Noname manuscript No.
(will be inserted by the editor)

Solving Stochastic Nonlinear Resource Allocation Problems Using Continuous Learning Automata

Anis Yazidi and Hugo L. Hammer

Received: date / Accepted: date

Abstract This paper deals with the Stochastic Non-linear Fractional Equality Knapsack (NFEK) problem which is a fundamental resource allocation problem based on incomplete and noisy information [7, 8]. The NFEK problem arises in many applications such as in web polling under polling constraints, and in constrained estimation. The primary contribution of this paper is a *continuous* Learning Automata (LA)-based, *optimal*, efficient and yet simple solution to the NFEK problem. Our solution is distinct from the first-reported optimal solution to the problem due to Granmo and Oommen [7, 8] which resorts to utilizing multiple two-action discretized LA, organized in a hierarchical manner which comes with extra implementation and computational complexity. In this work, we present an optimal solution to the problem using a *continuous* LA which does not involve mapping the materials onto a binary hierarchy. As opposed to the traditional family of Reward-Inaction (R-I) LA, our scheme is modified in order to accommodate non-absorbing barriers, thus guaranteeing convergence to the optimal allocation. The experimental results that we have presented for numerous simulations demonstrate the efficiency of our scheme and its superiority compared to the state-of-the-art in terms of peak performance.

Keywords Continuous Learning Automata, Stochastic Non-linear Fractional Equality Knapsack, Resource Allocation.

The author gratefully acknowledges the assistance of Dr. B. John Oommen, a *Chancellor's Professor* from Ottawa, Canada. John helped a lot with reviewing the paper, its style and the language. Thank you, John!

Anis Yazidi
Dept. of Computer Science
Oslo Metropolitan University
E-mail: anis.yazidi@hioa.no

1 Introduction

This paper deals with the Stochastic Non-linear Fractional Equality Knapsack (NFEK) Problem which is the central underlying problem pertinent to allocating resources based on incomplete and noisy information. Such situations are not merely hypothetical – rather, they constitute the vast majority of allocation problems in the real-world. Resource allocation problems which involve such incomplete and noisy information are particularly intriguing. They cannot be solved by traditional optimization techniques, rendering them ineffective.

The NFEK problem involves n materials, $1 \leq i \leq n$, where 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. The problem is to fill a knapsack of fixed volume c with the material mix $\mathbf{x} = [x_1, \ldots, x_n]$ of maximal value $\sum_1^n f_i(x_i)$ [2]. It is characterized by separable and concave objective functions. The problem can be stated as follows [12]:

$$\begin{array}{l} \text{maximize } f(\mathbf{x}) = \sum_1^n f_i(x_i) \\ \text{subject to } \sum_1^n x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0. \end{array}$$

In the above, the objective function is concave implying that the value function $f_i(x_i)$ of each material is also concave, and that the derivatives, f_i', of the material value functions $f_i(x_i)$ with respect to x_i, are not constant but non-increasing. Thus, the optimization problem becomes:

$$\begin{array}{l} \text{maximize } f(\mathbf{x}) = \sum_1^n f_i(x_i), \text{where } f_i(x_i) = \int_0^{x_i} f_i'(x_i)dx_i \\ \text{subject to } \sum_1^n x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0. \end{array}$$

The *Stochastic* NFEK Problem[1] is a generalization of the above where the material value per unit volume for any x_i is a *probability* function $p_i(x_i)$, and to render the problem non-trivial, the distribution of $p_i(x_i)$ is assumed to be *unknown*. Since the unit volume values are random, the problem is formulated in terms of expected unit volume values rather than the actual unit volumes themselves. From this perspective, the expected value of the amount x_i of material i, $1 \leq i \leq n$, becomes $f_i(x_i) = \int_0^{x_i} p_i(u)du$. Accordingly, the expected value per unit volume[2] of material i becomes $f_i'(x_i) = p_i(x_i)$. In this stochastic and non-linear version of the FK problem, the goal is to fill the knapsack so that the expected value $f(\mathbf{x}) = \sum_1^n f_i(x_i)$ of the material mix contained in the knapsack is maximized. Thus, we aim to:

[1] This generalization is attributed to Granmo and Oommen [7,8].

[2] We hereafter use $f_i'(x_i)$ to denote the derivative of the expected value function $f_i(x_i)$ with respect to x_i.

$$\text{maximize} f(\mathbf{x}) = \sum_{1}^{n} f_i(x_i),$$

$$\text{where} f_i(x_i) = \int_0^{x_i} p_i(u)du, \text{ and } p_i(x_i) = f_i'(x_i),$$

$$\text{subject to} \sum_{1}^{n} x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0.$$

With regard to the model of the problem studied in [9], it assumes that one is provided with a knapsack of fixed volume c, to be filled with a mix of n different materials. However, unlike the NFEK, in the Stochastic NFEK Problem the unit volume value of a material i, $1 \leq i \leq n$, is random, and assumes the value 1 with probability $p_i(x_i)$ and the value 0 with probability $1 - p_i(x_i)$, respectively. As an additional complication, $p_i(x_i)$ is nonlinear in the sense that it decreases monotonically with x_i, i.e., $x_{i_1} \leq x_{i_2} \Leftrightarrow p_i(x_{i_1}) \geq p_i(x_{i_2})$.

The only available solution to the latter problem, which is on-line and incremental, works as follows. At each time instant, an amount x_i of material i is placed in the knapsack. The complexity of the problem arises because we are only allowed to observe an instantiation of $p_i(x_i)$ at x_i, and not $p_i(x_i)$ itself. The solution was able to converge to a mixture of the materials of maximal *expected* value, through a series of informed guesses.

Some real-life problems under uncertainty can be modelled as instances of the Stochastic NFEK Problem. In [7, 8], the web polling problem was addressed and modelled as *Stochastic* NFEK. The aim was to maximize the number of changes detected given limited polling capacity. The frequency of changes of the web pages are supposed to be unknown. In [7, 8], it was shown that the probability to uncover an update monotonically decreases as the polling probability of the web page increases. In [7], Granmo and Oommen show an application of *Stochastic* NFEK Problem to determining the optimal size for estimation from different classes where the optimality criterion is minimizing the aggregated variance. The authors suppose a large number of classes and a large number of elements per class. The proposed solution maintains a moving average estimate of each class i called $\hat{q}_i$. The feedback is "engineered" in a form of a random variable with outcomes reward or penalty based on a probability function that depends on $\hat{q}_i$. Under the latter settings, the probability of observing a reward decreases as the polling probability of the class increases. In the literature, there is a large class of multi-armed bandit problems that can be modelled as *Stochastic* NFEK Problem where the reward probability decreases as the probability of polling the arm increases. Examples of those problems include congestion monitoring under limited bandwidth [1], adaptive link monitoring in software defined networks [19] and dynamic probing for intrusion detection under resource constraints [15].

The cited paper, [9], has two main characterizing facets, namely, that the unit volume values of each material are *stochastic* variables with *unknown* distributions, and that the expected value of a material could decrease as additions are made to the knapsack. This optimal solution utilizes a hierarchy of two-action discretized Learning Automata (LA). Although this solution is elegant, its implementation is complex because it involves updates at different levels of a balanced binary tree.

The absolute majority of the work in LA can be seen as an instance of an *unconstrained* optimization problem where the objective is to maximize the reward over time. Loosely speaking, a possible approach for accommodating constraints is to redesign the feedback of the LA so that the penalty increases whenever the constraints are violated. A notable exception is the work due to Poznyak and Najim [30] that uses regularized Lagrange function for addressing constrained repeated games. Another possible avenue of research is to investigate deploying Adaptive Barrier Lyapunov [10,16,17]. Adaptive Barrier Lyapunov [3] converts a constrained optimization problem to an unconstrained one. By definition, a barrier function is a continuous function with values increasing to infinity in the boundary of the feasible region. Adaptive Barrier Lyapunov has been applied in stochastic approximation problems [10] and in [16,17] for adaptive control of for a class of stochastic nonlinear systems. The work [25] uses Lyapunov-like barrier functions for multi-agent coordination under different objectives, such as collision avoidance and maintaining the local laws leading to swarm behavior. As a future work, we plan to investigate making use of Lyapunov-like barrier functions for solving constrained reinforcement learning problems including the Stochastic NFEK problem[3].

2 Fractional Knapsack Problems: State-of-the-Art

Without going back too far into the past, we start discussions by mentioning that the Fractional Equality Knapsack problem, was studied in its virgin form in [6]. The problem that these authors studied involved the added complication that the *future* availability of the indivisible resources was uncertain due to the other players in the field. Their solution, on the other hand, invoked an intelligent agent-based stochastic ruler-based approach. It operated with the model in which the system had to execute a sequence of resource allocation decisions over time.

With regard to the *stochastic* knapsack problem, from a theoretical perspective, the authors of [14] studied it in the setting where the system permitted dynamic pricing *and* switch-over policies. Their solution represented a general discrete stochastic optimization model and utilized a multi-period bounded multiple-choice knapsack framework with a nonlinear objective function.

[3] We thank one anonymous reviewer for drawing our attention to the field of Adaptive Barrier Lyapunov Functions and its possible link to LA.

A distinct approximate solution was proposed more recently in [4]. In this paper, the authors presented a greedy non-adaptive algorithm that approximated the optimal adaptive policy within a factor of 7, demonstrating that adaptivity provided only a "constant-factor" improvement. They also designed an adaptive *approximate* polynomial-time algorithm to resolve the optimal adaptive policy within a factor of $5 + \epsilon$, for a given $\epsilon > 0$.

With regard to applications, the stochastic knapsack formulation has been used to distributing layered encoded videos by caching [11]. The authors of [29] also solved a more practical problem, where they modeled the multi-period, single resource capacity reservation problem as a dynamic, stochastic, multiple knapsack problem. Here, the state space grew exponentially with the number of knapsacks. Since the task of computing the optimal solution is computationally intractable, their solution was approximate, but fast. In a similar vein, Sachs [35] enforced a stochastic knapsack model to study the capacity evaluation of multi-radio access networks. His solution was unique in that it was able to consider the non-uniform geographic distribution of both the radio link capacities and their corresponding traffic loads.

The authors of [4, 40] considered solution policies for stochastic generalizations of the NP-hard *linear* integer knapsack problem. In their works, they assumed that the value distributions were considered known and constant. Such an assumption rendered the use of dynamic programming to be a viable solution. Another variant of the knapsack problem was found in [34] where, although the problem studied involved a *deterministic* knapsack, the randomization complexity was obtained by having objects arriving to and departing from the knapsack at random times. The optimization problem that was then considered involved accepting or blocking the arriving objects so that the average value of the knapsack was maximized.

In a similar vein, [13], Kosuch and Lisser treated the chance-constrained model for knapsack problems with random weights. However, their model assumed a two-stage stochastic program with recourse, i.e., it provided the scheme the possibility to remove or add items in the second stage, after their weights were revealed.

To conclude this survey, we mention that although several studies on related problems were reported earlier, the pioneering proposal and solution to the stochastic NFEK problem was reported in [9]. Various specific instantiations, particularly within the web monitoring domain, were reported earlier. In these, the web pages were polled mainly for estimation purposes [26, 44], and the unknown parameters were *estimated* by means of a tracking phase. The real drawback of such an approach is that the parameter estimation phase was done prior to the assignment, and this had the effect that it significantly delayed the computation of the solution, which was further exasperated in *dynamic* environments, because, in such settings, the optimal solution would be time-varying, introducing the need for additional re-invocations of the parameter estimation phase [9].

In contrast to the above approaches, we base our work on the principles of LA [20]. LA are stochastic machines that have been used to model biological

systems [41]. They have attracted considerable interest in the last few decades because they are able to learn the optimal actions when operating in (or interacting with) unknown stochastic environments. Furthermore, they combine rapid and accurate convergence with low computational complexity. The theory of LA has found numerous applications in the field of computer science. One of the most recent applications of LA include sampling algorithms for stochastic graphs [31], trust propagation in online social networks [36], allocation hub location problem [5], selecting caching nodes in delay tolerant networks [18] and feature subset selection [38] to mention a few. For an updated overview over the theory and applications of LA we refer the reader to the following book [32] and to a recent special issue [33] dedicated to the applications of LA.

The novel Learning Automata Knapsack Game (LAKG) scheme that was proposed in [9] does not rely on estimating parameters, and can be used to solve the stochastic NFEK problem in both static and dynamic settings.

2.1 The Hierarchy of Twofold Resource Allocation Automaton (H-TRAA) Solution

The state-of-the-art scheme for hierarchically solving n-material problems [7, 8] involves a primitive module, namely the Twofold Resource Allocation Automaton (TRAA) for the *two-material* problem. This module has been proven to be asymptotically optimal. The authors of [7, 8] then used the primitive TRAA as a building block, and arranged a set of TRAAs in a hierarchy so as to solve *multi-material* Stochastic NFEK Problems.

The hierarchy of TRAAs, referred to as H-TRAA, assumes that $n = 2^\gamma, \gamma \in \mathbb{N}^+$. If the number of materials is less than this, one trivially assumes the existence of additional materials whose values are "zero", and which are, thus, not able to contribute to the final optimal solution. The hierarchy is organized as a balanced binary tree with depth $D = \log_2(n)$. Each node in the hierarchy can be related to three entities: (1) a set of materials, (2) a partitioning of the material set into two subsets of equal size, and (3) a dedicated TRAA that allocates a given amount of resources among the two subsets. At depth D, then, each individual material can be separately assigned a fraction of the overall capacity by way of recursion, using a subtle mechanism described, in detail, in [8]. The principal theorem that guarantees the convergence of the H-TRAA [7, 8] has cleverly shown that if all the individual TRAAs converge to their *local* optimum, the overall system attains to the global optimum.

2.2 Contributions of this Paper

The contributions of this paper are the following:

1. We report an optimal solution to the stochastic NFEK problem based on the theory of *continuous* LA. The solution is based on the principles of

the Linear Reward-Inaction (L_{RI}) scheme. Through a subtle modification of the L_{RI}, we introduce artificial barriers that have the effect that they prevent the instantaneous allocation's probability vector from getting trapped in a unit vector in the $\mathbb{R}^n$ space, where n is the number of materials.
2. The scheme has been shown to converge to the unique optimal solution even though our scheme is ergodic, as opposed to being absorbing.
3. We provide a formal and rigorous *analysis* for our solution, termed as the Continuous Multi-action Learning Automata Solution (CMLS), based on the theory of small-step learning processes, due to Norman [22].
4. In contrast to the H-TRAA solution [7, 8], our CMLS solution does not involve a hierarchy, and it is thus easier to implement. This is because, in fact, TRAAs must be arranged in a hierarchy in order for them to be able to solve a *multi-material* Stochastic NFEK Problems. Further, through empirical experiments, we confirm that the CMLS provides desirable convergence properties that makes it superior to the H-TRAA.

As a result of the above contributions, we believe that the CMLS is a viable realistic strategy for solving demanding real-world knapsack-like problems such as the optimal allocation of sampling resources, and other problems related to the world wide web [9].

2.3 Paper Organization

The paper is organized as follows. In Section 3 we present the CMLS for the *n-material* problem, and prove its asymptotic optimality. We proceed in Sections 4 to empirically verify that the CMLS solution provides superior convergence results to the H-TRAA while being, at the same time, simpler to implement. Finally, we offer suggestions for further work and conclude the paper in Section 5.

3 A CMLS Solution to Resource Allocation

3.1 The Need for a Continuous-Space Solution

One can indeed, raise the question: If the H-TRAA solution is valid, why should we seek for an alternative. The reason is precisely because unlike traditional LA, in which discretization is advantageous, within the area of the knapsack-like problems, this is not the case. Indeed, the disadvantages of working within a discretized probability space for such problems are the following:

- The first optimal solution to the problem due to Granmo and Oommen [7, 8] resorts to the concept of primitive two-action discretized automata arranged hierarchically. The main shortcoming of the latter solution is its

complexity as it involves different *levels of updates* in the hierarchy. It also demands the splitting of the sets of materials into disjoints sets. The $H - TRAA$ solution assumes that $n = 2^{\gamma}, \gamma \in \mathbb{N}^{+}$, i.e, the number of materials is a power of 2. If the number of materials is less than this, one assumes the existence of additional materials whose values are "zero", and which are thus not able to contribute to the final optimal solution. We would like to remove this constraint.
- According to Granmo and Oommen [7,8], the efficiency of the H-TRAA is dependent on placing the items that get frequently updated together in the same branches in order to speed up the convergence. In fact, the leaf nodes have a resolution of N^{γ} where N is the resolution of each level. For instance, with a resolution 10 for each TRAA, and a number of levels $\gamma = 6$ the leaf nodes will have resolution 10^6, which would slow down the convergence of the scheme. Thus, the resolution of the leaf nodes grow at an exponential rate as the number of levels increase.
- The $H - TRAA$ can potentially have many layers and consequently, a lot of updates, so as to maintain the integrity of the entire tree. The implementation of the scheme is thus, complex.

In this perspective, the disadvantage of the $H - TRAA$ solution [7,8] is the fact that it is forced to resort to a hierarchy of LA whenever the number of actions exceeds 2 which is not a necessity for our scheme. For example, for 3 actions, a tree with 4 leaf nodes needs to be created, while the fourth lead node is a simply dummy node. Our solution can also be easily generalized to a hierarchical setting in a similar manner to [7,8] but the main difference between our solution and the $H - TRAA$ solution is the fact that the tree in our solution does not need to be binary. According to the literature, it is worth mentioning that hierarchical LA are suitable for a large number of actions. Therefore, our CMLS solution can benefit from using a hierarchy, which is not necessary binary tree, for a large number of materials.

3.2 Overview of the CMLS Solution

Our aim is to find a scheme that optimizes the following NFEK problem in an *on-line* manner:

$$\text{maximize} f(\mathbf{x}) = \sum_{1}^{n} f_i(x_i),$$
$$\text{where} f_i(x_i) = \int_0^{x_i} p_i(u)du, \text{ and } p_i(x_i) = f_i'(x_i),$$
$$\text{subject to} \sum_{1}^{n} x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0.$$

Note that we allow only instantiations of the material values per unit volume to be observed. That is, each time an amount x_i of material i is placed in the knapsack, an instantiation v_i at x_i is observed.

Because of the above intricacies, we approach the problem by relying on informed material mix *guesses*, i.e., by experimenting with different material mixes and learning from the resulting random unit volume value outcomes. 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, then, 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 using continuous Multi-action LA.

3.3 Details of the CMLS Solution

The family of LA used to develop our solution is akin to the acclaimed Linear Reward-Inaction (L_{RI}) scheme. However, this landmark scheme is now extended to develop the CMLS. It is worth highlighting the distinct way by which this extension is developed: While the CMLS is, in essence, absorbing, we have rendered the scheme to be artificially *ergodic*, and this consequent ergodicity provides us with the stochastic properties required for any solution to a knapsack-like problem, like the hierarchical H-TRAA.

3.3.1 Absorbing vs. Ergodic L_{RI} LA

Before we proceed, it is prudent to record our first primary contribution to the field of LA. The literature reports a few ergodic LA that have been rendered to be absorbing by artificially introducing absorbing barriers [24]. Our current scheme is arguably, the first reported algorithm in which the Markov Chain, that is inherently absorbing, has been rendered ergodic, by removing the absorbing nature of all its limiting barriers. Although the consequence and significance of this transformation will be evident presently, the strategy for achieving this is as follows: Rather than use the actual limits of the probability space as zero and unity, we work with the constraint that no probability value can take on a value below a pre-specified lower threshold of x_{min} or a value above a pre-specified upper threshold of x_{max}. The action-choosing probability values, which traditionally move proportionally towards zero and unity for the (L_{RI}) scheme are now made to move towards the respective values of x_{min} and x_{max} respectively. Interestingly enough, this minor modification renders the scheme to be ergodic, making the analysis also to be correspondingly distinct from that of the L_{RI}.

To achieve this, we enforce a minimal value x_{min}, where $0 < x_{min} < 1$ for each selection probability x_i, where $1 \leq i \leq r$ and r is the number of actions. As a result the maximum value any selection probability x_i, where $1 \leq i \leq r$, can achieve is $x_{max} = 1 - (r-1)x_{min}$. This happens when the other $r-1$ actions take their minimum value x_{min}, while the action with the

highest probability takes the value x_{max}. Therefore x_i, for $1 \leq i \leq r$, will take value in the interval $[x_{min}, x_{max}]$. The informed reader would remark that in the classical L_{RI}, $x_{max} = 1$ while $x_{min} = 0$. Therefore our scheme can be seen as a generalized version of the L_{RI}. However, if we use classical L_{RI}, the instantaneous allocation's probability vector will get trapped in a unit vector in the $\mathbb{R}^n$ space, which is undesirable in this case as it does not yield an optimal solution for our problem. The formal scheme for accomplishing this follows.

3.3.2 *The CMLS Solution*

The *Stochastic Environment* for the n materials case can be characterized by:

1. The capacity c of the knapsack;
2. $n-material$ unit volume value probability functions $[p_1(x_1), \ldots, p_n(x_n)]$.

In brief, if the amount x_i of material i is suggested to the Stochastic Environment, the Environment replies with a unit volume value $v_i = 1$ with probability $p_i(x_i)$ and a unit volume value $v_i = 0$ with probability $1 - p_i(x_i)$. To render the problem both interesting and non-trivial, we assume that $p_i(x_i)$ is unknown to the LA.

Let $\alpha(t)$ be the index of the chosen action at time instant t. Then, the value of $x_i(t)$ is updated as per the following simple rule (the rules for other values of $x_j(t), j \neq i$, are similar):

$$\begin{aligned} x_i(t+1) \quad & \leftarrow x_i(t) + \theta(x_{max} - x_i(t)) \\ & \text{when } \alpha(t) = i \text{ and } v_i = 1 \\ x_i(t+1) \quad & \leftarrow x_i(t) + \theta(x_{min} - x_i(t)) \\ & \text{when } \alpha(t) = j, j \neq i \text{ and } v_i = 1, \end{aligned}$$

where θ is a user-defined parameter $0 < \theta < 1$, typically close to zero.

We shall first characterize the optimal solution to a Stochastic NFEK Problem.

Lemma 1 *The material mix* $\mathbf{x} = [x_1, \ldots, x_n]$ *is a solution to a given Stochastic NFEK Problem if (1) the derivatives of the expected material amount values are all equal at* $\mathbf{x}$*, (2) the mix fills the knapsack, and (3) every material amount is positive, i.e.:*

$$\begin{aligned} & f_1'(x_1) = \cdots = f_n'(x_n) \\ & \textstyle\sum_1^n x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0. \end{aligned}$$

The above lemma is based on the well-known principle of Lagrange Multipliers, and its proof is therefore omitted here for the sake of brevity [7,8].

3.4 The Formal Analysis of CMLS

We now analyze the feedback connection of the CMLS and the Stochastic Environment, and thus prove that the CMLS is asymptotically optimal in the sense that it can find material allocations arbitrarily close to the solution of the Stochastic NFEK Problem. But before we embark on such a formal proof, it is necessary for us to see how the traditional proof for the L_{RI} scheme is achieved.

3.4.1 Martingale-based Proofs for Absorbing LA

Consider the recorded proof of the convergence accuracy of the L_{RI} scheme [20]. If α_m is the best action, the concluding step of the proof of the ϵ-optimality of the L_{RI} scheme is based on the condition that the sequence of $\{p_m(t)_{t>t_0}\}$ is a submartingale. By invoking the martingale convergence theorem, this is shown to be equivalent to proving that $Pr\{p_m(\infty) = 1\} \rightarrow 1$ as $\theta \rightarrow 0$, where θ is the parameter of the L_{RI} scheme. To achieve this, one utilizes the theory of Regular functions. Firstly, the action selecting probabilities are represented in their vectorial form. One then observes that the goal is to prove that the convergence probability $\Gamma_m(P) = Pr\{p_m(\infty) = 1\} = Pr\{P_m(\infty) = e_m\} \rightarrow 1$. Secondly, as per the definition of Regular functions and the submartingale convergence theory, one shows that $\Gamma_m(P) = U^{\infty}(\Phi(P))$, where U and Φ are a functional operator and a clearly-specified function respectively [20]. More precisely, the convergence probability is exactly the function $\Phi(P)$, upon which, the operation U is applied an infinite number of times. Further, if Φ is a Regular function of P, then $\Gamma_m(P) = U^{\infty}(\Phi(P)) = \Phi(P)$. This implies that the convergence probability can be investigated by means of a Regular function of P. However, it is not easy to find such a Regular function, though its existence is guaranteed. Therefore, one resorts to utilizing a subregular function to bound the unique Regular function, i.e., the convergence probability, from below. We thus prove the result that $\Gamma_m(P) \rightarrow 1$ by showing that this subregular function converges to 1 as $\theta \rightarrow 0$.

The above proof would work only for the case when the unit vectors are the absorbing barriers of the scheme. It would thus be ineffective if the chain is ergodic. This motivates the need for a proof that invokes a completely different strategy, which is what we embark on now.

3.4.2 Norman's Strategy for Analyzing Ergodic LA

Norman theory was used in the field of LA in the case of ergodic LA to proof the convergence to an optimal point that is different from the convergence to a unit vector in the space. However, our scheme is the first ergodic scheme that uses the phenomena associated with the L_{RI} concept in non-stationary environments. The most widespread method for similar schemes is the Linear Reward-ϵ Penalty [21,39]. Norman theory was applied in conflict avoidance algorithms for WDM networks [27,28] and also for project scheduling [42].

3.4.3 Main theorem

We will first present a theorem due to Norman, who studied *distance diminishing models* in probability spaces [22]. This result will be heavily used in our proof. Indeed, the convergence of $x(t)$ to x^* is a consequence of this theorem.

Theorem 1 *Let $\{x^\theta(t)\}$ be a stationary Markov process dependent on a constant parameter $\theta \in [0,1]$. Each $x^\theta(t) \in I$, where I is a subset of the n dimensional space $\mathbb{R}^n$. Let $\delta x^\theta(t) = x^\theta(t+1) - x^\theta(t)$. We are concerned with the behavior of $\delta x^\theta(t)$ as $\theta \to 0$ and $t\theta \to \infty$. The following assumptions are assumed to hold:*

1. $E[\delta x^\theta(t) | x^\theta(t) = y] = \theta W(y) + O(\theta^2)$
2. $E[[\delta x^\theta(t) - W(y)][\delta x^\theta(t) - W(y)]^T | x(t) = y] = \theta^2 s(y) + o(\theta^2)$
3. $E[\mid \delta x^\theta(t) \mid^3 \mid x(t) = y] = O(\theta^3)$ *where* $\mid \delta. \mid$ *is a norm defined in I and all orders of magnitude are uniform in y. Here we define:* $\mid x \mid^3 \mid = \sum_{i=1}^n \mid x_i \mid^3$.
4. *$W(y)$ has bounded Lipschitz derivative in I.*
5. *$s(y)$ is Lipschitz I.*

Let $\mu_t(\theta, x) = E[x^\theta(t) | x^\theta(0) = x$ and $w_t(\theta, x) = [[\delta x^\theta(t) - \mu_t(\theta, x)][\delta x^\theta(t) - \mu_t(\theta, x)]^T | x^\theta(0) = x]$. Let $W'(y)$ denote the jacobian of $W(y)$ and $< x, y >$ denote an inner product in $\mathbb{R}^n$.

If, in addition to Assumptions (i)-(v):

1. *I is compact.*
2. *there exist a unique $\lambda \in I$ such that $W(\lambda) = 0$.*
3. *$< z, W'(y)z > < 0$ for all $z \in \mathbb{R}^n$ and $z \neq 0$.*

then the following conclusions are true when $\delta x^\theta(t)$ as $\theta \to 0$ and $t\theta \to \infty$:

1. *$w_t(\theta, x) = 0(\theta)$ uniformally for all $x \in I$ and $\tau \geq 0$.*
2. *For any $x \in I$, the (vector) differential equation*

$$f'(\tau) = W[f(\tau)] \tag{1}$$

has a unique solution $f(\tau)$, where $f(0) = x$ and

$\mu_t(\theta, x) = f(t\theta) + 0(\theta)$ uniformally for $x \in I$ and $t \geq 0$.

3. *The (matrix) differential equation*
$g'(\tau) = W'[f(\tau)]g(\tau) + g(\tau)W'^T[f(\tau)] + S[f(\tau)]$ has a unique solution $g(\tau)$ with $g(0) = 0$ as follows
$\Xi(Z_t^\theta) \sim N(0, g(\tau))$ as $\theta \to t < \infty$
where $Z_t^\theta = \frac{x^\theta(t) - f(t\theta)}{\sqrt{\theta}}$, ($\Xi$ refers to the distribution of x and $N(a, b)$ refers to the normal distribution with mean a and covariance matrix b) and as $\theta \to 0$, $t\theta \to \infty$, $g(\infty)$ is obtained as the unique solution of the system of the linear equations

$$W'(\theta)g(\infty) + g(\infty)W'^T(\theta) + S(\theta) \tag{2}$$

Proof The proof of the above theorem is due to Norman and can be found in [23].

Using the above, we present the main theorem of our work.

Theorem 2 *For sufficiently small x_{min} approaching 0, the CMLS solution scheme specified by (1)–(2) is asymptotically optimal.*

$\lim_{\theta \to 0} \lim_{t \to \infty} E(x(t)) = x^*$

Proof Without loss of generality, we assume that $c = 1 = \sum_{i=1i}^{n} x_i$, i.e, that the x_i's are normalized. The case where $c \neq 1$ can also be handled by invoking a normalization phase, and so the materials are accessed according to a probability vector. The proof is divided into three main parts.

Part 1: In the first part of the proof, the most crucial part is verifying the conditions (i)-(v) of Theorem 1. We first start by defining $\delta x_i(t) = x_i(t+1) - x_i(t)$.

$$\begin{aligned} E[\delta x_i(t)|x(t) = x] = x_i p_i(x_i)[\theta(x_{max} - x_i)] \\ + \sum_{\substack{j=1 \\ j \neq i}}^{n} x_j p_j(x_j) \\ . [\theta(x_{min} - x_i)], \end{aligned}$$

where $x = (x_1, x_2, ..., x_n)$. Then:

$$\begin{aligned} & E[\delta x_i(t)|x(t) = x] = \\ & \quad x_i p_i(x_i).[\theta(1 - x_{max} + 1 - x_i)] \\ & \quad + \sum_{\substack{j=1 \\ j \neq i}}^{n} x_j p_j(x_j)[\theta(x_{min} - x_i)] \\ & = x_i p_i(x_i) \\ & \quad . [\theta(1 - x_{max} + \sum_{\substack{j=1 \\ j \neq i}}^{n} x_j)] \\ & \quad + \sum_{\substack{j=1 \\ j \neq i}}^{n} x_j p_j(x_j)[\theta(x_{min} - x_i)]. \end{aligned} \tag{3}$$

(4)

By taking into account the fact that $1 - x_{max} = (r-1)x_{min}$, Eq. (4) can be simplified (after some algebraic manipulations) and written as:

$$
\begin{aligned}
&E[\delta x_i(t)|x(t)=x] = \\
&\quad \theta \sum_{\substack{j=1 \\ j\neq i}}^{n} x_i x_j (p_i(x_i) - p_j(x_j)) \\
&\quad + \theta x_{min} (\sum_{\substack{j=1 \\ j\neq i}}^{n} x_j p_j(x_j)) \\
&\quad - \theta(r-1) x_{min} x_i p_i(x_i) \\
&= \theta \sum_{\substack{j=1 \\ j\neq i}}^{n} x_i x_j (p_i(x_i) - p_j(x_j)) \\
&\quad + \theta x_{min} \sum_{\substack{j=1 \\ j\neq i}}^{n} (x_j p_j(x_j) - x_i p_i(x_i)) \\
&\approx \theta w_i(x),
\end{aligned}
$$

where $w_i(x)$ is defined by $w_i(x) = \sum_{\substack{j=1 \\ j\neq i}}^{n} x_i x_j (p_i(x_i) - p_j(x_j))$.

For small values of x_{min}, i.e, as $x_{min} \to 0$, we can approximate $E[\delta x_i(t)|x(t) = x]$ by:

$$E[\delta x_i(t)|x(t)=x] = \theta w_i(x). \tag{5}$$

Part 2 Existence and Uniqueness: In the second part of the proof, we will verify the uniqueness and existence of the root of $w(x)$. We will show that $w(x) = (w_1(x), w_2(x), ..., w_n(x))$ has a unique zero in the neighborhood of $x^* = (x_1^*, ..., x_n^*)$, where x^* is the solution of the Lagrange multipliers' conditions obtained in Proposition 1. Let us consider $w(x) = (w_1(x), w_2(x), ..., w_n(x)) = (0, 0, .., 0)$.

This implies that the system of n equations

$$
\begin{cases}
\sum_{\substack{j=1 \\ j\neq 1}}^{n} x_1 x_j (p_1(x_1) - p_j(x_j)) = 0 \\
\sum_{\substack{j=1 \\ j\neq 2}}^{n} x_2 x_j (p_2(x_2) - p_j(x_j)) = 0 \\
\vdots \\
\sum_{\substack{j=1 \\ j\neq n}}^{n} x_n x_j (p_n(x_n) - p_j(x_j)) = 0.
\end{cases}
$$

$$
\Leftrightarrow
\begin{cases}
x_1 \sum_{\substack{j=1 \\ j\neq 2}}^{n} x_j (p_1(x_1) - p_j(x_j)) = 0 \\
x_2 \sum_{\substack{j=1 \\ j\neq 2}}^{n} x_j (p_2(x_2) - p_j(x_j)) = 0 \\
\vdots \\
x_n \sum_{\substack{j=1 \\ j\neq n}}^{n} x_j (p_n(x_n) - p_j(x_j)) = 0.
\end{cases}
$$

The reader should observe that a crucial concept in our approach is that we are using the barrier x_{min}, which ensures that $x_1 \neq 0$, $x_2 \neq 0 ... x_n \neq 0$. We can thus confidently divide the first equation by x_1, the second equation by x_2 and so on, yielding:

$$\Leftrightarrow \begin{cases} \sum_{\substack{j=1 \\ j\neq 1}}^{n} x_j(p_1(x_1) - p_j(x_j)) = 0 \\ \vdots \\ \sum_{\substack{j=1 \\ j\neq 2}}^{n} x_j(p_2(x_2) - p_j(x_j)) = 0 \\ \vdots \\ \sum_{\substack{j=1 \\ j\neq n}}^{n} x_j(p_n(x_n) - p_j(x_j)) = 0. \end{cases}$$

After invoking some algebraic manipulations, we obtain that:

$$\Leftrightarrow \begin{cases} p_1(x_1) = \sum_{j=1}^{n} x_j p_j(x_j) \\ \vdots \\ p_2(x_2) = \sum_{j=1}^{n} x_j p_j(x_j) \\ \vdots \\ p_1(x_n) = \sum_{j=1}^{n} x_j p_j(x_j), \end{cases}$$

which implies that $p_1(x_1) = p_2(x_2) = \ldots = p_n(x_n)$

Thus, $w(x) = (w_1(x), w_2(x), ..., w_n(x))$ has a zero in the neighborhood of $x^* = (x_1^*, ..., x_n^*)$, where x^* is the solution of the Lagrange multipliers conditions obtained in Proposition 1. The existence of the solution is thus proven.

Now we will show that the solution is unique.

Uniqueness: The uniqueness of x^* is proven by contradiction. Suppose there exists $y^* = (y_1^*, y_2^*, ..., y_n^*)$ that is a zero of $w(y)$ such that $x^* \neq y^*$.

Without loss of generality since x^* and y^* are two probability vectors such that $x^* \neq y^*$, we are guaranteed[4] that they have at least two components i and j such that $x_i^* > y_i^*$ and $x_j^* < y_j^*$. Intuitively this means, that if we increase any one component of a probability vector, we should decrease another component so as to ensure that the sum of the components is still unity.

Suppose now that $x_i^* > y_i^*$. Then, by invoking the monotonicity of the function $p_i(.)$, we obtain that $p_i(x_i^*) < p_i(y_i^*)$. On the other hand, the condition $x_j^* < y_j^*$ implies that $p_j(x_j^*) > p_j(y_j^*)$, where this is obtained by virtue of the monotonicity of $p_j(.)$. But since x^* and y^* are equilibrium points, we know that $p_i(x_i^*) = p_j(x_j^*)$ and that $p_i(y_i^*) = p_j(y_j^*)$. This forces a contradiction since it is impossible to simultaneously maintain that : $p_i(x_i^*) < p_i(y_i^*)$ which is equivalent to $p_j(x_j^*) < p_j(y_j^*)$ and $p_j(x_j^*) > p_j(y_j^*)$.

Therefore x^* is unique.

[4] Please note that the result is general and applies for any two distinct probability vectors.

Part 3 Jacobian Properties at x^*: Now, we move to the last part of the proof. Consider the Jacobian at the point $x^* = (x_1^*, x_2^*, ..., x_n^*)$, where the matrix A has the arbitrary element $a_{ij} = dw_i/dx_j\Big|_{x=x^*}$.

We can see that for $i \neq j$, $a_{ij} = \frac{dw_i}{dx_j}\Big|_{x=x^*} = x_i^* x_j^* \frac{dp_j(x_j)}{dx_j}\Big|_{x=x^*}$.

Similarly, for $i = j$, $a_{ii} = \frac{dw_i}{dx_i}\Big|_{x=x^*} = -x_i^*(1-x_i^*)\frac{dp_i(x_i)}{dx_i}\Big|_{x=x^*}$.

A careful examination of the above matrix A, permits us to re-write it as the product of two matrices B and D, where, $A = BD$, and where B and D are the two matrices defined below.

$$B = \begin{pmatrix} -x_1^*(1-x_1^*) & x_1^*x_2^* & \cdots & x_1^*x_n^* \\ x_2^*x_1^* & -x_2^*(1-x_2^*) & \cdots & x_2^*x_n^* \\ \vdots & \vdots & \ddots & \vdots \\ x_n^*x_1^* & x_n^*x_2^* & \cdots & -x_n^*(1-x_n^*) \end{pmatrix},$$

$$D = \begin{pmatrix} \frac{dp_1(x_1)}{dx_1}\Big|_{x_1=x_1^*} & 0 & \cdots & 0 \\ 0 & \frac{dp_2(x_2)}{dx_2}\Big|_{x_2=x_2^*} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{dp_n(x_n)}{dx_n}\Big|_{x_n=x_n^*} \end{pmatrix}.$$

It is clear that D is a negative semi-definite matrix since $\frac{dp_i(x_i)}{dx_i}\Big|_{x_i=x_i^*} < 0$ (it is negative because the function itself is decreasing).

We will show that the matrix B is positive semi-definite. First of all, observe that B is symmetric. Consider now the term $z^T Bz$ where z is a vector of dimension n. Then:

$z^T Bz = \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} x_i^* x_j^* (z_i - z_j)^2 \geq 0$

Based on the results in [37], more generally, if D is a diagonal negative definite matrix, and B is a positive definite matrix, we can infer that the product BD is negative definite.

Rest of the Conditions: It is immediately seen that Assumptions (ii)-(v) are satisfied. It can also be proven that $s(.)$ and $W'(.)$ are Lipschitz by showing that their first derivatives are bounded.

For small values of x_{min}, it can be shown that the zeros of $W(x)$ remain in the vicinity of x^*. Theorem 1 can now be applied, implying that as $\theta \to 0$, and $t\theta \to \infty$, the normalized random vector $\frac{x(t)-f(t\theta)}{\sqrt{\theta}}$ converges in distribution. Therefore, $x(t)$ converges in distribution to a normal distribution with mean x^* independent of the initial allocation vector $x(0)$, which concludes that $E(x(t))$ converges to x^*.

Following closely the techniques in [23], it is easy to prove that for a sufficiently small x_{min} approaching 0 and for any $\delta > 0$, there exists $0 < \theta^* < 1$

such that for all $0 < \theta < \theta^* < 1$ we have:

$$\lim_{t \to \infty} |E(x(t)) - x^*| < \delta$$

4 Empirical Results

The CMLS solutions has been rigorously tested for numerous cases and the solutions obtained have been, in our opinion, categorically remarkable. It is superior to the H-TRAA in terms of its simplicity, low computational complexity and peak performance.

4.1 Problem Specification

We have tested our algorithm against some datasets used in the past so that they can serve as benchmarks. [5] These are two parametric objective functions (referred to as $E_i(x_i)$ and $L_i(x_i)$) to be optimized, given below for a material with index i as:

$$E_i = \frac{0.7}{i}(1 - e^{-ix_i}) \tag{6}$$

$$L_i = 0.7 \cdot x_i - \frac{1}{2} i \cdot x_i^2, \qquad \text{If } x_i \leq \frac{0.7}{i} \tag{7}$$

$$= \frac{0.7^2}{i}. \qquad \text{If } x_i > \frac{0.7}{i}. \tag{8}$$

To ease the readability, we have used the notation that the profitability of materials that have a smaller index decreases *slower* than the profitability of materials that have higher indices.

The constants in the above functions (Eq. (6-8)) are based on the boundary conditions due the contributions of x_i at the boundary values, and are not crucial in the optimization process. This is because the corresponding unit value functions are the respective derivatives of the functions, and these derivatives fall exponentially and linearly as per Eq. (9) and (10) respectively:

$$E_i'(x_i) = 0.7 \cdot e^{-i \cdot x_i} \tag{9}$$

$$L_i' = \text{Max}\,[0.7 - i \cdot x_i, 0]. \tag{10}$$

It is expedient to glean some input about the significance of these unit value functions. To understand this, consider the functions $E_i'(x_i)$, where the

[5] We have also conducted experiments with a number of other objective functions, including those reported in [7,8]. But $E_i(x_i)$ and $L_i(x_i)$ are particularly useful in the sense that they appropriately model a large family of distinct material unit value functions used for these datasets are representative of the class of concave objective functions addressed here.

relative profitability of material i decreases with x_i, its presence in the mixture, exponentially. Indeed, if $x_2 = 0.3$ (i.e., material 2 fills 30% of the knapsack), the marginal profitability of increasing the amount of x_2 is $e^{-2\cdot(0.3)} = e^{-0.6}$.

Unlike the exponential function, the linear function, $L_i'(x_i)$ has an interesting peculiarity that the function for material i intersects the X-axis at a finite point, implying that the function being optimized is quadratic. Thus, it attains a maximum value at this point, after which it remains constant. Clearly, after this intersection point, it is futile to add any additional quantity of material i.

From an experimental viewpoint[6], our aim is to find $\mathbf{x}$, the amounts of the materials to be added into the knapsack so as to maximize its value. In the case of the first function, the goal is to:

$$\text{maximize} E(\mathbf{x}) = \sum_{1}^{n} E_i(x_i),$$
$$\text{where } E_i(x_i) = \int_0^{x_i} p_i(u)du, \text{ and } p_i(x_i) = E_i'(x_i),$$
$$\text{subject to } \sum_{1}^{n} x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0,$$

where $E_i'(x_i)$ is given by Eq. (9).

Further, in the second case we aim to:

$$\text{maximize} L(\mathbf{x}) = \sum_{1}^{n} L_i(x_i),$$
$$\text{where } L_i(x_i) = \int_0^{x_i} p_i(u)du, \text{ and } p_i(x_i) = L_i'(x_i),$$
$$\text{subject to } \sum_{1}^{n} x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0,$$

where $L_i'(x_i)$ is given by Eq. (10).

4.2 Determining the CMLS Solution

To obtain the CMLS solutions to these problems, we formalize the Stochastic Environments that the LA are to interact with. From what we presented earlier in Section 3, the Stochastic Environments consists of the corresponding unit volume value functions $\mathcal{F}' = \{f_1'(x_1), f_2'(x_2), \ldots, f_n'(x_n)\}$, unknown to CMLS. We can analyze their properties by applying the principle of Lagrange multipliers to these maximization problems. After some simplification, one

[6] In any general application domain, one will not be able to observe $f_i'(x_i)$ directly. Rather, examining a potential solution may be the only way to reveal the success of the chosen allocation.

sees that the conditions that characterize the optimal solution for the exponential function of Eq. (9) are:

$$E_1'(x_1) = E_2'(x_2) = \cdots = E_n'(x_n)$$
$$\sum_1^n x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0.$$

Similarly, the conditions that characterize the optimal solution for the linear function of Eq. (10) are:

$$L_1'(x_1) = L_2'(x_2) = \cdots = L_n'(x_n)$$
$$\sum_1^n x_i = c \text{ and } \forall i \in \{1, \ldots, n\}, x_i \geq 0.$$

4.3 Experimental Set-up

In the interest of completeness, in this section, we report the results of comparing our learning scheme with the H-TRAA policy. We consider here the scenario when the data is synthetic. Although numerous experiments, for various settings, were conducted we briefly report, the results for a binary, quaternary and hexadecimal number of primitive materials. These were for the following environments:

- In the first case, the material unit value function was the exponential function given by Eq. (9).
- In the second case, the material unit value function was the linear function given by Eq. (10).

The results of our experiments are conclusive. They confirm the power of the CMLS.

In sections 4.4 and 4.5, we chose some typical values of the tuning parameters. For these values, an ensemble of 1,000 independent replications with different random number streams were performed to minimize the variance of the reported results. In the different experiments, we report the different values of the parameter $\lambda = 1 - \theta$ for the CMLS and the parameter N of the H-TRAA. In all the experiments, we assumed that $c = 1$.

In section 4.6, we rather perform a more thorough comparison by covering all the range of possible tuning parameters. We ran the experiment for 10^7 iterations and for every choice of the tuning parameters .

4.4 *Two-material* Case:

In this set of experiments we focus on the *two-material* case. We compare the CMLS with the H-TRAA under different learning parameters and under different environments (dynamic and static) and for the above-mentioned linear and exponential functions for the *two-material* case. In the interest of simplicity, we chose 6 typical values of λ and 6 typical values of the resolution N.

4.4.1 Static Environment

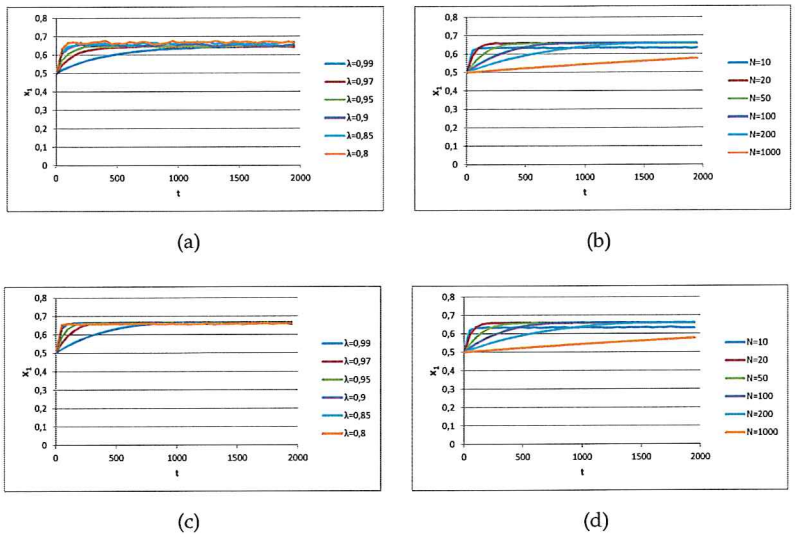

Fig. 1 This figure depicts the variation of $x_1(t)$ in a static environment for: (a) The CMLS scheme with different values of λ for the exponential function, (b) The H-TRAA for different values of the resolution N for the exponential function, (c) The CMLS scheme with different values of λ for the linear function, and (d) The H-TRAA for different values of the resolution N for the linear function.

We first report the simulation results for the exponential function. Figure 1(a) illustrates the evolution over time of $x_1(t)$ of the ensemble corresponding to a static environment for the CMLS under different values of λ, while Figure 1(b) illustrates the behavior for the H-TRAA. Interestingly enough, the convergence of both schemes exhibit similar properties. Similarly, we report the simulation results for the linear function in which Figure 1(c) illustrates the evolution of $x_1(t)$ for the CMLS, while Figure 1(d) illustrates the case of the H-TRAA. We focus on the rate of convergence by examining the time instant where the ensemble average of $x_1(t)$ exceeds the value 90% of the optimal value $x_1* = 0.66$ (where the $90\% x_1*$ values is 0.6) for some chosen values. These are given in Tables 1 and 2. From the tables, we can conclude that both the CMLS and the H-TRAA schemes are comparable for different values of λ and N respectively.

Table 2 reports the rates of convergence of the CMLS and the H-TRAA for the linear function for 90% value of x_1*. By examining Table 2, we observe that both the CMLS and the H-TRAA have comparable rates of convergence for the different values of λ and N respectively. For example, the rate of convergence of the CMLS with $\lambda = 0.85$ is 36 iterations, which is comparable to the corresponding H-TRAA's value for $N = 10$, i.e., 49 iterations. Similarly,

λ	Time	N	Time
0.85	28	10	28
0.9	32	20	55
0.95	67	50	143
0.97	101	100	280
0.99	307	200	591

Table 1 The rates of convergence of the CMLS and the H-TRAA for the exponential function for different values of λ and N respectively.

λ	Time	N	Time
0.85	36	10	49
0.9	42	20	62
0.95	97	50	147
0.97	144	100	271
0.99	497	200	596

Table 2 The rates of convergence of the CMLS and the H-TRAA for the linear function for different values of λ and N respectively.

the rate of convergence for the CMLS with $\lambda = 0.97$ of 144 iterations is comparable to the H-TRAA's for $N = 100$, namely, 271. Another obvious remark, is that as we increase N and λ, the rates of convergences of the H-TRAA and the CMLS are increased.

In addition, from Table 1 and Table 2, we observe that for our CMLS scheme, for the same value of λ, the convergence is faster for the linear function than the exponential function.

Figure 2(a), Figure 2(b), Figure 2(c) and Figure 2(d) depict the evolution of the variance over time in a static environment for: (a) The CMLS scheme with different values of λ in the case of an exponential function, (b) The H-TRAA for different values of the resolution N in the case of an exponential function, (c) The CMLS scheme with different values of λ in the case of a linear function, and (d) The H-TRAA for different values of the resolution N in the case of a linear function.

From both Figures 1(b) and 2(b), we observe that a large value of N yields a low variance, i.e., the accuracy is high. The problem is that the rate of convergence is slower than when we are using a lower value of N. A low value of N results in a faster convergence, but it yields a higher variance from the true underlying parameter. This confirms that the choice of the user-defined learning parameter, N, reduces to a trade off between the speed and the corresponding accuracy.

Similarly, from both Figures 1(a) and 2(a), we observe that a large value of λ yields low variance, i.e., the accuracy is high. The problem is that the rate of convergence is slower than when we are using a lower value of λ. A low value of λ results in a faster convergence, but it yields a higher variance from the true underlying parameter. Again, the choice of the user-defined learning

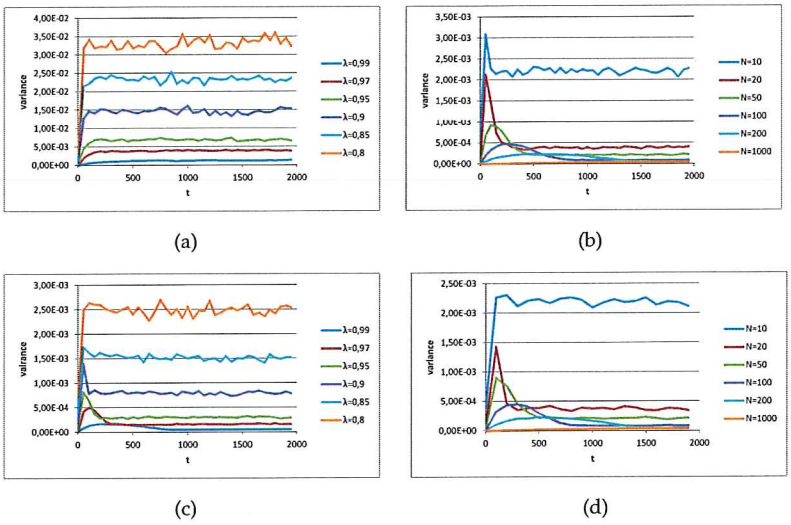

Fig. 2 This figure depicts the evolution of the variance over time in a static environment for: (a) The CMLS scheme with different values of λ in the case of an exponential function, (b) The H-TRAA for different values of the resolution N in the case of an exponential function, (c) The CMLS scheme with different values of λ in the case of a linear function, and (d) The H-TRAA for different values of the resolution N in the case of a linear function.

parameter, λ, reduces to a trade off between the speed and the corresponding accuracy.

4.4.2 Dynamic Environment

In this experiment, we investigate the behavior of the H-TRAA and the CMLS for the two-material case in a dynamic environment. The dynamic environment was modelled by periodically swapping the rank-index k of the updating probabilities for each fixed period. We report here the results for the environment that switches after every 500 and 250 iterations in Figures 3 and 4 respectively. In the interest of brevity, we will not comment on the results of the figures here because the same comments that we present later for the multi-material case (reported in Section 4.5.2) apply here too.

4.5 Multi-material Case

To consider the multi-material case, we examine the scenario for the convergence for four materials and for the case of the same two test functions. In the case of the experiments reported, the optimal vector x^* had the following components: $x_1^* = 12/25$, $x_2^* = 6/25$, $x_3^* = 4/25$, $x_4^* = 3/25$.

The results for the static and dynamic environments are given in the next two sub-sections.

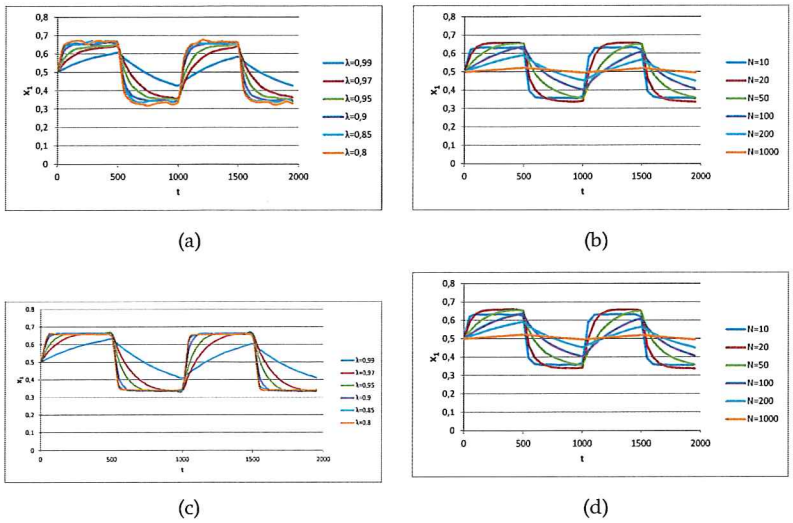

Fig. 3 This figure depicts the variation of $x_1(t)$ in a dynamic environment switching every 500^{th} iteration for: (a) The CMLS scheme with different values of λ for the exponential function, (b) The H-TRAA for different values of the resolution N for the exponential function, (c) The CMLS scheme with different values of λ for the linear function, and (d) The H-TRAA for different values of the resolution N for the linear function.

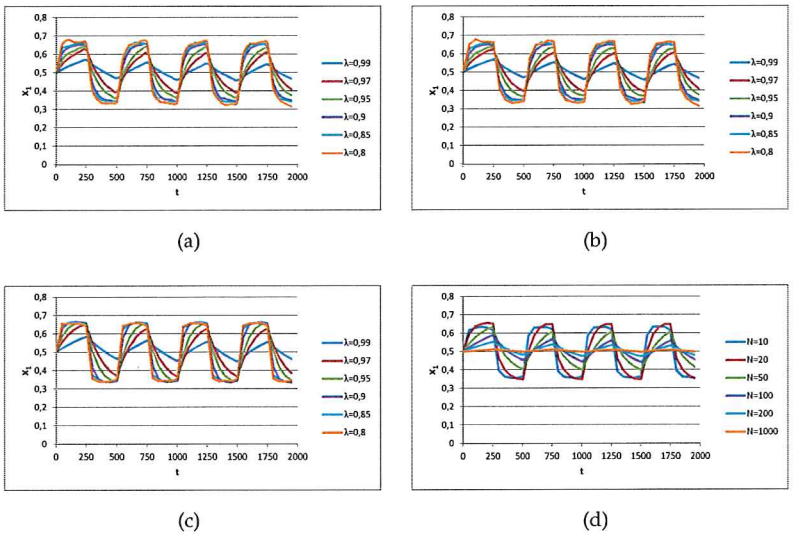

Fig. 4 This figure depicts the variation of $x_1(t)$ in a dynamic environment switching every 250^{th} iteration for: (a) The CMLS scheme with different values of λ for the exponential function, (b) The H-TRAA for different values of the resolution N for the exponential function, (c) The CMLS scheme with different values of λ for the linear function, and (d) The H-TRAA for different values of the resolution N for the linear function.

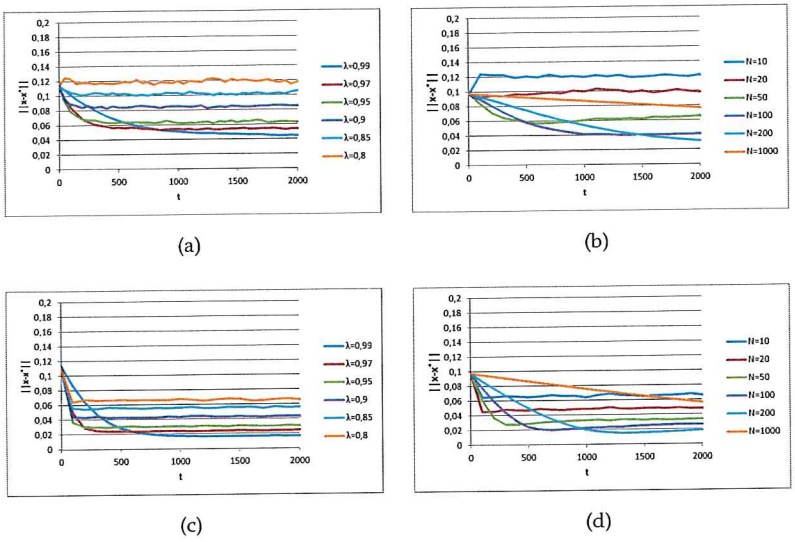

Fig. 5 The ensemble average of $||x - x^*||$ over time in a multi-material static environment for: (a) The CMLS scheme with different values of λ, in the case of an exponential function, (b) The H-TRAA for different values of the resolution N in the case of an exponential function, (c) The CMLS scheme with different values of λ in the case of a linear function, and (d) The H-TRAA for different values of the resolution N in the case of a linear function.

4.5.1 Static Environment

Figure 5(a), Figure 5(b), Figure 5(c) and Figure 5(d) illustrate the evolution ensemble average of $||x - x^*||$ over time in a static environment for: (a) The CMLS scheme with different values of λ, in the case of an exponential function, (b) The H-TRAA for different values of the resolution N in the case of an exponential function, (c) The CMLS scheme with different values of λ in the case of a linear function, and (d) The H-TRAA for different values of the resolution N in the case of a linear function.

Figure 6(a), Figure 6(b), Figure 6(c) and Figure 6(d) depict the evolution of the variance over time in a static environment for the same four cases.

In the case of the CMLS, we note that a small value for λ yields less accuracy, but a faster convergence. Similarly, for the H-TRAA, large values of N yield more accuracy but a slower convergence speed.

4.5.2 Dynamic Environment

To investigate the performance of the schemes for dynamic environments, we considered the case when the updating probabilities decreased with the rank-index k. In order to starve the H-TRAA and the CMLS schemes from this information, we opted to perturb the updating probabilities by permuting them using a circular permutation at each "environment switch". Thus,

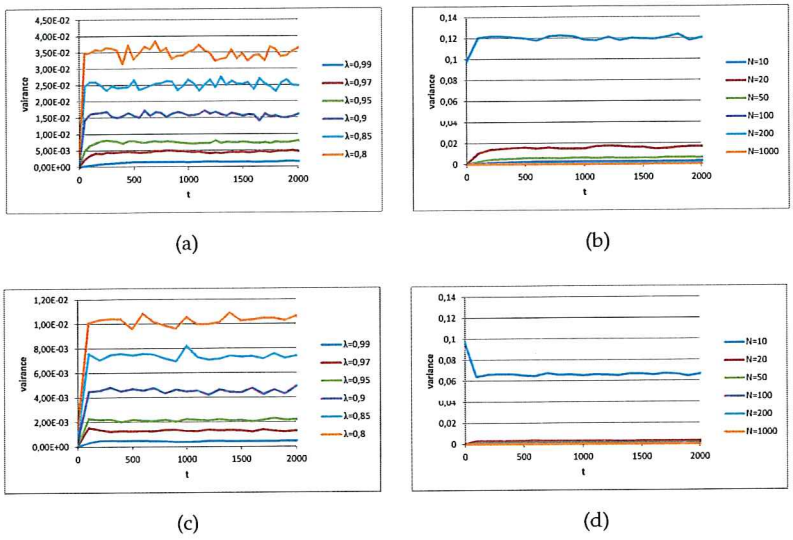

Fig. 6 The variance over time in a multi-material static environment for: (a) The CMLS scheme with different values of λ, in the case of an exponential function, (b) The H-TRAA for different values of the resolution N in the case of an exponential function, (c) The CMLS scheme with different values of λ in the case of a linear function, and (d) The H-TRAA for different values of the resolution N in the case of a linear function.

in this sense, between time instant 0 and 499, the environment was characterized by $x_1^* = 12/25$, $x_2^* = 6/25$, $x_3^* = 4/25$, $x_4^* = 3/25$, and after the environment switch, between instant 500 and 999, a circular permutation by one position was invoked yielding $x_1^* = 6/25$, $x_2^* = 4/25$, $x_3^* = 3/25$, $x_4^* = 12/25$.

We report the evolution of the ensemble average of $||x - x^*||$ over time in a dynamic environment switching each 500 iterations. Figure 7(a) yields the result of the CMLS in the case of the exponential function, Figure 7(b) yields the result of the H-TRAA in the case of the exponential function, Figure 7(c) displays the result of the CMLS in the case of the linear function, and Figure 7(d) shows the results of the H-TRAA in the case of the linear function.

The next figure presents the same results but for increased values of switching frequency. In fact, both schemes are handicapped and are not able to accurately track the optimal vector $x*$ for higher values of N and λ.

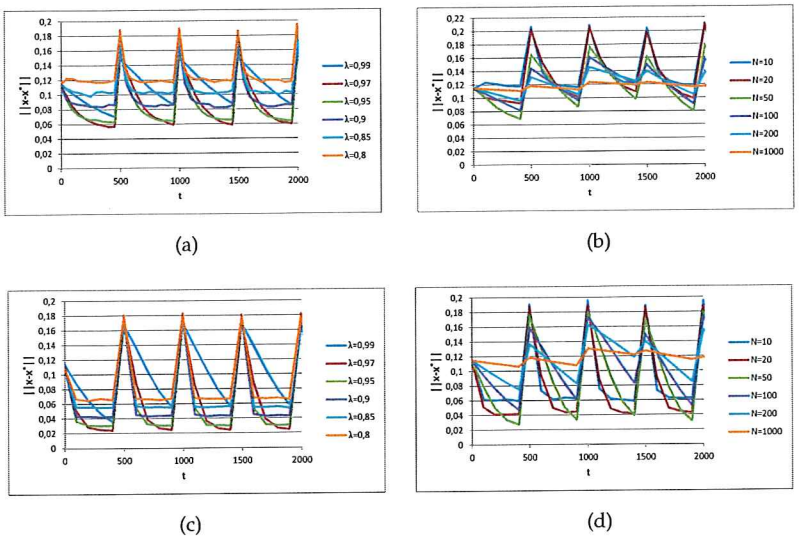

Fig. 7 This figure depicts the evolution of the ensemble average of $||x - x^*||$ over time in a multi-material dynamic environment switching that switches each 500 iterations for: (a) The CMLS scheme with different values of λ, in the case of an exponential function, (b) The H-TRAA for different values of the resolution N in the case of an exponential function, (c) The CMLS scheme with different values of λ in the case of a linear function, and (d) The H-TRAA for different values of the resolution N in the case of a linear function.

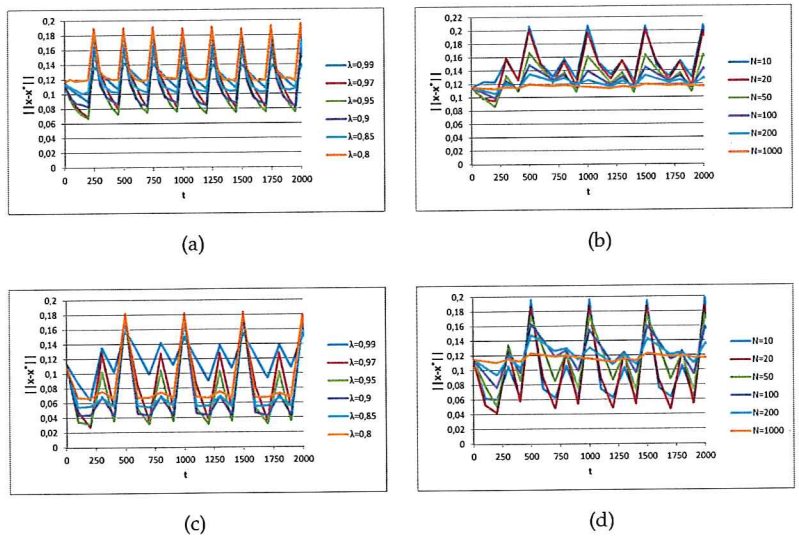

Fig. 8 This figure depicts the evolution of the ensemble average of $||x - x^*||$ over time in a multi-material dynamic environment switching that switches each 250 iterations for: (a) The CMLS scheme with different values of λ, in the case of an exponential function, (b) The H-TRAA for different values of the resolution N in the case of an exponential function, (c) The CMLS scheme with different values of λ in the case of a linear function, and (d) The H-TRAA for different values of the resolution N in the case of a linear function.

4.6 Comparison of performance for a variety of values of the tuning parameters

The results above give a good impression of the performance of the H-TRAA and CMLS algorithms. In this section, we do a more systematic experiment where the performance of the algorithms are compared for all possible choices of the tuning parameters. We consider the case of quaternary and hexadecimal number of primitive materials.

We consider the exponential and linear decay objective functions $E_i(x_i)$ and $L_i(x_i)$ as given in Eq. (6) and (8). We assume a dynamic system, which means that the reward probabilities for the different materials vary with time. More specifically, after a period of T iterations, the reward for the different materials are randomly switched. For example, let's assume a four materials case, and that the rewards for materials 1, 2, 3 and 4 are given by $E_1(x_1)$, $E_2(x_2)$, $E_3(x_3)$ and $E_4(x_4)$, respectively. After the period T, we randomly switch the reward functions such that the rewards for material 1, 2, 3 and 4 are given by e.g. $E_3(x_1), E_1(x_2), E_4(x_3)$ and $E_2(x_4)$, respectively. We consider three different cases for the period T between every switch:

- $T = 2000$ iterations. We refer to this scenario as SHORT below.
- $T = 2 \cdot 10^4$ iterations which is referred to as LONG below.
- We assume that T is a stochastic variable with possible outcomes $2000, 8000$ and $4 \cdot 10^4$. The probabilities are $P(T = 2000) = 20/26, P(T = 8000) = 5/26$ and $P(T = 4 \cdot 10^4) = 1/26$ which means that, in average, the estimation process spends an equal amount of time in each of the states 'fast' ($T = 2000$), 'medium' ($T = 8000$) and 'slow' ($T = 4 \cdot 10^4$). We refer to this scenario as RAND below.

The motivation for the RAND scenario, is that an algorithm should be able handle environments where the dynamics change arbitrarily with time, i.e. changing between slowly and rapidly varying. Naturally, the optimal values of tuning parameters of an algorithm depend on the dynamics of the environment, but ideally the performance of the algorithm should not be too sensitive with respect to these dynamics.

We compute estimation error using the root mean squared error (RMSE)

$$\text{RMSE} = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\frac{1}{R}\sum_{t=1}^{N}(\widehat{x}_{it} - x_{it})^2} \tag{11}$$

where R refers to the number of iterations while x_{it} refers the true and optimal amount of material i at iteration t and $\widehat{x}_{it}$ denotes the estimate. We compute the RMSE for each material and then take the average. To remove Monte Carlo error, we ran the experiment for $R = 10^7$ iterations for every case and for every choice of the tuning parameters in the algorithms.

The results are shown in Figures 9 and 10. For both the exponential and linear decay reward function (Figures 9 and 10, respectively) and for all the

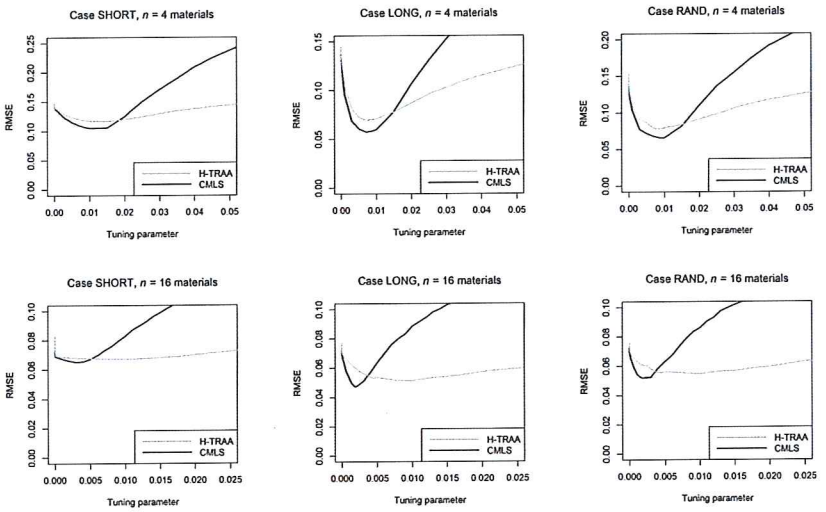

Fig. 9 Exponential decay reward case ($E_i(x_i)$): Estimation error (RMSE) for the H-TRAA and CMLS algorithms for a wide range of values of the tuning parameters. The tuning parameter on the x axis refers to the interval width (inverse of the leaf node resolution) for the H-TRAA algorithm and to θ for the CMLS algorithm. The top and bottom rows refer to cases with 4 and 16 materials, respectively. The columns from left to right refers to the cases SHORT, LONG and RAND, respectively.

six cases, we see that the CMLS algorithm outperforms the H-TRAA algorithm with respect to peak performance. For the H-TRAA, the optimal leaf node interval width seems to be about 0.01 and 0.02 for the exponential and linear cases, respectively, while for the CMLS, the optimal value of the θ decreases with the dimension n. For both the CMLS and the H-TRAA, the performance for the RAND case is about the same as for the LONG. This means that both algorithms perform good for a system where the dynamics change with time.

At this juncture, we shall draw some final remarks. There is a tradeoff between the speed of convergence to the optimal action and the accuracy. To conclude, it seems that in a dynamic environment, our scheme gives best results for values of the learning rate θ in the neighborhood of 0.01, corresponding to $\lambda = 1 - \theta$ in the neighborhood of 0.99. Such values do not only yield the peak performance in dynamic environments but also correspond to the fastest convergence rate. From the update equations of the CMLS specified by (1)–(2), we see that θ controls how much the probability vector during each iteration. When it comes to static environment, we observed that a value as small as 0.001 for θ is able to guarantee high accuracy. Similar conclusions and range of values are reported too in [43].

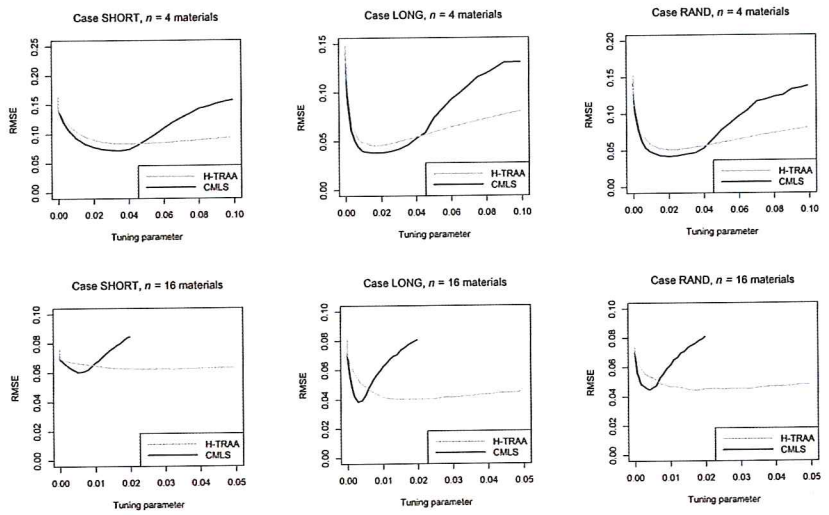

Fig. 10 Linear decay reward case ($L_i(x_i)$): Estimation error (RMSE) for the H-TRAA and CMLS algorithms for a wide range of values of the tuning parameters. The tuning parameter on the x axis refers to the interval width (inverse of the leaf node resolution) for the H-TRAA algorithm and to θ for the CMLS algorithm. The top and bottom rows refer to cases with 4 and 16 materials, respectively. The columns from left to right refers to the cases SHORT, LONG and RAND, respectively.

5 Conclusions and Further Work

In this paper, we have presented an optimal and efficient solution to Stochastic Non-linear Fractional Equality Knapsack (NFEK) Problem, which is a fundamental resource allocation problem based on incomplete and noisy information [7,8]. Unlike the existing solutions [7,8], our primary contribution is a *continuous* Learning Automata (LA)-based, *optimal*, efficient and yet simple solution to the NFEK problem. Our solution is distinct from the one reported in solutions [7,8] that uses multiple two-action discretized LA, organized in a hierarchical manner, so as to be able to tackle the case of multi-materials. Our solution does not need a hierarchical partitioning, and does not require us to maintain dedicated two-action discretized LA that allocate a given amount of resources among the two subsets.

Our LA solution is, rather, of a Reward-Inaction (R-I) type. Further, this has been modified in order to accommodate non-absorbing barriers. The proof of the convergence is based on Norman's theory for learning processes characterized by small learning steps [20, 22]. Our solution, termed as the Continuous Multi-action Learning Automata Solution (CMLS) to the Nonlinear Resource Allocation Problem, is easily implemented, and is, as such, a superior alternative to the state-of-the-art solution devised in [7,8].

The paper also contains numerous experimental results for static and dynamic environments that clearly demonstrate the efficiency of the CMLS scheme.

References

1. A. A. Al Islam, S. I. Alam, V. Raghunathan, and S. Bagchi. Multi-armed bandit congestion control in multi-hop infrastructure wireless mesh networks. In *2012 IEEE 20th International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS)*, pages 31–40. IEEE, 2012.
2. P. E. Black. Fractional knapsack problem. *Dictionary of algorithms and data structures*, 2004.
3. J. Chen and A. H. Sayed. Distributed pareto optimization via diffusion strategies. *IEEE Journal of Selected Topics in Signal Processing*, 7(2):205–220, 2013.
4. B. C. Dean, M. X. Goemans, and J. Vondrák. Approximating the stochastic knapsack problem: The benefit of adaptivity. *Mathematics of Operations Research*, 33(4):945–964, 2008.
5. M. Ghavipour and M. R. Meybodi. Trust propagation algorithm based on learning automata for inferring local trust in online social networks. *Knowledge-Based Systems*, 2017.
6. M. R. Gibson, J. W. Ohlmann, and M. J. Fry. An agent-based stochastic ruler approach for a stochastic knapsack problem with sequential competition. *Computers & Operations Research*, 37(3):598–609, 2010.
7. O.-C. Granmo and B. J. Oommen. Optimal sampling for estimation with constrained resources using a learning automaton-based solution for the nonlinear fractional knapsack problem. *Applied Intelligence*, 33(1):3–20, 2010.
8. O.-C. Granmo and B. J. Oommen. Solving stochastic nonlinear resource allocation problems using a hierarchy of twofold resource allocation automata. *IEEE Transactions on Computers*, 59(4):545–560, 2010.
9. O.-C. Granmo, B. J. Oommen, S. A. Myrer, and M. G. Olsen. Learning automata-based solutions to the nonlinear fractional knapsack problem with applications to optimal resource allocation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 37(1):166–175, 2007.
10. P. Kall. Approximation to optimization problems: An elementary review. *Mathematics of Operations Research*, 11(1):9–18, 1986.
11. J. Kangasharju, F. Hartanto, M. Reisslein, and K. W. Ross. Distributing layered encoded video through caches. *IEEE Transactions on Computers*, 51(6):622–636, 2002.
12. H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack problems*. Springer Science & Business Media, 2004.
13. S. Kosuch and A. Lisser. Upper bounds for the 0-1 stochastic knapsack problem and a b&b algorithm. *Annals of Operations Research*, 176(1):77–93, 2010.
14. G. Y. Lin, Y. Lu, and D. D. Yao. The stochastic knapsack revisited: Switch-over policies and dynamic pricing. *Operations Research*, 56(4):945–957, 2008.
15. K. Liu, Q. Zhao, and A. Swami. Dynamic probing for intrusion detection under resource constraints. In *Proceedings of IEEE International Conference on Communications, ICC 2013, Budapest, Hungary, June 9-13, 2013*, pages 1980–1984, 2013.
16. Y.-J. Liu, S. Lu, D. Li, and S. Tong. Adaptive controller design-based ablf for a class of nonlinear time-varying state constraint systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(7):1546–1553, 2017.
17. Y.-J. Liu, S. Lu, S. Tong, X. Chen, C. P. Chen, and D.-J. Li. Adaptive control-based barrier lyapunov functions for a class of stochastic nonlinear systems with full state constraints. *Automatica*, 87:83–93, 2018.
18. Z. Ma, H. Wang, K. Shi, and X. Wang. Learning automata based caching for efficient data access in delay tolerant networks. *Wireless Communications and Mobile Computing*, 2018, 2018.
19. M. Malboubi, L. Wang, C.-N. Chuah, and P. Sharma. Intelligent sdn based traffic (de) aggregation and measurement paradigm (istamp). In *2014 Proceedings IEEE INFOCOM*, pages 934–942. IEEE, 2014.
20. K. S. Narendra and M. A. L. Thathachar. *Learning automata: an introduction*. Courier Corporation, 2012.

21. O. V. Nedzelnitsky and K. S. Narendra. Nonstationary models of learning automata routing in data communication networks. *IEEE Transactions on Systems, Man and Cybernetics*, 17(6):1004–1015, 1987.
22. M. F. Norman. *Markov processes and learning models*, volume 84. Academic Press New York, 1972.
23. M. F. Norman. Markovian learning processes. *SIAM Review*, 16(2):143–162, 1974.
24. B. J. Oommen. Absorbing and ergodic discretized two action learning automata. *IEEE Transactions on Systems, Man and Cybernetics*, 16:282–293, March/April 1986.
25. D. Panagou, D. M. Stipanović, and P. G. Voulgaris. Distributed coordination control for multi-robot networks using lyapunov-like barrier functions. *IEEE Transactions on Automatic Control*, 61(3):617–632, 2016.
26. S. Pandey, K. Ramamritham, and S. Chakrabarti. Monitoring the dynamic web to respond to continuous queries. In *Proceedings of the 12th international conference on World Wide Web*, pages 659–668. ACM, 2003.
27. G. I. Papadimitriou and D. G. Maritsas. Learning automata-based receiver conflict avoidance algorithms for wdm broadcast-and-select star networks. *IEEE/ACM Transactions on Networking (TON)*, 4(3):407–412, 1996.
28. G. I. Papadimitriou and A. S. Pomportsis. Learning-automata-based tdma protocols for broadcast communication systems with bursty traffic. *IEEE Communications Letters*, 4(3):107–109, 2000.
29. T. C. Perry and J. C. Hartman. An approximate dynamic programming approach to solving a dynamic, stochastic multiple knapsack problem. *International Transactions in Operational Research*, 16(3):347–359, 2009.
30. A. S. Poznyak and K. Najim. Learning through reinforcement for n-person repeated constrained games. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 32(6):759–771, 2002.
31. A. Rezvanian and M. R. Meybodi. Sampling algorithms for stochastic graphs: a learning automata approach. *Knowledge-Based Systems*, 127:126–144, 2017.
32. A. Rezvanian, A. M. Saghiri, S. M. Vahidipour, M. Esnaashari, and M. R. Meybodi. *Recent Advances in Learning Automata*, volume 754. Springer, 2018.
33. A. Rezvanian, S. M. Vahidipour, and M. Esnaashari. New applications of learning automata-based techniques in real-world environments. *Journal of Computational Science*, 24:287 – 289, 2018.
34. K. W. Ross. The stochastic knapsack. In *Multiservice Loss Models for Broadband Telecommunication Networks*, pages 17–70. Springer, 1995.
35. J. Sachs. A stochastic knapsack model for the capacity evaluation of (multi-) radio access networks. *Systems Science*, 32:79–87, 2006.
36. A. M. Saghiri and M. R. Meybodi. Open asynchronous dynamic cellular learning automata and its application to allocation hub location problem. *Knowledge-Based Systems*, 139:149–169, 2018.
37. D. Serre. *Matrices: Theory and Applications*. Graduate texts in mathematics. Springer, 2002.
38. S. H. Seyyedi and B. Minaei-Bidgoli. Estimator learning automata for feature subset selection in high-dimensional spaces, case study: Email spam detection. *International Journal of Communication Systems*, 2018.
39. P. R. Srikantakumar and K. S. Narendra. A learning model for routing in telephone networks. *SIAM Journal on Control and Optimization*, 20(1):34–57, 1982.
40. E. Steinberg and M. S. Parks. A preference order dynamic program for a knapsack problem with stochastic rewards. *Journal of the Operational Research Society*, pages 141–147, 1979.
41. M. L. Tsetlin. *Automaton theory and modeling of biological systems*. Academic Press, New York, 1973.
42. T. Wauters, K. Verbeeck, P. De Causmaecker, and G. V. Berghe. A learning-based optimization approach to multi-project scheduling. *Journal of Scheduling*, 18(1):61–74, 2015.
43. T. Wauters, K. Verbeeck, P. De Causmaecker, and G. V. Berghe. A learning-based optimization approach to multi-project scheduling. *Journal of Scheduling*, 18(1):61–74, 2015.
44. J. L. Wolf, M. S. Squillante, P. S. Yu, J. Sethuraman, and L. Ozsen. Optimal crawling strategies for web search engines. In *Proceedings of the 11th international conference on World Wide Web*, pages 136–147. ACM, 2002.