoring so as to "optimize" information discovery when the polling capacity is constrained. The disadvantages of the currently reported solutions are explained in this paper. The second problem concerns allocating limited sampling resources in a "real-time" manner with the purpose of estimating multiple binomial proportions. This is the scenario encountered when the user has to evaluate multiple web sites by accessing a limited number of web pages, and the proportions of interest are the fraction of each web site that is successfully validated by an HTML validator. Using the general LA paradigm to tackle both of the real-life problems, the proposed scheme improves a current solution in an online manner through a series of informed guesses that move toward the optimal solution. At the heart of the scheme, a team of deterministic LA performs a controlled random walk on a discretized solution space. Comprehensive experimental results demonstrate that the discretization resolution determines the precision of the scheme, and that for a given precision, the current solution (to both problems) is consistently improved until a nearly optimal solution is found—even for switching environments. Thus, the scheme, while being novel to the entire field of LA, also efficiently handles a class of resource allocation problems previously not addressed in the literature.**

***Index Terms*—Game playing, learning automata (LA), nonlinear knapsack problem, resource allocation, stochastic optimization.**

I. Introduction

A. Motivation of Application-Domain Problems

Dealing with the web is particularly fascinating because it leads to numerous extremely interesting problems involving real-life resource allocation and scheduling. Such problems are even more intriguing when access to resources is constrained and the parameters of the underlying system are unknown and not easily estimated. One such classic problem is that which involves "optimal search," where a user has to determine the amount of time a particular "site" (which may be a web-site, a library, or in general a geographical area) is to be searched when the probability of locating the object searched for in that site cannot be estimated, e.g., because the underlying event is unobservable [20].

The first problem that we study in this paper involves another such captivating problem, namely, the problem of "optimizing" the allocation of polling resources in web page monitoring[1] when the polling capacity is restricted [12], [18]. Consider a basic web monitoring resource allocation system. Such a system may involve n web pages that are updated periodically, although with different periods. Clearly, each web page can be polled with a maximum frequency, which essentially involves polling every single web page at "every" time step. Obviously, this would effectively "flood" the system with "status-probing" polls, as opposed to information-delivering searches. We consider the problem of determining the web page polling frequencies that the system must use, so as to maximize the number of web page updates detected, without exceeding the available monitoring capacity.

The above problem has been previously modeled as a knapsack problem.[2] This has been achieved by making the polling frequencies correspond to material amounts and the restricted polling capacity correspond to a knapsack of fixed volume. Typically, the parameters of such knapsack problems are "estimated," e.g., based on a tracking phase where the web pages are polled mainly for estimation purposes [12], [18]. One major disadvantage of the latter approach is that the parameter estimation phase significantly delays the implementation of the optimal solution. This disadvantage is further aggravated in "dynamic" environments where the optimal solution changes over time, introducing the need for parameter reestimation.

The second problem that we study in this paper is the problem of "minimizing the variance" while estimating[3] "multiple binomial proportions" when the sampling resources to be

Manuscript received January 9, 2006; revised March 29, 2006. Preliminary versions of some of the results of this paper appear in the Proceedings of CIS-RAM 2006 and in the Proceedings of IIS:IIPW'06. This paper was recommended by Associate Editor R. Lynch.

O.-C. Granmo and M. G. Olsen are with the Department of Information and Communication Technology, Agder University College, 4876 Grimstad, Norway (e-mail: ole.granmo@hia.no).

B. J. Oommen is with the School of Computer Science, Carleton University, Ottawa, ON K1S 5B6, Canada, and also with Agder University College, 4876 Grimstad, Norway (e-mail: oommen@scs.carleton.ca).

S. A. Myrer is with Bouvet, 0403 Oslo, Norway.

Digital Object Identifier 10.1109/TSMCB.2006.879012

[1]Web page monitoring consists of repeatedly polling (i.e., observing) a selection of web pages so that any changes or updates that may occur over time are detected.

[2]The fractional knapsack problem concerns n materials each characterized by its "value per unit volume." The problem involves filling a knapsack of fixed volume with a mixture of materials so as to attain a maximal value [1].

[3]If we are only interested in ranking the proportions, we do not actually need to estimate them (for example, using a maximum likelihood or Bayesian computation). Rather, we could just as well attempt to "learn" their comparative values, for example, using a random race [9].

allocated are limited. This is, for instance, the scenario encountered in web monitoring when "evaluating" multiple web sites by accessing a limited number of web pages, and the proportions of interest are the fraction of each web site that is successfully validated by an HTML validator [14] and/or a web content accessibility guidelines (WCAG)-based validator [19]. In the latter problem, the web site sample sizes correspond to material amounts, and the restricted combined sample size corresponds to the fixed knapsack volume.

Again, the problem becomes intriguing when the underlying distributions are unknown, which is often the case when dealing with the world wide web. Indeed, dealing with the web introduces further complexity to the problem in focus because web data are becoming increasingly dynamic and are possibly relevant for only a limited period of time as it either rapidly changes, gets replaced, or becomes part of the deep web.[4]

In this paper, we propose a "single" generic solution applicable to both of the above problems, and also, hopefully, to a number of similar problems. The solution involves modeling the problem as a fractional knapsack (FK) problem and using a "team" of learning automata (LA) to solve it.

B. Variants of the Knapsack Problem

Several variants of the knapsack problem have been studied in the literature [7]. In the interest of posing our problem in the right perspective, we "briefly" catalogue them below.

Linear FK Problem: The linear FK problem is a classical continuous optimization problem that also has applications within the field of resource allocation. The problem involves n materials of different value v_i per unit volume, $1 \leq i \leq n$. Each material is available in a certain amount $x_i \leq b_i$. Let $f_i(x_i)$ denote the value of the amount x_i of material i, i.e., $f_i(x_i) = v_i x_i$. The problem is to fill a knapsack of fixed volume c with the material mix $\vec{x} = [x_1, \ldots, x_n]$ of maximal value $\sum_1^n f_i(x_i)$ [1].

Nonlinear Equality FK (NEFK) Problem: One important extension of the above classical problem is the NEFK problem with a separable and concave objective function. The problem can be stated as [7]

$$\text{maximize} \sum_1^n f_i(x_i)$$

$$\text{subject to} \sum_1^n x_i = c$$

$$0 \leq x_i, \qquad i = 1, \ldots, n.$$

Since the objective function is considered to be concave, the value function $f_i(x_i)$ of each material is also concave. This means that the derivatives of the material value functions $f_i(x_i)$ with respect to x_i (hereafter denoted as f_i') are nonincreasing. In other words, the material value "per unit volume" is no longer constant as in the linear case, but decreases with the material amount. The latter problem characteristics have led to efficient solution methods based on the principle of Lagrange multipliers. In short, the optimal value occurs when the derivatives f_i' of the material value functions are equal, subject to the knapsack constraints [3], [6]

$$f_1'(x_1) = \cdots = f_n'(x_n)$$

$$\sum_1^n x_i = c$$

$$0 \leq x_i, \qquad i = 1, \ldots, n.$$

Stochastic NEFK Problem: In this paper, we generalize the above nonlinear equality knapsack problem. First, we let the material value per unit volume for any x_i be a "stochastic" function $F_i'(x_i)$, where the function[5] $F_j'(x_j)$ need not be related in any way to $F_k'(x_k)$ if $j \neq k$. Furthermore, we consider the distribution of $F_i'(x_i)$ to be "unknown." That is, each time an amount x_i of material i is placed in the knapsack, we are only allowed to observe an instantiation of $F_i'(x_i)$ at x_i, and not $F_i'(x_i)$ itself. Given this stochastic environment, we intend to devise an online incremental scheme that learns the mix of materials of maximal "expected" value through a series of informed guesses.

Stochastic Knapsack Problems—State-of-the-Art: To the best of our knowledge, our targeted stochastic NEFK problem has not been addressed in the literature before. However, one does find several studies on related problems. For example, the works in [5] and [15] consider solution policies for stochastic generalizations of the so-called NP-hard "linear" integer knapsack problem. Value distributions are considered known and constant, which makes dynamic programming a viable solution. Another variant of the knapsack problem is found in [13], where a deterministic knapsack is used, however, with objects arriving to and departing from the knapsack at random times. The optimization problem considered is to accept/block arriving objects so that the average value of the knapsack is maximized.

Contributions of This Paper: We base our work on the principles of LA [8], [16]. LAs have been used to model biological systems [17] and have attracted considerable interest in the last decade because they can learn optimal actions when operating in (or interacting with) unknown stochastic environments. Furthermore, they combine rapid and accurate convergence with low computational complexity. Recently, Oommen [10] proposed an LA-based mechanism for locating the optimal point on a line. An environment tells the learning mechanism whether its point guesses lie to the left or to the right of the optimal point, possibly erroneously, and by interacting with the environment, the learning mechanism converges toward selecting the optimal point with arbitrary accuracy, i.e., the scheme is ϵ optimal. While the latter work considers stochastic optimization of a single parameter, we target optimization of n parameters (n material amounts), where the different parameters are mutually

[4]Data from the deep web are dynamically produced in response to a direct request, usually involving a searchable database.

[5]Typically, in stochastic processes, the symbol F is used to represent a distribution, and f is used to represent the corresponding density. This is not the case in our current notation.

constrained (the knapsack). Furthermore, our environment is less informative compared to the environment in [10] because our environment does not tell us whether a parameter guess x_i lies to the left or to the right of an optimal value. The environment only replies with an instantiated unit volume value at x_i drawn from the stochastic unit volume value function $F_i'(x_i)$. The solution to this stochastic generalization is then used to solve the two resource allocation problems from web monitoring described earlier.

Thus, in all brevity, to the best of our knowledge, the salient contributions of this work are as follows.

1) We present the first reported solution to the stochastic NEFK problem in which the material values per unit volume are "stochastic" with "unknown" distributions.
2) We present the first reported solution to the stochastic NEFK problem that uses a team of LA operating in a multidimensional environment.
3) Unlike the work in [10], we do not require the "Oracle" to inform us of whether the current parameter guess x_i lies to the left or to the right of the optimal value.
4) We present the first reported solutions to the polling frequency determination problem and to the so-called sample size determination problem when modeled in unknown stochastic settings.

C. Organization of the Paper

This paper is organized as follows. In Section II, we present a "game" of LA that is designed to solve our stochastic version of the NEFK problem incrementally and online. In Section III, we detail the polling frequency determination problem and present a solution to it based on the LA knapsack game (LAKG) scheme from Section II. The solution to the variance minimization problem is then presented in Section IV. For both of the proof-of-concept optimization problems, we evaluate the LAKG scheme empirically based on simulating stationary and nonstationary environments with as many as 500 materials. We conclude this paper in Section V and offer prospects for further work.

II. Nonlinear Knapsack Game of Finite Automata

A. Overview of the LAKG Solution

In order to put our work in the right perspective, we start this section by briefly discussing solution concepts for more basic variants of the knapsack problem. As indicated in Section I, solving the classical linear FK problem involves finding the most valuable mix $\vec{x} = [x_1, \ldots, x_n]$ of n materials that fits within a knapsack of fixed capacity c. Each material is available in a certain amount $x_i \leq b_i$. The value of the amount x_i of material i, $f_i(x_i) = v_i x_i$, is linear with respect to x_i, and accordingly, the material value per unit volume is constant, i.e., $f_i'(x_i) = v_i$. Because a fraction of each material can be placed in the knapsack, the following greedy algorithm from [1] finds the most valuable mix: "Take as much as possible of the material that is most valuable per unit volume. If there is still room, take as much as possible of the next most valuable material. Continue until the knapsack is full."

Let us now assume that the material unit volume values are "random" variables $\{V_1, \ldots, V_n\}$ with "time-invariant" and "known" distributions. Furthermore, for the sake of conceptual clarity, let us only consider binary variables V_i that instantiate to the value of either 0 or 1. Let p_i^0 denote the probability that $V_i = 0$ and $p_i^1 = 1 - p_i^0$ denote the probability that $V_i = 1$. Then the value $F_i(x_i)$ of the amount x_i of material i also becomes stochastic, i.e., $F_i(x_i) = V_i \times x_i$. In other words, $F_i(x_i)$ takes the value 0 with probability p_i^0 and the value x_i with probability p_i^1. Under such conditions, the above greedy strategy can be used to maximize the "expected" value of the knapsack $\sum_1^n E[F_i(x_i)]$ by simply selecting the material based on the expected unit volume values $E[V_i] = p_i^1$ rather than actual unit volume values.

However, the above-indicated solution approach is obviously inadequate when the distributions of $F_i(x_i)$ and V_i are unknown. The problem becomes even more challenging when the distribution of the material value per unit volume is no longer constant but rather depends on the amount x_i. Let $F_i'(x_i)$ denote the stochastic function that determines the unit volume value of material i given material amount x_i. To elaborate, $F_i'(x_i)$ takes the value 0 with probability $p_i^0(x_i)$ and the value 1 with probability $p_i^1(x_i)$. Under the latter conditions, our aim is to find a scheme that moves toward optimizing the following NEFK problem online

$$\text{maximize} \sum_1^n E\left[F_i(x_i)\right]$$
$$\text{subject to} \sum_1^n x_i = c$$
$$0 \leq x_i, \qquad i = 1, \ldots, n.$$

Note that we allow only instantiations of the material value per unit volume $F_i'(x_i)$ to be observed. That is, each time an amount x_i of material i is placed in the knapsack, an instantiation of $F_i'(x_i)$ at x_i is observed. We shall assume that x_i is any number in the interval (0, 1]. The question of generalizing this will be considered later.

The crucial issue that we have to address is that of determining how to change our current guesses on x_i, $1 \leq i \leq n$. We shall attempt to do this in a discretized manner by subdividing the unit interval into N points $\{(1^\lambda/N^\lambda), (2^\lambda/N^\lambda), \ldots, (N-1)^\lambda/N^\lambda, 1\}$, where N is the resolution of the learning scheme, and $\lambda > 0$ determines the linearity of the discretized solution space.[6] We will see that a larger value of N will ultimately imply a more accurate solution to the knapsack problem.

At this juncture, it is pertinent to mention that although the rationale for this updating is the stochastic point location solution proposed in [10], the two schemes are quite distinct for the following reasons.

1) The scheme in [10] is linear.
2) The method proposed in [10] assumes the existence of an Oracle that informs the LA whether to go "right" or "left."

[6]The importance of this parameter will become evident in Sections III and IV, where empirical results are presented.

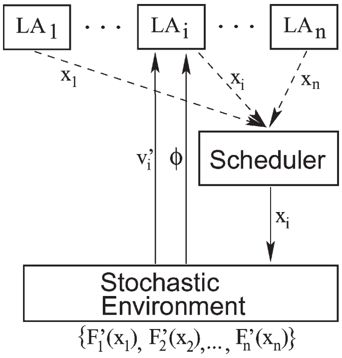

Fig. 1. "Team" of LA interacting with a Scheduler and an unknown stochastic environment.

In our application domain, this has now to be "inferred" by the system.

3) The method proposed in [10] assumes that there is only a single LA in the picture. Here, we specifically understand that there are multiple LAs, each of them being constrained to work together with the others.[7]

4) As opposed to the scheme in [10], our present approach is also applicable to dynamic (nonstationary) environments.

B. Details of the LAKG Solution

At the heart of our scheme is a "game" between n finite automata that interact with a Scheduler and a stochastic environment. Fig. 1 provides an overview of this interaction and is explained below.

1) Stochastic Environment: The environment consists of a set of stochastic material unit volume value functions $\mathcal{F}' = \{F'_1(x_1), F'_2(x_2), \ldots, F'_n(x_n)\}$. If amount x_i of material i is suggested to the environment, $F'_i(x_i)$ takes the unit volume value $v'_i = 0$ with probability $p_i^0(x_i)$ and the unit volume value $v'_i = 1$ with probability $p_i^1(x_i)$. In addition, the environment provides a signal ϕ that indicates whether the knapsack is full, i.e.,

$$\phi = \begin{cases} \text{true}, & \text{if } \sum_{i=1}^{n} x_i \geq c \\ \text{false}, & \text{otherwise.} \end{cases}$$

2) Scheduler: The scheduler takes material amounts $\vec{x} = [x_1, \ldots, x_n]$ as its input. The purpose of the Scheduler is 1) to render accesses to the stochastic environment sequential, and 2) to make sure that the unit volume value functions $F'_1(x_1), F'_2(x_2), \ldots, F'_n(x_n)$ are accessed with frequencies proportional to $\vec{x}$.

The reader should note that our scheme does not rely on accessing the unit volume value functions sequentially with frequencies proportional to $\vec{x}$ for solving the knapsack problem. However, this restriction is obviously essential for solving the problem "incrementally" and "online" (or rather in a "real-time" manner). For the sake of simplicity, we choose to access the functions randomly by sampling them from a probability distribution proportional to $\vec{x}$.

3) Team of LA: Each material i is assigned a finite fixed structure automaton LA_i with states $1, \ldots, N$. Let the current state of automaton LA_i be $s_i(t)$. When the automaton acts, it suggests the amount $x_i(t) = (s_i(t)^\lambda / N^\lambda)$ of material i to the Scheduler, which in turn interacts with the stochastic environment. Assume that $v'_i(t)$ and $\phi(t)$ are the resulting feedbacks from the stochastic environment. Then the state of the automaton is updated as

$$\begin{aligned} s_i(t+1) &:= s_i(t) + 1, && \text{if } v'_i(t) = 1 \text{ and } 1 \leq s_i(t) < N \text{ and not } \phi(t) \\ s_i(t+1) &:= s_i(t) - 1, && \text{if } v'_i(t) = 0 \text{ and } 1 < s_i(t) \leq N \text{ and } \phi(t) \\ s_i(t+1) &:= s_i(t), && \text{otherwise.} \end{aligned}$$

Notice that although the above state-updating rules are deterministic, because the knapsack is stochastic, the state transitions will also be stochastic.

The purpose of the above LA game is to form a stochastic competition between the n automata so that the competition directs the automata toward the optimal solution. In essence, the competition is governed by two situations, namely: 1) there is still room for more material in the knapsack, and 2) the knapsack is full.

In the first situation, the competition consists of increasing the material amount guesses as quickly as possible. However, a guess x_i can only be increased when the corresponding unit volume value function $F'_i(x_i)$ instantiates to 1. Accordingly, if $p_i^1(x_i) < p_j^1(x_j)$ for materials i and j, then automaton j will have an edge in the competition. The second situation is the opposite of the first one. In the second situation, each automaton i "decreases" its material amount guess x_i whenever the corresponding unit volume value function $F'_i(x_i)$ instantiates to 0. Accordingly, if $p_i^1(x_i) < p_j^1(x_j)$ for materials i and j, then automaton i will have an edge, rather than automaton j. Operating simultaneously, the two situations of the competition are designed to stochastically move the current allocation of materials toward the knapsack capacity with the aim of approaching the optimal expected knapsack value.

Note that because each automaton LA_i acts (e.g., polls a web page) with an average frequency proportional to $x_i(t)$, the F_is are accessed with a frequency corresponding to the current knapsack solution. In other words, our scheme actually applies the "current solution" when seeking its improvement. As we will see in Sections III and IV, the latter property permits us to use the scheme incrementally and online.

III. Application I: Polling Frequency Determination

Recent approaches to resource allocation in web monitoring attempt to "optimize" the performance of the system when the monitoring capacity is restricted [12], [18]. The principle cited in the literature essentially invokes Lagrange multipliers to solve a "nonlinear equality" knapsack problem with a separable and concave objective function [7]. Thus, for example, a basic

[7]It is conceivable that this problem can be resolved with a single LA possessing an extended number of actions. But we do not recommend it for scalability reasons—the action space would grow exponentially. Furthermore, the benefit of having a decentralized decision making process would be lost (e.g., for distributed environments such as wireless sensor networks).

web monitoring resource allocation problem may involve n web pages that are updated periodically, although with different periods. Clearly, each web page can be polled with a maximum frequency, which would result in a sluggish system. The problem that we study involves determining the web page polling frequencies (i.e., how often each web page is accessed by the monitoring system) so as to maximize the number of web page updates detected. Observe that this must be achieved without exceeding the available monitoring capacity, e.g., the maximum number of web pages that can be accessed per unit of time as dictated by the governing communication bandwidth, and processing speed limitations.

The approaches that represent the state-of-the-art assume that the knapsack problem is "deterministic" and fully "known." However, from a web monitoring perspective, the web must often be seen as a stochastic and more or less unknown environment. In such cases, the parameters of the knapsack problem could be "estimated" instead, e.g., based on a tracking phase where web pages are polled mainly for estimation purposes [12], [18]. The main disadvantage of the latter approach is that the parameter estimation phase significantly delays the implementation of the optimal solution. This disadvantage is further aggravated in "dynamic" environments where the optimal solution may change over time, introducing the need for parameter reestimation.

A. Problem Background

Although several nonlinear criterion functions for measuring web monitoring performance have been proposed in the literature (e.g., see [12] and [18]), from a broader viewpoint, they are mainly built around the basic concept of "update detection probability," i.e., the probability that polling a web page results in new information being discovered. Therefore, for the purpose of clarification and for the sake of conceptual clarity, we will use the update detection probability as the token of interest in this paper. To further define our notion of web monitoring performance, we consider that time is discrete with the time interval length T to be the atomic unit of decision making. In each time interval, every single web page i has a constant probability q_i of remaining "unchanged."[8] Furthermore, when a web page is updated/changed, the update is available for detection only until the web page is updated again. After that, the original update is considered lost. For instance, each time a newspaper web page is updated, previous news items are replaced by the most recent ones.

In the following, we will denote the update detection probability of a web page i as d_i. Under the above conditions, d_i depends on the frequency x_i that the page is polled with and is modeled using the expression

$$d_i(x_i) = 1 - q_i^{\frac{1}{x_i}}.$$

By way of example, consider the scenario that a web page remains unchanged in any single time step with probability 0.5. Then polling the web page uncovers new information with probability $1 - 0.5^3 = 0.875$ if the web page is polled every third time step (i.e., with frequency 1/3) and $1 - 0.5^2 = 0.75$ if the web page is polled every second time step. As seen, increasing the polling frequency reduces the probability of discovering new information on each polling.

Given the above considerations, our aim is to find the page polling frequencies $\vec{x}$ that maximize the expected number of pollings uncovering new information per time step, i.e.,

$$\begin{aligned} \text{maximize} & \sum_1^n x_i \times d_i(x_i) \\ \text{subject to} & \sum_1^n x_i = c \\ & 0 \le x_i, \qquad i = 1, \ldots, n. \end{aligned}$$

Note that in the general web monitoring case, we are not able to observe $d_i(x_i)$ or q_i directly; polling a web page only reveals whether the web page has been updated "at least once" since our last poll.[9] As such, web monitoring forms a proof-of-concept application for resource allocation in unknown stochastic environments.

B. LAKG Solution

In order to find an LAKG solution to the above problem, we must define the stochastic environment that the LAs are to interact with. As seen in Section II, the stochastic environment consists of the stochastic unit volume value functions $\mathcal{F}' = \{F_1'(x_1), F_2'(x_2), \ldots, F_n'(x_n)\}$, which are unknown to the team of LA. We identify the nature of these functions by applying the principle of Lagrange multipliers to the above maximization problem. In short, after some simplification, it can be seen that the following conditions characterize the optimal solution:

$$\begin{aligned} & d_1(x_1) = d_2(x_2) = \cdots = d_n(x_n) \\ & \sum_1^n x_i = c \\ & 0 \le x_i, \qquad i = 1, \ldots, n. \end{aligned}$$

Since we are not able to observe $d_i(x_i)$ or q_i directly, we base our definition of $\mathcal{F}'$ on the result of polling web pages. Briefly stated, we want $F_i'(x_i)$ to instantiate to the value 0 with probability $1 - d_i(x_i)$ and to the value 1 with probability $d_i(x_i)$. Accordingly, if the web page i is polled and i has been updated since our last polling, then we consider $F_i'(x_i)$ to have been instantiated to 1. And if the web page i is unchanged, we consider $F_i'(x_i)$ to have been instantiated to 0.

[8] Note that in our empirical results, we also report high monitoring performance even with changing q_i. The high performance can be explained by the ability of our scheme to adaptation.

[9] Some web pages are also annotated with the time of last update. However, this information is not generally available/reliable [4] and is therefore ignored in our scheme.

C. Empirical Results

In this section, we evaluate our learning scheme by comparing it with four classical policies using synthetic data. We have implemented the following classical policies.

1) *Uniform:* The uniform policy allocates monitoring resources uniformly across all web pages. This is the only classical policy of the four that can be applied directly in an unknown environment.
2) *Proportional:* In the proportional policy, the allocation of monitoring resources to web pages is proportional to the update frequencies of the web pages. Accordingly, this policy requires that the web page update frequencies are known.
3) *Estimator:* The estimator policy handles unknown web update frequencies by polling web pages "uniformly" in a parameter estimation phase with the purpose of estimating update frequencies. After the parameter estimation phase, the proportional policy is applied; however, the latter is based on the estimated update frequencies rather than the true ones.
4) *Optimal:* The optimal policy requires that web page update frequencies are known, and finds the optimal solution based on the principle of Lagrange multipliers [12], [18].

To evaluate web resource allocation policies, recent research advocates Zipf-like distributions [21] to generate realistic web page update frequencies [12], [18]. The Zipf distribution can be stated as

$$Z(k; s, N) = \frac{1/k^s}{\sum_{n=1}^{N} 1/n^s}$$

where N is the number of elements, k is their rank, and s is a parameter that governs the skewedness of the distribution (e.g., for $s = 0$, the distribution is uniform).

For our experiments, web pages are considered ranked according to their update frequencies, and the update probability of a web page is calculated from its rank. We use the following function to determine the update probability of each web page:

$$q_k(\alpha, \beta) = \frac{\alpha}{k^\beta}.$$

In this case, k refers to the web page of rank k, the parameter β determines the skewedness of the distribution, and $\alpha \in [0.0, 1.0]$ represents the magnitude of the update probabilities (i.e., the web page of rank 1 is updated with probability α each time step).

Without loss of generality, we normalize the web page polling capacity in our experiments to 1.0 poll per time step, and accordingly, we vary the average total number of web page updates per time step instead.

The results of our experiments are truly conclusive and confirm the power of the LAKG. Although several experiments were conducted using various α, β, and number of automata, we report for the sake of brevity only the results for 500 web pages (the main case from [12]) within the following environments.

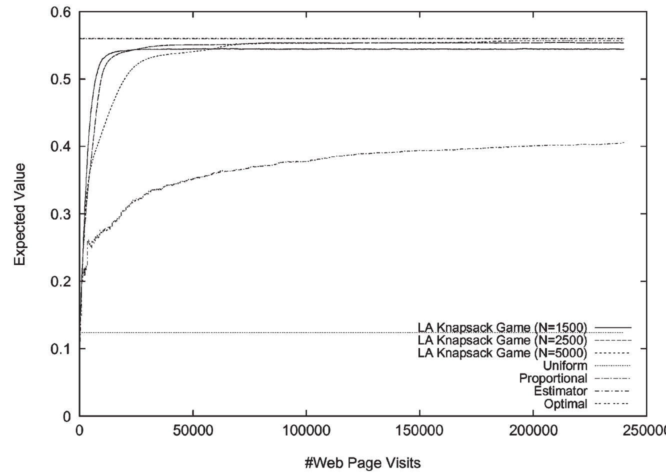

Fig. 2. In the ($\alpha = 0.3$, $\beta = 1.5$) environment, the LA knapsack scheme is superior to the estimator scheme, allowing increased convergence accuracy at the cost of convergence speed, without any parameter estimation phase.

- $\alpha = 0.3$; $\beta = 1.5$, where the average number of updates per time step is 0.76, and accordingly, below the web page polling capacity. The web page update distribution is highly skewed, as explored in [12].
- $\alpha = 0.3$; $\beta = 1.0$, where the average number of updates per time step is increased to 2.0 (twice the web page polling capacity) by making the web page update distribution less skewed (the normal Zipf distribution).
- $\alpha = 0.9$; $\beta = 1.5$, where the average number of updates is set to 2.3 by increasing the web page update probability. Because of the high values of both α and β, this environment turns out to be the most challenging one, discriminating clearly between the optimal policy and the proportional policy.

For these values, an ensemble of several independent replications with different random number streams was performed to minimize the variance of the reported results.

1) Static Environments: We see from Fig. 2 that the proportional policy and the optimal policy provide more or less the same solution—a solution superior to the uniform policy solution. We also observe that the performance of the estimator scheme increases steadily with the length of the parameter estimation phase. The figure also shows the performance of the three different LAKG configurations reported in this section, namely; 1) $[N = 1500, \gamma = 1.3]$; 2) $[N = 2500, \gamma = 1.3]$; and 3) $[N = 5000, \gamma = 1.2]$. As seen, the performance of the LAKG increases significantly quicker compared to the estimator scheme, and in contrast to the estimator scheme, the performance is improved online (in real-time manner) "without" invoking any parameter estimation phase. We also observe that, as expected, increasing the resolution N improves the convergence accuracy at the cost of convergence speed.

As seen in Fig. 3, a less skewed web page update distribution function makes the uniform policy more successful mainly because a larger number of web pages will have a significant probability of being updated. For the same reason, the estimator

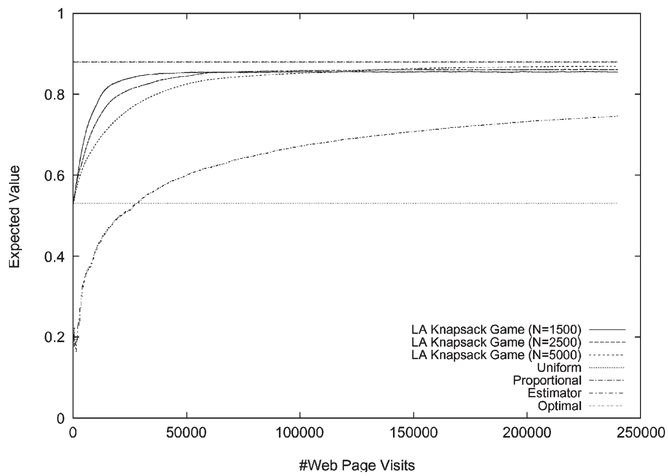

Fig. 3. In the ($\alpha = 0.3$, $\beta = 1.0$) environment, a less skewed web page update distribution makes the uniform policy as well as the estimator policy more successful mainly because of more widely and abundant updating of web pages.

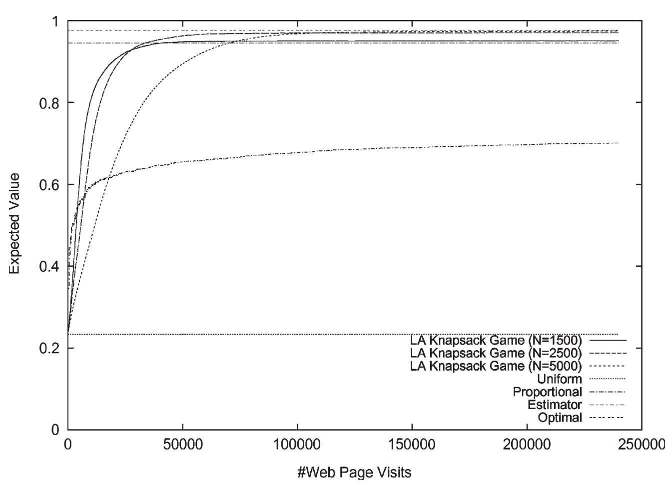

Fig. 4. In the ($\alpha = 0.9$, $\beta = 1.5$) environment, the LAKG scheme breaches the performance boundary set by the proportional policy, converging toward nearly optimal solutions.

scheme is able to lead to an improved performance quicker. In spite of this, the LAKG yields superior performance.

The most difficult class of environments we simulate is an environment with a highly skewed web page update distribution ($\beta = 1.5$) combined with a high update probability ($\alpha = 0.9$). In such an environment, the optimal policy performs significantly better than the proportional policy, and so any scheme that converges toward a proportional policy solution will not reach optimal performance. As seen in Fig. 4, the LAKG breaches the performance boundary set by the proportional policy and converges toward nearly optimal solutions.

2) Dynamic Environments: A dynamically changing environment is particularly challenging because the optimal solution is time dependent. In such cases, the current resource allocation solution should be modified according to environmental changes. When, additionally, the environment and its characteristics are unknown, any changes must first be learnt before any meaningful modification can take place.

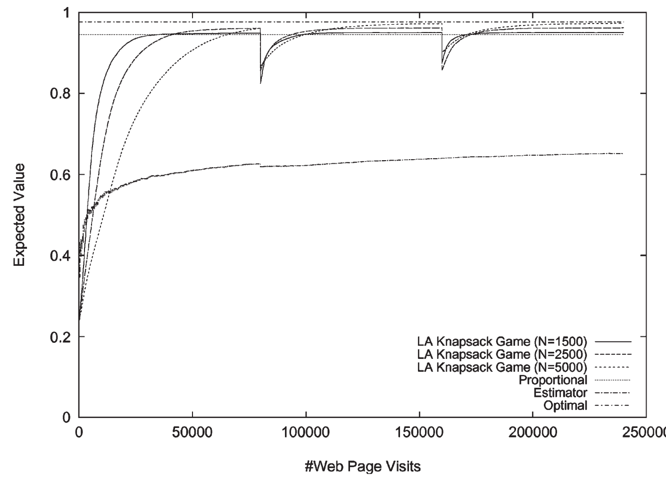

Fig. 5. In the ($\alpha = 0.9$, $\beta = 1.5$) environment where web page ranking changes at every 80 000th web page poll. LAKG finds nearly optimal solutions initially, and recovers quickly after environmental changes.

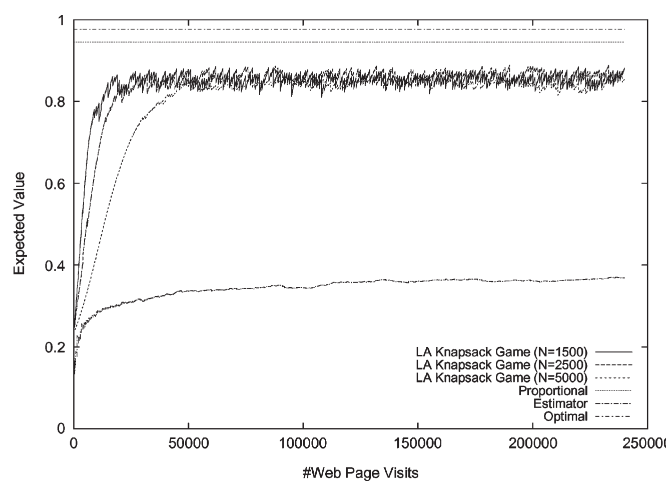

Fig. 6. In the ($\alpha = 0.9$, $\beta = 1.5$) environment where web page ranking changes every 1000th poll. Observe that the LAKG is able to steadily improve the initial solution, but is never allowed to reach an optimal solution.

In order to simulate a dynamic environment, we change the ranking of the web pages at every rth web page poll—a single web page is selected by sampling from the current Zipf distribution, and this web page switches rank with the succeeding web page in the ranking. As a result, the Zipf distribution also changes. This means that the web monitor is allowed to conduct r web page polls before the environment changes. Fig. 5 demonstrates the ability of our scheme to relearn in a switching environment for $r = 80\,000$. As seen in the figure, the LAKG quickly recovers after the environment has changed and then moves toward a new nearly optimal solution. The reader should again note how the speed of recovery depends on the resolution of the automata N.

In the previous dynamic environment, the LAKG was able to fully recover to a nearly optimal solution because of the low frequency of environmental changes. Fig. 6 demonstrates the behavior of the automata in a case when this frequency is increased to $r = 1000$. As seen, the automata still quickly and steadily improve the initial solution, but are obviously never allowed to reach an optimal solution. However, note how the quickly changing environment is not able to hinder the automata stabilizing on a solution superior to the solutions found by the estimator scheme.

Clearly, these results demonstrate how the LAKG can perform when the environment is switching with a fixed period (in this case $r = 80\,000$ and $r = 1000$). However, we believe that similar results will be obtained if r is not fixed, but changing in such a way that the scheme has enough time to learn the parameters of the updated environment.

IV. Application II: Sample Size Determination

In this section, we consider the problem of estimating the proportion of a population having some specific characteristic. Specifically, we assume that n populations are to be evaluated and that each population i is characterized by an independent unknown binomial proportion q_i. We will pursue here the goal of minimizing the variance of the proportion estimates when the total number of samples available for estimating the proportions is restricted to c. The purpose is to make the estimates as accurate as possible. For instance, let us assume that the task at hand is to determine the proportion of a web site that is successfully validated by an HTML validator [14] and/or a WCAG accessibility validator [19], and that n web sites are to be evaluated by only accessing c web pages.

A. Problem Background

Let x_i be the number of elements sampled randomly from population i, and let the count Y_i be the number of sampled elements that possess a chosen characteristic. For large x_i and when q_i is not too near 0 or 1, the estimator $\hat{q}_i = (Y_i/x_i)$ is approximately normal with mean q_i and standard deviation $s_i = \sqrt{(q_i(1-q_i)/x_i)}$ [2]. As seen, the standard deviation can be reduced (and the estimate accuracy increased) by increasing the number of samples x_i. In the problem targeted in this section, n different populations can be sampled c times, and the goal is to distribute the samples among the populations to minimize the aggregated variance of the estimates. The problem can be reformulated as

$$\begin{aligned} \text{maximize} \quad & \sum_{i=1}^{n} -\frac{q_i(1-q_i)}{x_i} \\ \text{subject to} \quad & \sum x_i = c \\ & 1 \le x_i, \qquad i = 1, \ldots, n. \end{aligned}$$

The above optimization problem is an NEFK problem with concave and separable objective function. Since the q_i's are assumed unknown, we apply our LAKG to find a nearly optimal solution incrementally and online.

B. LAKG Solution

We must first define the stochastic environment that the LAKG is to interact with. That is, we must define the stochastic functions $\mathcal{F}' = \{F_1'(x_1), F_2'(x_2), \ldots, F_n'(x_n)\}$. By again applying the principle of Lagrange multipliers, we find the following conditions that characterize the optimal solution:

TABLE I
True Population Proportions Used in the Experiment, and Number of Populations Associated With Each Proportion

True Proportion	Populations
0.5	5
0.750 / 0.250	5
0.900 / 0.100	40
0.990 / 0.010	50
0.999 / 0.001	400

$$\begin{aligned} \frac{q_1(1-q_1)}{x_1^2} &= \cdots = \frac{q_n(1-q_n)}{x_n^2} \\ \sum x_i &= c \\ 1 \le x_i&, \quad i = 1, \ldots, n. \end{aligned}$$

Accordingly, we define $F_i'(x_i)$ as follows. First of all, each time $F_i'(x_i)$ is accessed by the LAKG, population i is sampled once, and the proportion estimate $\hat{q}_i$ is updated accordingly.[10] After $\hat{q}_i$ has been updated, we instantiate $F_i'(x_i)$ by a random draw—$F_i'(x_i)$ is instantiated to the value 0 with probability $1 - (\hat{q}_i(1-\hat{q}_i)/x_i^2)$ and to the value 1 with probability $\hat{q}_i(1-\hat{q}_i)/x_i^2$. In other words, we keep running estimates of the q_is in order to calculate the outcome probabilities of the $F_i'(x_i)$s.[11]

C. Empirical Results

In this section, we evaluate our learning scheme by comparing it with the optimal and uniform policies using synthetic data. Only the uniform policy can be applied in practice because the optimal policy requires that the q_i's are known.

The data used in the experiment are summarized in Table I. The table shows the true population proportions used and the number of populations associated with each proportion. The experiment encompasses 500 populations, and the corresponding proportions are to be estimated by allocating 50 000 samples (window based).

Fig. 7 plots the variance of the current solution each time a unit volume value function $F_i'(x_i)$ has been sampled. The graphs show the results of applying two LA configurations, namely: 1) $[N = 12\,500, \gamma = 1.1]$ and 2) $[N = 125\,000, \gamma = 1.1]$. As seen in the figure, LAKG steadily reduces the variance of the initial solution where the populations are sampled uniformly. Already after the first 50 000 samples, significant reduction can be observed. Notice that a larger number of automata states finds better values for $\vec{x}$ but possesses a slower convergence, and vice versa.

Fig. 8 plots the length of the widest 95% confidence interval among the n estimates after each sampling. We also plot the length of the fifth widest interval (first percentile), whence we see that the confidence interval of each estimated proportion is reduced by minimizing the total variance.

[10] For a dynamic environment, we would utilize a "window-based" strategy and only use the last c samples to estimate q_i. However, we are currently studying how recently proposed weak estimators can be used in this setting [11].

[11] Because the outcome probabilities are always available for the populations, we can normalize the outcome probabilities to speed up convergence.

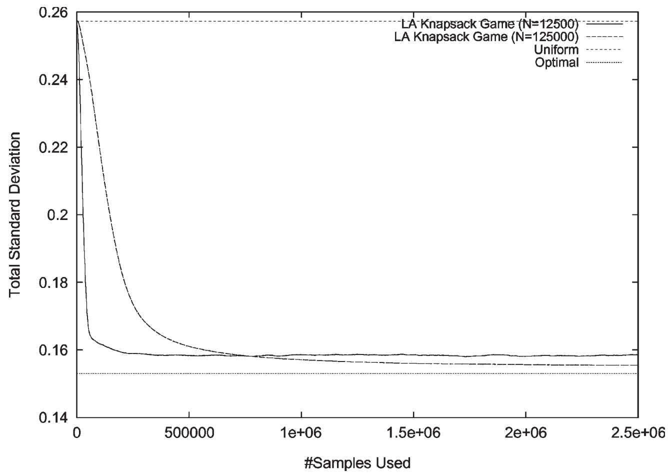

Fig. 7. LAKG steadily reduces the total variance of the initial solution (the uniform policy) as it moves toward nearly optimal solutions.

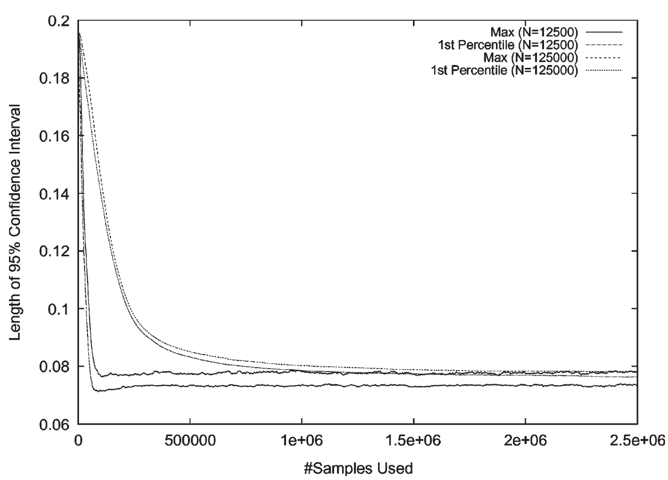

Fig. 8. Confidence interval of each estimated proportion is reduced as the total variance is minimized.

To conclude, our experimental results demonstrate that the discretization resolution determines the precision of our scheme. Additionally, for a given precision, our scheme determines the material fractions of maximal expected value by online interaction with the knapsack. Our results also demonstrate that the scheme adapts to switching material value distribution functions, permitting excellent performance for dynamic environments.

V. Conclusion and Further Work

In this paper, we extended the FK problem in two ways. First, we treat the unit volume value of each material as a "stochastic" variable of "unknown" distribution. Second, we assume that the expected value of a material may decrease after each addition to the knapsack. The learning scheme we proposed for solving this knapsack problem was based on a team of LA that performed a controlled random walk on a discretized fraction space. Comprehensive experimental results demonstrated that the discretization resolution determines the precision of our scheme. Additionally, for a given precision, our scheme determines the material fractions of maximal expected value by invoking online interactions with the knapsack. Finally, we demonstrated that the scheme adapts to switching material value distribution functions, allowing us to operate in dynamic environments.

In our further work, we aim to develop alternate LA-based solutions for different classes of knapsack problems, including the NP-hard integer knapsack problem. Essentially, we propose to do this by enhancing the concepts introduced in this paper with a branch-and-bound-based relaxation capability. Furthermore, we are working on a hierarchical variant of our scheme with the aim of tackling large-scale problems (e.g., 10 000 to 100 000 material problems). Finally, we also intend to implement a web crawler based on the results presented here.

References

[1] NIST, *The Fractional Knapsack Problem*, 2004. [Online]. Available: http://www.nist.gov/dads/HTML/fractionalKnapsack.html
[2] G. K. Bhattacharyya and R. A. Johnson, *Statistical Concepts and Methods*. Hoboken, NJ: Wiley, 1977.
[3] K. M. Bretthauer and B. Shetty, "The nonlinear knapsack problem-algorithms and applications," *Eur. J. Oper. Res.*, vol. 138, no. 3, pp. 459–472, May 2002.
[4] C. Castillo, "Effective web crawling," Ph.D. dissertation, Univ. Chile, Santiago, 2004.
[5] B. C. Dean, M. X. Goemans, and J. Vondrdk, "Approximating the stochastic knapsack problem: The benefit of adaptivity," in *Proc. 45th Annu. IEEE Symp. Found. Comput. Sci.*, 2004, pp. 208–217.
[6] B. Fox, "Discrete optimization via marginal analysis," *Manage. Sci.*, vol. 13, no. 3, pp. 211–216, 1966.
[7] H. Kellerer, U. Pferschy, and D. Pisinger, *Knapsack Problems*. Berlin, Germany: Springer-Verlag, 2004.
[8] K. S. Narendra and M. A. L. Thathachar, *Learning Automata: An Introduction*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[9] D. Ng, B. J. Oommen, and E. R. Hansen, "Adaptive learning mechanisms for ordering actions using random races," *IEEE Trans. Syst., Man, Cybern. A, Syst.*, vol. 23, no. 5, pp. 1450–1465, Sep./Oct. 1993.
[10] B. J. Oommen, "Stochastic searching on the line and its applications to parameter learning in nonlinear optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 27, no. 4, pp. 733–739, Aug. 1997.
[11] B. J. Oommen and L. Rueda, "Stochastic learning-based weak estimation of multinomial random variables and its applications to pattern recognition in non-stationary environments," *Pattern Recognit.*, vol. 39, pp. 328–341, 2006.
[12] S. Pandey, K. Ramamritham, and S. Chakrabarti, "Monitoring the dynamic web to respond to continuous queries," in *Proc. 12th Int. World Wide Web Conf.*, 2003, pp. 659–668.
[13] K. W. Ross and D. H. K. Tsang, "The stochastic knapsack problem," *IEEE Trans. Commun.*, vol. 37, no. 7, pp. 740–747, Jul. 1989.
[14] M. Snaprud, N. Ulltveit-Moe, O.-C. Granmo, M. Rafoshei-Klev, A. Wiklund, and A. Sawicka, "Quantitative assessment of public web sites accessibility-some early results," in *Proc. Accessibility All Conf.*, 2003.
[15] E. Steinberg and M. S. Parks, "A preference order dynamic program for a knapsack problem with stochastic rewards," *J. Oper. Res. Soc.*, vol. 30, no. 2, pp. 141–147, 1979.
[16] M. A. L. Thathachar and P. S. Sastry, *Networks of Learning Automata: Techniques for Online Stochastic Optimization*. Norwell, MA: Kluwer, 2004.
[17] M. L. Tsetlin, *Automaton Theory and Modeling of Biological Systems*. New York: Academic, 1973.
[18] J. L. Wolf, M. S. Squillante, J. Sethuraman, and K. Ozsen, "Optimal crawling strategies for web search engines," in *Proc. 11th Int. World Wide Web Conf.*, 2002, pp. 136–147.
[19] Z. Xiaoming, "Evaluation and enhancement of web content accessibility for persons with disabilities," Ph.D. dissertation, Univ. Pittsburgh, Pittsburgh, PA, 2004.
[20] Q. Zhu and B. J. Oommen, "Some fundamental results on optimal search with unknown target distributions," in *Proc. 17th Int. Conf. Chilean Comput. Sci. Soc.*, Valparaiso, Chile, Nov. 1997.
[21] G. Zipf, *Human Behavior and the Principle of Least Effort*. Reading, MA: Addison-Wesley, 1949.

Ole-Christoffer Granmo was born in Porsgrunn, Norway, on August 26, 1974. He received the M.Sc. and Ph.D. degrees from the University of Oslo, Oslo, Norway, in 1999 and 2004, respectively.

He is currently an Associate Professor in the Department of Information and Communication Technology, Agder University College, Grimstad, Norway. His research interests include stochastic modeling and inference, machine learning, pattern recognition, learning automata, distributed computing, and surveillance and intrusion detection in computer networks.

B. John Oommen (S'79–M'83–SM'88–F'03) was born in Coonoor, India, on September 9, 1953. He received the B.Tech. degree from the Indian Institute of Technology, Madras, India, in 1975, the M.E. degree from the Indian Institute of Science, Bangalore, India, in 1977, and the M.S. and Ph.D. degrees from Purdue University, West Lafayette, IN, in 1979 and 1982, respectively, all in electrical engineering.

In 1981, he joined the School of Computer Science, Carleton University, Ottawa, ON, Canada, where he is currently a Full Professor and holds the honorary rank of Chancellor's Professor. He is the author of more than 255 refereed journal and conference publications. His research interests include automata learning, adaptive data structures, statistical and syntactic pattern recognition, stochastic algorithms, and partitioning algorithms. He is an editor for *Pattern Recognition*.

Dr. Oommen is on the Editorial Board of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS. He is a Fellow of the IAPR.

Svein Arild Myrer was born in Oslo, Norway, on March 27, 1980. He received the M.S. degree in information and communication technology from Agder University College, Grimstad, Norway, in 2005.

He is currently a System Developer and Consultant with Bouvet, Oslo, Norway, where he is engaged in the development of portal solutions based on the ISO/IEC 13250:2003 topic map standard.

Morten Goodwin Olsen was born in Tønsberg, Norway, on November 23, 1980. He received the B.S. degree in computer science and the M.S. degree in information and communication technology from Agder University College, Grimstad, Norway, in 2003 and 2005, respectively. His thesis was on incremental web crawling using learning automata.

He is part of the European research project European Internet Accessibility Observatory and has research interests in web accessibility, web crawling, universal design, and learning algorithms.

Mr. Olsen received the award for Best Information and Communication Technology Thesis of the Year from Agder University College.