2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Absorption in Model-based Search Algorithms for Combinatorial Optimization

Zijun Wu and Michael Kolonko
Institut für Angewandte Stochastik und Operations Research,
TU Clausthal, Erzstr.1, D-38678 Clausthal-Zellerfeld, Germany

Email: zwu10@tu-clausthal.de, kolonko@math.tu-clausthal.de

***Abstract*—Model-based search is an abstract framework that unifies the main features of a large class of heuristic procedures for combinatorial optimization, it includes ant algorithms, cross entropy and estimation of distribution algorithms. Properties shown for the model-based search therefore apply to all these algorithms.**

A crucial parameter for the long term behavior of model-based search is the learning rate that controls the update of the model when new information from samples is available. Often this rate is kept constant over time. We show that in this case after finitely many iterations, all model-based search algorithms will be absorbed into a state where all samples consist of a single solution only. Moreover, it cannot be guaranteed that this solution is optimal, at least not when the optimal solution is unique.

I. Introduction

Heuristic optimization algorithms are often divided into the so-called model-based approaches and the solution-based approaches. In [1] the model-based approach was given a more formal definition as *model-based search* (MBS). This is a very general framework for heuristic combinatorial optimization procedures that claims to cover the essential parts e.g. of ant algorithms (Ants, see [2]), cross entropy algorithm (CE, see [3]), estimation of distribution algorithms (EDAs, see [4]), classical stochastic gradient ascent [5] or population-based incremental learning [6]. Not model-based in this sense are e.g. some population-based procedures like genetic algorithms, local search methods and tabu-search.

Generally, MBS concentrates on a 'model' which is a probability distribution on the set of solutions of a combinatorial optimization problem. The aim is to find a distribution that gives high probability to optimal solutions. More precisely, MBS proceeds by repeating the following two steps in each iteration $t = 0, 1, \dots$:

Sampling Draw a random sample $\boldsymbol{X}_t$ from the present model $\boldsymbol{\Pi}_t$, which is from some family $\mathbb{P}$ of distributions on the solution space;

Adaptation Construct $\boldsymbol{\Pi}_{t+1} \in \mathbb{P}$ using $\boldsymbol{\Pi}_t$, $\boldsymbol{X}_t$ and some auxiliary memory $\mathcal{M}_t$ in order to bias future sampling toward good solutions.

The 'Adaptation' step is crucial for MBS. In this paper we assume that this step consists of the following two sub-steps:

A1 'Learn' a distribution $\boldsymbol{W}_t \in \mathbb{P}$ from $\boldsymbol{X}_t$ and $\mathcal{M}_t$ using a *learning method*, update the memory to $\mathcal{M}_{t+1}$;

A2 construct $\boldsymbol{\Pi}_{t+1} \in \mathbb{P}$ from $\boldsymbol{\Pi}_t$ and $\boldsymbol{W}_t$ using an *update rule*.

As was pointed out in [1], many formulations of Ants, CE and EDAs have this structure.

In this paper we assume that the MBS has an update rule of the form

$$\boldsymbol{\Pi}_{t+1} = (1 - \varrho_{t+1})\boldsymbol{\Pi}_t + \varrho_{t+1}\boldsymbol{W}_t, \qquad \text{(I.1)}$$

where ϱ_{t+1} is a *learning rate* which reflects the relative importance of the sample based distribution $\boldsymbol{W}_t$ in iteration t. Often this learning rate is chosen to be a constant $\varrho \in (0, 1)$, see e.g. implementations of CE [7] and Ants [8] where the learning rate is called *smoothing parameter* and *evaporation rate* respectively.

In genetic algorithms the phenomenon of *genetic drift* is well-known, see e.g. [9]. It describes the loss of all variation in the solutions produced until finally the process is absorbed into a single solution. A similar thing may also happen in MBS. Under fairly weak conditions on the 'Adaptation' step, all solutions seen in the 'Sampling' step of MBS will be identical after finitely many iterations. This will happen if the learning rate is bounded away from 0 (which is fulfilled if ϱ_t is a constant) and if in some sense the influence of the memory is limited (see assumptions (III.3) - (III.5) below). These conditions are fulfilled e.g. in [7] and [8], therefore absorption into a single solution after finitely many iterations will occur in these models.

Moreover, we can show that the solution finally reached need not be optimal, in fact absorption and the almost sure reachability of optimal solutions are mutually exclusive at least if the optimal solution is unique. In particular, with a fixed learning rate we cannot be sure to ever reach the optimal solution. This indicates that one should avoid constant learning rates in practice, but for a clear understanding, a more thorough analysis of the finite time behavior is needed which is not the topic of the present paper.

In a former paper [10], we showed how absorption occurs in a cross entropy algorithm, the present paper extends this result to the more general class of MBS

This paper is arranged as follows: In Section II, we describe in detail the problem encoding and the extension of

The work of Z. Wu was supported by China Scholarship Council, No.2010624003.

the MBS algorithm we are using. In Section III, we present our main results and discuss their implications. For better readability, the lengthy proofs of our results are given in section IV. In Section V, we give a short summary.

II. The Mathematical Model

A. Problem Encoding and Feasibility Concept

We are considering a combinatorial minimization problem (S, f) where S represents the finite set of *feasible solutions*, and f is the *cost function*. Let $S^* \subset S$ denote the collection of *optimal solutions*.

We assume that there is a finite set $\mathcal{A} := \{a_1, \ldots, a_K\}$ of symbols (or components) from which the solutions are built, i.e. $S \subset \mathcal{A}^L$ for some fixed length L. Each solution $s \in S$ can therefore be written as a finite string $s = (s_1, \ldots, s_L)$ of components from $\mathcal{A}$. As is well known, any combinatorial optimization problem can be represented in such a way, in fact $\mathcal{A} = \{0, 1\}$ is sufficient though often not very efficient.

A particular feature of our model is the feasibility concept that allows to introduce some dependence between the components of a solution. We assume that we are given a *feasibility function* $C_i(y, a)$ that for every possible partial solution y of length i assigns a weight to each $a \in \mathcal{A}$. If $C_i(y, a) = 0$, the partial solution y cannot be continued by adding the symbol a. The larger the value of $C_i(y, a)$, the more desirable it seems from a greedy point of view to continue with symbol a. We assume that these values are normed, so that $C_i(y, \cdot)$ is a probability on the set $\mathcal{A}$.

As an example, let the solution s represent a tour in a traveling salesman problem, then $C_i(y, \cdot)$ can be used to prevent loops if $C_i(y, a) = 0$ for all cities a already visited in the partial tour y. Also, $C_i(y, a)$ could be defined as the relative distance from the end of the partial tour y to a feasible city $a \in C_i(y)$, relative to the sum of all distances to $a' \in C_i(y)$. This is sometimes called 'visibility' in ant algorithms.

With the help of feasibility distributions, we can formally define the set R_i of *feasible partial solutions* of length i recursively as follows: Let the empty string $\diamond = ()$ denote the feasible solution of length 0 and put $R_0 = \{\diamond\}$. Assume that we have defined R_i for some $i \in \{0, \ldots, L-1\}$ and that we are given a probability distribution $C_i(y, \cdot)$ on the set $\mathcal{A}$ for each $y \in R_i$. Then we define

$$R_{i+1} := \big\{(y, a) \mid y \in R_i, a \in \mathcal{A}, C_i(y, a) > 0\big\}$$

where (y, a) denotes the concatenation of the partial solution y and a, in particular we have $(\diamond, a) = (a)$. The set of feasible solution is now defined as $S := R_L$. For $i \in I := \{0, \ldots, L-1\}$ and $y \in R_i$ let

$$C_i(y) := \{a \in \mathcal{A} \mid C_i(y, a) > 0\}$$

be the *support* of $C_i(y, \cdot)$.

This feasibility concept allows to include the unconstrained case $S = \mathcal{A}^L$, in this case $C_i(\cdot, \cdot)$ is chosen as the uniform distribution on $\mathcal{A}$, i.e. $C_i(y, a) \equiv \frac{1}{|\mathcal{A}|}$. Also the case with constraints but no degree of desirability can be included: if for a given partial solution $y \in R_i$, the choice of the next component is restricted to a set $A \subset \mathcal{A}$, then we choose $C_i(y, \cdot)$ as the uniform distribution on A and have $C_i(y) = A$.

B. Definition of the MBS Algorithm

We now formalize the elements occurring in **A1** and **A2**. As mentioned before, we concentrate on MBS using update rule (I.1).

Let $\mathbb{P}(\mathcal{A})$ denote the set of all probability measures on the set $\mathcal{A}$ and put $\mathbb{P} := \mathbb{P}(\mathcal{A})^L = \mathbb{P}(\mathcal{A}) \times \cdots \times \mathbb{P}(\mathcal{A})$. Then $\boldsymbol{p} = (\boldsymbol{p}(1), \ldots, \boldsymbol{p}(L)) \in \mathbb{P}$ is a product probability measure on $\mathcal{A}^L$ that describes the selection of a solution $s = (s_1, \ldots, s_L) \in \mathcal{A}^L$ where the L symbols $s_1, \ldots, s_L$ are chosen independently of each other. Here, $\boldsymbol{p}(i) = \big(\boldsymbol{p}(a; i)\big)_{a \in \mathcal{A}} \in \mathbb{P}(\mathcal{A})$ is the distribution for the symbol on the i-th position of the string. In this work, we use $\mathbb{P}$ as the underlying family of models. Note that these distributions cannot capture any interaction between the components and need not be restricted to the feasible solutions S.

The distribution $\boldsymbol{W}_t$ learned in step **A1** above takes values in $\mathbb{P}$. Given a sample $\boldsymbol{X}_t = \boldsymbol{x}$ and a memory content $\mathcal{M}_t = \boldsymbol{m}$, its value $\boldsymbol{W}_t = \mathcal{L}(X_t, \mathcal{M}_t)$ is determined by a *learning function*

$$\mathcal{L}_t(\boldsymbol{x}, \boldsymbol{m}) := \big(\mathcal{L}_t(1, \boldsymbol{x}, \boldsymbol{m}), \ldots, \mathcal{L}_t(L, \boldsymbol{x}, \boldsymbol{m})\big),$$

where

$$\mathcal{L}_t(i, \boldsymbol{x}, \boldsymbol{m}) := \big(\mathcal{L}_t(a; i, \boldsymbol{x}, \boldsymbol{m})\big)_{a \in \mathcal{A}}$$

is a distribution on $\mathcal{A}$. Examples for $\mathcal{L}_t(\boldsymbol{x}, \boldsymbol{m})$ include the empirical distribution based on a selection of a subsample $\mathcal{N}(\boldsymbol{x}, \boldsymbol{m})$ from the sample $\boldsymbol{x}$ and the memory $\boldsymbol{m}$:

$$\mathcal{L}_t(a; i, \boldsymbol{x}, \boldsymbol{m}) = \frac{1}{|\mathcal{N}(\boldsymbol{x}, \boldsymbol{m})|} \sum_{s \in \mathcal{N}(\boldsymbol{x}, \boldsymbol{m})} \mathbb{1}_{\{a\}}(s_i),$$

where $\mathbb{1}_A$ denotes the indicator function of a set A. The selection of $\mathcal{N}(\boldsymbol{x}, \boldsymbol{m})$ is typically biased towards the good solutions, it may contain additional randomness or weighting of the solutions, see e.g. [3], [7], [11] or [12].

In the sequel, we shall not use details of the learning method or the content and update of the memory. Instead we concentrate on the effects that the present sample has on the resulting distribution $\boldsymbol{W}_t$, see assumptions (III.3)–(III.5) below

We can now formally define the MBS. Besides the sets R_i and S of (partial) feasible solutions, it requires the following items as input:

- the feasibility distributions $C_i(\cdot, \cdot), i = 0, \ldots, L-1$;
- a sequence $(\varrho_t)_{t \geq 1}$ with $\varrho_t \in [0, 1]$ of learning rates;
- a sample size $N > 0$;
- a starting distribution $\boldsymbol{p}_0 \in \mathbb{P}$;
- a sequence of learning functions $\mathcal{L}_t$.

Model-based Search Algorithm

Starting: For time-step $t = 0$, put $\boldsymbol{p} := \boldsymbol{p}_0$, let the memory $\boldsymbol{m}$ be empty and then run through the following steps for $t = 0, 1, \dots$ until some stopping criterion is fulfilled.

Sampling: If the present distribution is $\boldsymbol{p} \in \mathbb{P}$, feasible solutions $s = (s_1, \dots, s_L) \in S$ are drawn according to the probability

$$Q(s, \boldsymbol{p}) := Q(s_1; 1, \diamond, \boldsymbol{p}) \cdot \prod_{i=2}^{L} Q(s_i; i, (s_1, \dots, s_{i-1}), \boldsymbol{p}) \tag{II.1}$$

where

$$Q(a; i, y, \boldsymbol{p}) := \frac{\boldsymbol{p}(a; i) C_{i-1}(y, a)}{\sum_{a' \in \mathcal{A}} \boldsymbol{p}(a'; i) C_{i-1}(y, a')} \tag{II.2}$$

is the probability that the feasible symbol $a \in \mathcal{A}$ is added at position i to the feasible partial solution $y \in R_{i-1}$. We use the convention $\frac{0}{0} = 0$ throughout this paper. Note that $Q(\cdot, \boldsymbol{p})$ is concentrated on the feasible solutions S even if $\boldsymbol{p}$ is not.

In this way a sample $\boldsymbol{x} = (s^{(1)}, \dots, s^{(N)})$ of size N is drawn independently and identically distributed (i.i.d.).

Learning: The learning function $\mathcal{L}_t$ is applied to this sample $\boldsymbol{x}$ and the present memory $\boldsymbol{m}$ yielding a distribution $\boldsymbol{w} := \mathcal{L}_t(\boldsymbol{x}, \boldsymbol{m})$ in $\mathbb{P}$. The memory is updated according to some rule.

Update: The present distribution $\boldsymbol{p}$ is updated as a convex combination of $\boldsymbol{p}$ and the new sample-based distribution $\boldsymbol{w}$:

$$\boldsymbol{p} := (1 - \varrho_{t+1})\boldsymbol{p} + \varrho_{t+1}\boldsymbol{w}. \tag{II.3}$$

Next, the counter t is increased by 1 and the step 'Sampling' is performed with the new $\boldsymbol{p}$.

A schematic overview of the algorithm is given in the Figure below. This definition of MBS extends the algorithm as defined in [1] with respect to feasibility and, as was pointed out before, covers the essential parts of many popular iterative stochastic search procedures.

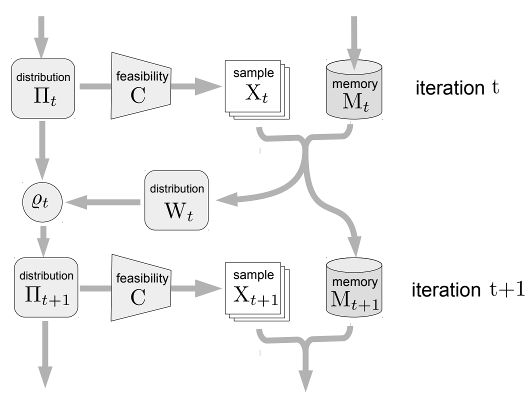

Fig. II.1. A schematic diagram of the MBS

C. Underlying Stochastic Processes

Applying the MBS results in a stochastic process

$$\Big(\boldsymbol{\Pi}_t, \boldsymbol{X}_t, \mathcal{M}_t, \boldsymbol{W}_t\Big)_{t=0,1,\dots}$$

where $\boldsymbol{\Pi}_t = \big(\boldsymbol{\Pi}_t(a; i)\big)_{i=1,\dots,L, a \in \mathcal{A}}$ is a random variable taking on values in $\mathbb{P}$ describing the modeling distribution underlying the sampling in the t-th iteration with $\boldsymbol{\Pi}_0 = \boldsymbol{p}_0$. $\boldsymbol{X}_t$ takes on values in the finite set S^N and is the sample of N solutions produced in the t-th 'Sampling' step of the algorithm using $Q(\cdot, \boldsymbol{\Pi}_t)$ as defined in (II.1). $\mathcal{M}_t$ is the content of the memory and $\boldsymbol{W}_t$ is the distribution from the learning step as a random variable with values in $\mathbb{P}$.

We write $\boldsymbol{X}_t = \big(\boldsymbol{X}_t^{(1)}, \dots, \boldsymbol{X}_t^{(N)}\big)$ and $\boldsymbol{X}_t^{(l)} = \big(\boldsymbol{X}_t^{(l)}(1), \dots, \boldsymbol{X}_t^{(l)}(L)\big)$. Then $\boldsymbol{X}_t^{(n)}(i)$ denotes the symbol at position $i, 1 \le i \le L$, in the n-th solution sampled in iteration t. $\boldsymbol{X}_t^{(n)}(1, \dots, i)$ denotes the partial solution up to the i-th position in that solution. Similarly, $\boldsymbol{W}_t = (\boldsymbol{W}_t(1), \dots, \boldsymbol{W}_t(L))$ and $\boldsymbol{W}_t(i) = \Big(\boldsymbol{W}_t(a; i)\Big)_{a \in \mathcal{A}}$. Then $\boldsymbol{W}_t(a; i)$ is the probability, that symbol a appears in position i in the distribution learned in the t-th iteration.

The complete transition probabilities of this process can be derived from the definition of the MBS. We give only the most important one that is used in the proofs. Let $\mathcal{H}_t := \big(\boldsymbol{\Pi}_m, \boldsymbol{X}_m, \mathcal{M}_m, \boldsymbol{W}_m\big)_{m=1,\dots,t}$ denote the history up to time t, then

$$\begin{aligned} &\mathbf{P}\Big[\boldsymbol{X}_t^{(n)}(i+1) = a \mid \boldsymbol{\Pi}_t = \boldsymbol{p}, X_t^{(n)}(1, \dots, i) = y, \mathcal{H}_{t-1}\Big] \\ &= Q(a;\, i+1, y, \boldsymbol{p}) \end{aligned} \tag{II.4}$$

for $a \in \mathcal{A}, y \in R_i$, $i \in I$, $n = 1, \dots, N$, $t \in \mathbb{N}$ and $\boldsymbol{p} \in \mathbb{P}$.

Due to the deterministic nature of the update mechanism we may also write

$$\begin{aligned} &\boldsymbol{\Pi}_{t+1}(a;\, i) \\ &\qquad = (1 - \varrho_{t+1})\boldsymbol{\Pi}_t(a;\, i) + \varrho_{t+1}\boldsymbol{W}_t(a;\, i). \end{aligned} \tag{II.5}$$

for $a \in \mathcal{A}, i = 1, \dots, L$ and $t \ge 0$. We shall refer to (II.5) as the 'basic recursion'.

III. Main Results

A. General Assumptions

Throughout this paper we assume that the starting distribution $\boldsymbol{\Pi}_0 = \boldsymbol{p}_0$ is given such that any item from $\mathcal{A}$ has a positive probability at any position i :

$$\boldsymbol{p}_0(a; i) > 0 \qquad \text{for all } a \in \mathcal{A}, i = 1, \dots, L. \tag{III.1}$$

Note, that in practice we often use an uniform distribution on $\mathcal{A}^L$ as $\boldsymbol{p}_0$. Also, without loss in generality we may assume that there are at least two feasible symbols $a \ne a'$ for the first position, i.e.

$$C_0(\diamond, a) > 0 \quad \text{and} \quad C_0(\diamond, a') > 0. \tag{III.2}$$

Otherwise, all feasible solutions would start with the same symbol, which could then be dropped from the encoding of the solutions.

Finally, we have to give conditions on the learning process. Remember that $\boldsymbol{W}_t = \mathcal{L}_t(\boldsymbol{X}_t, \mathcal{M}_t)$ is the probability distribution learned from the present sample $\boldsymbol{X}_t$ and the present memory content $\mathcal{M}_t$. $\boldsymbol{W}_t$ is needed to update the distribution for the next sample as in (II.5). Without fixing any details of the learning function $\mathcal{L}_t$ we assume that the resulting dependence between the present sample $\boldsymbol{X}_t$ and the distribution $\boldsymbol{W}_t$ is as follows: for any $i \in \{1, \dots, L\}$, $a \in \mathcal{A}$ and $t \geq 0$:

$$\text{if } \boldsymbol{X}_t^{(n)}(i) \neq a \text{ for all } n = 1, \dots, N \quad \text{then } \boldsymbol{W}_t(a; i) = 0, \tag{III.3}$$

$$\text{if } \boldsymbol{X}_t^{(n)}(i) = a \text{ for all } n = 1, \dots, N \quad \text{then } \boldsymbol{W}_t(a; i) = 1, \tag{III.4}$$

and there is a constant $\alpha \in (0, 1)$ such that

$$0 < \boldsymbol{W}_t(a; i) < 1 \ \Rightarrow \ \alpha < \boldsymbol{W}_t(a; i) < 1 - \alpha. \tag{III.5}$$

Here α is assumed to be independent of $\boldsymbol{X}_t$. It essentially means that if in the present sample a symbol a does not appear on position i, then the value $\boldsymbol{\Pi}_{t+1}(a, i)$ is not reinforced in (II.5), and if a appears on i in *all* sampled solutions then $\boldsymbol{\Pi}_{t+1}(a; i)$ is increased by the maximal amount ϱ_t. Finally, if $\boldsymbol{W}_t(a; i) > 0$, i.e. if a appears at least once in the sample on position i then $\boldsymbol{W}_t(a; i)$ must be larger than a fixed value $\alpha > 0$. These conditions generalize properties of empirical distributions obtained from the sample $\boldsymbol{X}_t$. The examples of learning functions given above will therefore fulfill these conditions if the memory does not override the effect of the sample. If we only use the best solution sampled so far for the update, then (III.3) and (III.4) need not hold. This popular learning function is therefore excluded from our model.

B. Main Results

We say that an MBS algorithm is *absorbed* almost surely if with probability one there is an iteration $T < \infty$ and a solution $s \in S$ such that for all $m \geq T$ and all $n \in \{1, \dots, N\}$ we have $\boldsymbol{X}_m^{(n)} = s$, that is, the whole sample consists of N identical solutions from some iteration T onwards.

The following Theorem implies that, under a constant learning rate $\varrho_t \equiv \varrho > 0$, the MBS will be absorbed, thus all algorithms that are covered by the MBS are affected by this behavior. As we are dealing with random variables, formal assertions always hold with probability one only allowing exceptions on a set of probability 0. To be mathematically correct, we therefore added 'almost surely' wherever appropriate.

Theorem 1. If $\varrho_t \geq \varrho > 0$ for all $t \geq 1$ and some constant ϱ, then the MBS proposed in section II-B becomes absorbed almost surely.

Technically, Theorem 1 states that the process $\boldsymbol{X}_t$ becomes absorbed into a state (=sample) $(s, \dots, s)$ in the finite state space S^N almost surely. It extends well-known results from time-homogeneous Markov chains as they are used to model drift in simple genetic algorithms to the inhomogeneous case we are considering here. The proof of the Theorem proceeds by induction on the length $i \in I = \{0, \dots, L-1\}$ of partial solutions for which absorption to a $y \in R_i$ has already occurred. The main induction step is contained in the following Lemma

Lemma 2. Assume that for a fixed $i \in I$ the following holds almost surely: there is a finite time $T < \infty$ and a $y \in R_i$ such that

$$\boldsymbol{X}_{T+m}^{(n)}(1, \dots, i) = y \tag{III.6}$$

for all $m \geq 0$ and for all solutions $n = 1, \dots, N$.

Then, if $\varrho_t \geq \varrho > 0$ for all $t \geq 1$ and some constant ϱ, the following will also hold almost surely: there are $T' < \infty$ and $a_0 \in \mathcal{A}$ such that

$$\boldsymbol{X}_{T'+m}^{(n)}(i+1) = a_0 \tag{III.7}$$

for all $m \geq 0$ and $n = 1, \dots, N$.

Lemma 2 says that if from some time T on all solutions sampled coincide in the first i positions then this will also be the case for the next position $i+1$ after finitely many additional steps. Now the proof of Theorem 1 follows by observing that (III.6) holds for $i = 0$ and the empty partial solution $y = \diamond$ as all solutions have to start with the empty string. The proof of Lemma 2 is given in section IV-B.

As a Corollary to the Theorem 1 we get

Theorem 3. If $\varrho_t \geq \varrho > 0$ for all $t \geq 1$ and some constant ϱ, then the density $\boldsymbol{\Pi}_t$ converges almost surely against a one point mass concentrated on the solution the $\boldsymbol{X}_t$-process is absorbed into.

For a proof of Theorem 3, see section IV-C. It shows that absorption of the $\boldsymbol{X}_t$-process implies convergence of the $\boldsymbol{\Pi}_t$-process. However, as was pointed out in [10], convergence of the $\boldsymbol{\Pi}_t$-process does not necessarily imply absorption of the $\boldsymbol{X}_t$-process.

In practice absorption would not be bad if we could be sure that the solution we end up with is optimal, but this cannot be guaranteed. The following Theorem shows that for the case of a unique optimal solution s^*, the probability to reach s^* in finite time is smaller than one if the learning rate is constant. Hence, with a positive probability, we may be absorbed into a fixed solution before we have reached the optimal one.

Let τ be the first iteration t in which s^* occurs in $\boldsymbol{X}_t$. Then *almost sure reachability* of the optimal solution holds if $\mathbf{P}(\tau < \infty) = 1$.

Theorem 4. Assume $|S^*| = 1$, and $\varrho_t \in (0, 1)$ for all $t \geq 1$, then absorption of $\boldsymbol{X}_t$ implies $\mathbf{P}(\tau < \infty) < 1$.

For a proof of Theorem 4, see section IV-D, where it is shown that almost sure reachability of the optimal solution and absorption of $\boldsymbol{X}_t$ are mutually exclusive.

In the above statements it was (tacitly) assumed that $\varrho_t \in (0, 1)$. In fact one can show that absorption of both $\boldsymbol{\Pi}_t$ and $\boldsymbol{X}_t$ still hold if $\varrho_t = 1$ for some t or even $\varrho_t \equiv 1$ and also that absorption of $\boldsymbol{X}_t$ then implies $\mathbf{P}(\tau < \infty) < 1$. As soon as $\varrho_t = 1$ has occurred, the algorithm forgets its former model $\boldsymbol{\Pi}_t$ and starts anew with $\boldsymbol{p}_0 := \boldsymbol{W}_t$. Hence the assertions then hold with S replaced by the subset S' of solutions that have positive weight in $\boldsymbol{W}_t$. To keep things simple, we assume in the sequel that $\varrho_t \in (0, 1)$ for each $t \geq 1$ and $\varrho \in (0, 1)$.

IV. Proof of Main Results

A. Auxiliary Results

First, we list some general results related to the basic recursion in (II.5) which is crucial for our work. Here, we assume that the empty product has value 1, i.e. $\prod_{i=m}^{k} \cdots \equiv 1$ for $m > k$.

Lemma 5. Let $r_t \in (0,1)$ for $t = 1, 2, \ldots$.

a)
$$\sum_{t=1}^{\infty} r_t = \infty \iff \prod_{t=1}^{\infty}(1 - r_t) = 0 \iff \prod_{t=1}^{\infty}(1 - cr_t) = 0 \qquad \text{for any } 0 < c \leq 1.$$

b) Let $w_t \in [0,1], t = 0, 1, \ldots$ be a given sequence and $q_0 \in (0,1)$ a starting value for the recursion
$$q_{t+1} = (1 - r_{t+1})q_t + r_{t+1}w_t, \quad t \geq 0. \tag{IV.1}$$

b 1) For $t \geq 0$ the recursion has the unique solution
$$q_t = q_0 \prod_{m=1}^{t}(1-r_m) + \sum_{m=1}^{t} r_m w_{m-1} \prod_{i=m+1}^{t}(1-r_i). \tag{IV.2}$$

b 2) If $w_t \equiv w \in [0,1]$ then
$$q_t = w - (w - q_0)\prod_{m=1}^{t}(1 - r_m), \quad t \geq 0 \tag{IV.3}$$

b 3) If $\prod_{m=1}^{\infty}(1 - r_m) = 0$ and if $\lim_{t\to\infty} w_t \to w$ for some $w \in [0,1]$ then also $\lim_{t\to\infty} q_t \to w$.

Proof: Most of the Lemma can be proved using standard methods, see [10] for more details. For b 3) use b 2) with the bounds $w \pm \epsilon$ and let $\epsilon \to 0$. ∎

The convergence of the sample generating probability $Q(a; i{+}1, y, \mathbf{\Pi}_t)$ as defined in (II.2) is the crucial step in proving absorption. We need some analogy of the basic recursion (II.5) for Q, but the dependence within the sampled solution introduced by the feasibility function $C_i(y, \cdot)$ prevents a direct generalization of (II.5). We therefore introduce a surrogate function Q' that helps to transfer bounds from the recursion to Q.

For $\boldsymbol{p} \in \mathbb{P}, i \in I, y \in R_i$ and $a \in \mathcal{A}$ define
$$Q'(a; i{+}1, y, \boldsymbol{p}) := \frac{\boldsymbol{p}(a; i{+}1)}{\sum_{a' \in C_i(y)} \boldsymbol{p}(a'; i{+}1)} \mathbb{1}_{C_i(y)}(a) \tag{IV.4}$$
where we again use $\frac{0}{0} = 0$. We want to bound $Q(a; i{+}1, y, \boldsymbol{p})$ with the help of $Q'(a; i{+}1, y, \boldsymbol{p})$. Let
$$c_1 := \max\{C_i(y,a) \mid y \in R_i,\ a \in \mathcal{A},\ i \in I\},$$
$$c_0 := \min\{C_i(y,a) > 0 \mid y \in R_i, a \in \mathcal{A}, i \in I\}.$$
Then $0 < c_0 \leq c_1 \leq 1$ and for $x \in [0,1]$ we may define the bounding functions
$$h(x) := \frac{c_1 x}{c_0 + (c_1 - c_0)x}, \quad \ell(x) := \frac{c_0 x}{c_1 - (c_1 - c_0)x}. \tag{IV.5}$$
We first collects a few properties of h and ℓ, the simple proofs are omitted (see [10]).

Lemma 6. **a)** $\ell(x) \leq x \leq h(x)$, $h(x) = 1 - \ell(1-x)$ and $\ell(x) = 1 - h(1-x)$.

b) h and ℓ are both continuous and strictly increasing with $\ell(0) = h(0) = 0,\ \ell(1) = h(1) = 1$.

c) Suppose $(x_n)_{n\in\mathbb{N}}$ is a convergent sequence in $[0,1]$ then for any constant $c \in (0,1]$, we have
$$\sum_n x_n < \infty \iff \sum_n h(cx_n) < \infty \iff \sum_n \ell(cx_n) < \infty.$$

Now the next Lemma gives the desired bounds of Q, for a proof see also [10].

Lemma 7. Let $i \in I, y \in R_i$ and $a \in C_i(y)$. Then,
$$\ell\big(Q'(a; i{+}1, y, \boldsymbol{p})\big) \leq Q(a; i{+}1, y, \boldsymbol{p}) \leq h\big(Q'(a; i{+}1, y, \boldsymbol{p})\big) \quad \text{for all } p \in \mathbb{P}.$$

B. Proof of Lemma 2

For $t \geq 0$, $i \in I$ and $y \in R_i$, define
$$G_i(y,t) := \sum_{a' \in C_i(y)} \mathbf{\Pi}_t(a'; i{+}1), \text{ and denote} \tag{IV.6}$$
$$\varrho_t^y := \frac{\varrho_t}{G_i(y,t)}, \text{ for all } t \geq 1. \tag{IV.7}$$
Note that from (III.1) and (IV.2) we may show that $G_i(y,t) > 0$. The next Lemma shows that under the condition (III.6) of partial absorption, $Q'(a; i{+}1, y, \mathbf{\Pi}_t)$ also satisfies the basic recursion (II.5) with ϱ_t^y replacing ϱ_t.

Lemma 8. Let $i \in I$ be fixed and assume that (III.6) holds with probability one for random variables T and y as given in Lemma 2. Then the following holds almost surely for all $t > T$:

a) $0 < \varrho_t \leq \varrho_t^y < 1$.

b) $Q'(a; i{+}1, y, \mathbf{\Pi}_t) = (1 - \varrho_t^y)Q'(a; i{+}1, y, \mathbf{\Pi}_{t-1}) + \varrho_t^y\, \boldsymbol{W}_{t-1}(a; i{+}1) \quad$ for all $a \in C_i(y)$.

Proof: Assume (III.6) holds and let $t > T$. From (III.3) we see that $\boldsymbol{W}_t(a; i{+}1) > 0$ is possible only for symbols a that are feasible on position $i{+}1$. Hence under the assumption (III.6) we know that $\boldsymbol{W}_t(a; i{+}1)$ must be concentrated on $C_i(y)$, i.e. $\sum_{a\in C_i(y)} \boldsymbol{W}_t(a; i{+}1) = 1$. Hence we obtain from the basic recursion (II.5) that
$$G_i(y,t) = (1 - \varrho_t)\, G_i(y, t-1) + \varrho_t > \varrho_t > 0. \tag{IV.8}$$

a) follows immediately from (IV.8).

b) For $a \in C_i(y)$ and $t > T$
$$\begin{aligned} Q'(a; i{+}1, y, \mathbf{\Pi}_t) &= \tfrac{\mathbf{\Pi}_t(a;i+1)}{\sum_{a'\in C_i(y)} \mathbf{\Pi}_t(a';i+1)} \\ &= \tfrac{\mathbf{\Pi}_t(a;i+1)}{G_i(y,t)} = \tfrac{(1-\varrho_t)\mathbf{\Pi}_{t-1}(a;i+1,y)}{G_i(y,t)} + \varrho_t^y \boldsymbol{W}_t(a; i{+}1) \\ &= (1 - \varrho_t^y)Q'(a; i{+}1, y, \mathbf{\Pi}_{t-1}) + \varrho_t^y \boldsymbol{W}_t(a; i{+}1), \end{aligned}$$
where we use (IV.8). ∎

Proof of Lemma 2: This proof extends the approach used in [13] and [10]. Let $i \in I$ be fixed and let y and T

be such that (III.6) holds with probability one. Note, that if $|C_i(y)| = 1$, the conclusion holds trivially, so we may assume $|C_i(y)| > 1$. We first show that if there is a (random variable) $a_0 \in \mathcal{A}$ such that almost surely

$$\lim_{t\to\infty} Q'(a_0; i{+}1, y, \mathbf{\Pi}_t) = \lim_{t\to\infty} Q(a_0; i{+}1, y, \mathbf{\Pi}_t) = 1 \tag{IV.9}$$

then the assertion of Lemma 2 holds.

We write $[\boldsymbol{X}_t^{(\cdot)}(i{+}1) \equiv a_0]$ for the event $[\boldsymbol{X}_t^{(n)}(i{+}1) = a_0, n = 1, \ldots, N]$ and $\mathfrak{X}_k^m$ for the event $[\boldsymbol{X}_l^{(\cdot)}(i{+}1) \equiv a_0, l = k, \ldots, m-1]$. In order to prove the Lemma, we have to show that the following probability is equal to one:

$$\begin{aligned}&\mathbf{P}\Big(\exists k\ \forall m \geq k \quad X_m^{(\cdot)}(i{+}1) \equiv a_0\Big) \\ &\quad= \lim_{k\to\infty} \Big[\mathbf{P}\Big(X_k^{(\cdot)}(i{+}1) \equiv a_0\Big) \\ &\qquad\cdot \prod_{m=k+1}^{\infty} \mathbf{P}\big(X_m^{(\cdot)}(i{+}1) \equiv a_0 \big| \mathfrak{X}_k^m\big)\Big]\end{aligned} \tag{IV.10}$$

Now from (II.4) and (IV.9) using bounded convergence, we see that for the first factor in (IV.10)

$$\begin{aligned}\lim_{k\to\infty} \mathbf{P}\big(X_k^{(\cdot)}(i{+}1) \equiv a_0\big) &= \lim_{k\to\infty} \mathbf{E}\big(Q(a_0; i{+}1, y, \mathbf{\Pi}_k)^N\big) \\ &= \mathbf{E}\Big(\lim_{k\to\infty} \big(Q(a_0; i{+}1, y, \mathbf{\Pi}_k)\big)^N\Big) = 1.\end{aligned} \tag{IV.11}$$

For the second expression in (IV.10) we obtain for $k \geq T$ and any $m \geq k+1$ using Lemma 7

$$\begin{aligned}&\mathbf{P}\big[X_m^{(\cdot)}(i{+}1) \equiv a_0 \ \big|\mathfrak{X}_k^m\big] \\ &\quad\geq \mathbf{E}\Big[\Big(\ell\big(Q'(a_0; i{+}1, y, \mathbf{\Pi}_m)\big)\Big)^N \ \Big|\mathfrak{X}_k^m\Big].\end{aligned} \tag{IV.12}$$

By (III.4), the condition $\boldsymbol{X}_l^{(\cdot)}(i{+}1) \equiv a_0$ implies $\boldsymbol{W}_l(a_0; i{+}1) = 1$, therefore $w_l := \boldsymbol{W}_l(a_0; i{+}1) = 1$ for $l = k, \ldots, m-1$ under $\mathfrak{X}_k^m$. We may now use Lemma 8 b), Lemma 5 b 2), (IV.2) and $\varrho_m^y \geq \varrho_m \geq \varrho > 0$ to obtain

$$\begin{aligned}&Q'(a_0; i{+}1, y, \mathbf{\Pi}_m) \\ &\quad\geq 1 - \big(1 - Q'(a_0; i{+}1, y, \mathbf{\Pi}_k)\big)\ (1-\varrho)^{m-k}.\end{aligned} \tag{IV.13}$$

Now let $H_m(k) := h\big((1 - Q'(a_0; i{+}1, y, \mathbf{\Pi}_k))(1-\varrho)^m\big)$, h as in (IV.5). Then by Lemma 6 a) we see from (IV.13)

$$\ell\big(Q'(a_0; i{+}1, y, \mathbf{\Pi}_m)\big) \geq 1 - H_{m-k}(k). \tag{IV.14}$$

Moreover, by assumption (IV.9) we have $\lim_{k\to\infty} H_m(k) = 0$ almost surely. Using (IV.12), (IV.13) and (IV.14), we may drop the condition $\mathfrak{X}_k^m$ and obtain

$$\begin{aligned}&\lim_{k\to\infty} \prod_{m=k+1}^{\infty} \mathbf{P}\big[X_m^{(\cdot)}(i{+}1) \equiv a_0 \ \big|\mathfrak{X}_k^m\big] \\ &\geq \lim_{k\to\infty} \prod_{m=1}^{\infty} \mathbf{E}\big[1 - H_m(k)\big]^N \\ &= \prod_{m=1}^{\infty} \mathbf{E}\Big(\lim_{k\to\infty} \big[1 - H_m(k)\big]^N\Big) = 1.\end{aligned} \tag{IV.15}$$

The interchange of $\lim$ and $\prod$ may be justified using logarithms and the fact that $0 \leq H_m(k) \leq h((1-\varrho)^m) \leq 1$ and $\prod_{m=1}^{\infty}\big[1 - h((1-\varrho)^m)\big] > 0$ by Lemma 6 c), Lemma 5 a).

Hence, the probability in (IV.10) must be one as was claimed in the Lemma.

We still have to prove (IV.9). We use an approach of [13] and first show that the learning process $\boldsymbol{W}_t$ must become monotone for large t, i.e. there is a random variable $\hat{T} \geq T$ such that for $t \geq \hat{T}$ and for any $a \in \mathcal{A}$

$$t \mapsto Z_t(a) = \boldsymbol{W}_t(a; i{+}1) - \boldsymbol{W}_{t-1}(a; i{+}1)$$

does not have any sign changes. As $\boldsymbol{W}_t(\cdot; i{+}1)$ is a probability measure on the set $\mathcal{A}$, it is then clear that $\boldsymbol{W}_t(a; i{+}1)$ must converge almost surely to a random variable V_a taking values in $[0, 1]$. We fix $a \in \mathcal{A}$ and use the abbreviation $W_t := \boldsymbol{W}_t(a; i{+}1)$. We show below that there is a $\kappa > 0$ such that for any $t > T$

$$\begin{aligned}&\mathbf{P}[W_m = 1 \text{ for all } m \geq t \mid W_{t-1} > W_{t-2}] \geq \kappa > 0 \\ &\qquad\text{and} \\ &\mathbf{P}[W_m = 0 \text{ for all } m \geq t \mid W_{t-1} < W_{t-2}] \geq \kappa > 0.\end{aligned} \tag{IV.16}$$

Let M_k be the time of the k-th sign change of $Z_t(a)$, then (IV.16) tells us that for any $k \geq 1$

$$\mathbf{P}[M_{k+1} < \infty \mid M_k < \infty] \leq 1 - \kappa < 1,$$

and then, as $M_k \leq M_{k+1}$

$$\begin{aligned}&\mathbf{P}(t \mapsto W_t \text{ has } \infty\text{-ly many sign changes}) \\ &\quad= \mathbf{P}[M_1 < \infty] \prod_{k=1}^{\infty} \mathbf{P}[M_{k+1} < \infty \mid M_k < \infty] \\ &\quad\leq \prod_{k=1}^{\infty} (1-\kappa) = 0\end{aligned}$$

So, in order to prove the monotonicity of the learning process, we have to prove (IV.16).

We use the abbreviation $\mathfrak{W}_k$ for the event $[W_{t-1} > W_{t-2}, W_{t+l} = 1 \text{ for } l = 0, \ldots, k-1]$. From assumption (III.4) we see that $\boldsymbol{X}_t^{(\cdot)}(i{+}1) \equiv a$ implies $W_t = 1$. Hence, for $t > T$

$$\begin{aligned}&\mathbf{P}\Big[W_{t+m} = 1, m \geq 0 \mid W_{t-1} > W_{t-2}\Big] \\ &\quad= \mathbf{P}\Big[W_t = 1 \mid \mathfrak{W}_0\Big] \prod_{m=1}^{\infty} \mathbf{P}\Big[W_{t+m} = 1 \mid \mathfrak{W}_m\Big] \\ &\quad\geq \mathbf{E}\Big[\big[Q(a; i{+}1, y, \mathbf{\Pi}_t)\big]^N \mid \mathfrak{W}_0\Big] \\ &\qquad\cdot \prod_{m=1}^{\infty} \mathbf{E}\Big[\big[Q(a; i{+}1, y, \mathbf{\Pi}_{t+m})\big]^N \ \Big|\ \mathfrak{W}_m\Big].\end{aligned} \tag{IV.17}$$

As before, we give a lower bound for the last expression using $\ell\big(Q'\big)$. From assumption (III.5) we see that $\boldsymbol{W}_{t-1} > \boldsymbol{W}_{t-2} \geq 0$ implies $\boldsymbol{W}_{t-1} \geq \alpha$. From Lemma 8 a) and b) we may therefore derive

$$Q'(a; i{+}1, y, \mathbf{\Pi}_t) \geq \alpha \cdot \varrho_t^y \geq \alpha \cdot \varrho > 0. \tag{IV.18}$$

For any $m \geq 1$ we may deduce from condition $\mathfrak{W}_m$ using Lemma 8 b) and Lemma 5 b 2) with $w = 1$ that

$$Q'(a; i{+}1, y, \mathbf{\Pi}_{t+m}) \geq 1 - \prod_{l=1}^{m}(1 - \varrho^{y}_{t+l}) \geq 1 - (1-\varrho)^m. \tag{IV.19}$$

Hence, using Lemma 7 we get from (IV.17) - (IV.19),

$$\begin{aligned}\mathbf{P}[W_m = 1 \text{ for all } m \geq t \mid W_{t-1} > W_{t-2}] \\ \geq & \ell(\alpha \cdot \varrho)^N \prod_{m=1}^{\infty} \Big[\ell\big(1-(1-\varrho)^m\big)\Big]^N \\ &= \Big[\ell(\alpha\cdot\varrho)\prod_{m=1}^{\infty}\Big(1 - h\big((1-\varrho)^m\big)\Big)\Big]^N := \kappa.\end{aligned}$$

From Lemma 6 c) we see that $\sum_{m=1}^{\infty}(1-\varrho)^m < \infty$ implies $\sum_{m=1}^{\infty} h\big((1-\varrho)^m\big) < \infty$, hence $\kappa > 0$.

In a completely analogous manner the second inequality in (IV.16) is shown.

As mentioned above, we now know that with probability one, $\boldsymbol{W}_t(a; i{+}1)$ converges to some V_a as $t \to \infty$. Now we show that $V_a \in \{0, 1\}$, more precisely:

$$\exists a_0 \in C_i(y)\ V_{a_0} = 1 \text{ and } \forall a \in \big(C_i(y) - \{a_0\}\big)\ V_a = 0 \tag{IV.20}$$

holds almost surely, where a_0 is a random variable itself. To prove (IV.20), we first show that for all $a \in C_i(y)$ we have $\mathbf{P}(V_a \in (0,1)) = 0$. We again use the abbreviation $W_m = \boldsymbol{W}_m(a; i{+}1)$, and use $\mathfrak{V}_k^m$ for the event $[W_l \in (0,1), l = k, \ldots, m-1]$. We then have

$$\begin{aligned}\mathbf{P}(V_a \in (0,1)) &= \mathbf{P}(\lim_{k\to\infty} W_k \in (0,1)) \qquad \text{(IV.21)} \\ &\leq \lim_{k\to\infty}\prod_{m=k+1}^{\infty}\mathbf{P}\big[W_m \in (0,1) \mid \mathfrak{V}_k^m\big].\end{aligned}$$

Recall that by assumption (III.3), $W_m > 0$ implies that a occurs at least once at $i{+}1$-th positions in the sample $\boldsymbol{X}_m$. For $k \geq T$ and any $m \geq k+1$, we therefore have

$$\begin{aligned}&\mathbf{P}[W_m \in (0,1) \mid \mathfrak{V}_k^m] \qquad \text{(IV.22)} \\ &\leq 1 - \mathbf{P}\big[\boldsymbol{X}_m^{(n)}(i{+}1) \neq a, n = 1, \ldots, N \ \big|\ \mathfrak{V}_k^m\big] \\ &\leq 1 - \Big(1 - h\big((1-\varrho)^{m-k} + 1 - \alpha\big)\Big)^N,\end{aligned}$$

where the second inequality is justified as we have by assumption (III.5) that $W_m \in (0,1)$ implies $\alpha \leq W_m \leq 1-\alpha$. Then Lemma 8 b) and Lemma 5 b 2) with $w := 1-\alpha$ show that for $m > k \geq T$

$$\begin{aligned}&Q'(a; i{+}1, y, \mathbf{\Pi}_m) \leq Q'(a; i{+}1, y, \mathbf{\Pi}_k)\cdot \\ &\prod_{l=k+1}^{m}(1-\varrho_l^y) + (1-\alpha)\big(1 - \prod_{l=k+1}^{m}(1-\varrho_l^y)\big) \\ &\leq \prod_{l=k+1}^{m}(1-\varrho_l^y) + 1 - \alpha \leq (1-\varrho)^{m-k} + 1 - \alpha.\end{aligned}$$

For m large enough, $(1-\varrho)^{m-k}$ will be smaller than α, hence the upper bound of the probability (IV.22) is smaller than 1 and the infinite product in (IV.21) vanishes. Therefore, we have shown

$$\mathbf{P}\big(V_a \in \{0,1\} \ \text{ for all } a \in C_i(y)\big) = 1. \tag{IV.23}$$

We also know that $\mathbf{P}$-almost surely

$$1 = \sum_{a\in C_i(y)} W_m = \sum_{a \in C_i(y)} \boldsymbol{W}_m(a; i{+}1)$$

for all $m \geq T$. Hence, this must also hold for the limits: $\mathbf{P}\big(\sum_{a\in C_i(y)} V_a = 1\big) = 1$. This together with (IV.23) proves (IV.20), i.e. there is a_0 such that $\mathbf{P}(V_{a_0} = 1) = 1$.

From Lemma 8 b) and the last assertion in Lemma 5 b) we see that then also

$$\lim_{t\to\infty} Q'(a; i{+}1, y, \mathbf{\Pi}_t) = \lim_{t\to\infty}\boldsymbol{W}_t(a; i{+}1) = V_{a_0} = 1$$

proving the first part of (IV.9). Now as the bounding functions h, ℓ are continuous (Lemma 6 b)) and from Lemma 7 we see that almost surely for all $a \in C_i(y)$

$$\begin{aligned}1 = \ell(V_{a_0}) &\leq \liminf_{t\to\infty} Q(a_0; i{+}1, y, \mathbf{\Pi}_t) \qquad \text{(IV.24)} \\ &\leq \limsup_{t\to\infty} Q(a_0; i{+}1, y, \mathbf{\Pi}_t) \leq h(V_{a_0}) = 1.\end{aligned}$$

proving the second part of (IV.9). This completes the proof of Lemma 2. ■

C. Proof of Theorem 3

Proof of Theorem 3: Absorption formally means that there are random variables $T < \infty$ and $\sigma = (\sigma_1, \ldots, \sigma_L)$ taking on values in S such that almost surely

$$\text{for all } t > T \quad \boldsymbol{X}_t^{(\cdot)} = \sigma. \tag{IV.25}$$

By assumption (III.3) and (III.4) this implies $\boldsymbol{W}_t(\sigma_i, i) = 1$ and $\boldsymbol{W}_t(a, i) = 0$ for all $a \neq \sigma_i$ and all $t > T$. Hence we may use the basic recursion and Lemma 5 b 2) to obtain for all $t > T$ almost surely

$$\begin{aligned}\mathbf{\Pi}_t(a\,;\ i) &\leq \mathbf{\Pi}_T(a\,;\ i)(1-\varrho)^{t-T} \qquad \text{for } a \neq \sigma_i \\ \mathbf{\Pi}_t(\sigma_i\,;\ i) &\geq 1 - \big(1 - \mathbf{\Pi}_T(\sigma_i\,;\ i)\big)(1-\varrho)^{t-T}\end{aligned}$$

which for $t \to \infty$ converges to the probability measure concentrated on σ. ■

D. Proof of Theorem 4

Proof of Theorem 4: Cp. [13]. Assume that $s^* = (s_1^*, \ldots, s_L^*)$ is the unique optimal solution. We use $[\boldsymbol{X}_t^{(\cdot)}(1) \neq s_1^*]$ to abbreviate $[\boldsymbol{X}_t^{(n)}(1) \neq s_1^*$ for all $n \in \{1, \ldots, N\}]$.

We first derive a necessary condition of reachability. Observe that $[\boldsymbol{X}_t^{(\cdot)}(1) \neq s_1^*]$ for all $t \in \mathbb{N}$ implies $\tau = \infty$. Hence, $\mathbf{P}(\tau < \infty) = 1$ requires $\mathbf{P}\big(\boldsymbol{X}_t^{(\cdot)}(1) \neq s_1^*$ for all $t \geq 0\big) = 0$. Denote $\mathfrak{X}_t$ for $[X_m^{(\cdot)}(1) \neq s_1^*, m = 0, \ldots, t-1]$, then

$$\begin{aligned}0 &= \mathbf{P}\big(\boldsymbol{X}_t^{(\cdot)}(1) \neq s_1^* \text{ for all } t \in \mathbb{N} \geq 0\big) \qquad \text{(IV.26)} \\ &= \mathbf{P}\big(\boldsymbol{X}_0^{(\cdot)}(1) \neq s_1^*\big)\prod_{t=1}^{\infty}\mathbf{P}\big[\boldsymbol{X}_t^{(\cdot)}(1) \neq s_1^* | \mathfrak{X}_t\big]\end{aligned}$$

Observe that using Lemma 7 for $t \geq 1$

$$\begin{aligned}
&\mathbf{P}\big[X_t^{(\cdot)}(1) \neq s_1^* \big| \mathfrak{X}_t\big] = \mathbf{E}\Big[\big(1 - Q(s_1^*; 1, \diamond, \mathbf{\Pi}_t)\big)^N \big| \mathfrak{X}_t\Big] \\
&\quad \geq \mathbf{E}\Big[\Big(1 - h\big(Q'(s_1^*; 1, \diamond, \mathbf{\Pi}_t)\big)\Big)^N \big| \mathfrak{X}_t\Big] \qquad \text{(IV.27)} \\
&\quad = \Big[1 - h\big(Q'(s_1^*; 1, \diamond, \mathbf{\Pi}_0) \prod_{m=1}^{t} (1 - \varrho_m^\diamond)\big)\Big]^N,
\end{aligned}$$

where we use the fact that under condition $\mathfrak{X}_t$ we have $\boldsymbol{W}_m(s_1^*, 1) = 0$ for $m = 0, \ldots, t-1$ by assumption (III.3). Hence we may apply Lemma 5 b 2) with $w = 0$ to the recursion of Lemma 8 b) as the assumptions of this Lemma hold for $i = 0, y = \diamond$, $T = 0$ and obtain the last equality in (IV.27). Recall that $\varrho_m^\diamond = \varrho_m / \sum_{a' \in C_0(\diamond)} \mathbf{\Pi}_m(a', 1)$.

Under assumption (III.2) and (III.1), we know $Q(s_1^*; 1, \diamond, \mathbf{\Pi}_0) \in (0, 1)$ and hence $\mathbf{P}\big(\boldsymbol{X}_0^{(\cdot)}(1) \neq s_1^*\big) > 0$. Now (IV.26) and (IV.27) show that reachability of the optimal solution requires

$$\prod_{t=1}^{\infty} \Big[1 - h\big(Q'(s_1^*; 1, \diamond, \mathbf{\Pi}_0) \prod_{m=1}^{t} (1 - \varrho_m^\diamond)\big)\Big] = 0,$$

and by Lemma 5 a) and Lemma 6 c) this is equivalent to

$$\sum_{t=1}^{\infty} \ell\big(\prod_{m=1}^{t} (1 - \varrho_m^\diamond)\big) = \infty. \qquad \text{(IV.28)}$$

We are now going to show that absorption implies that (IV.28) does not hold. If the algorithm is almost surely absorbed in finite time, then

$$\begin{aligned}
1 &= \mathbf{P}\big(\exists a \in \mathcal{A}\ \exists k \in \mathbb{N}\ \forall m \geq k \quad X_m^{(\cdot)}(1) \equiv a\big) \\
&= \sum_{a \in \mathcal{A}} \mathbf{P}\big(\exists k \in \mathbb{N}\ \forall m \geq k \quad X_m^{(\cdot)}(1) \equiv a\big) \qquad \text{(IV.29)} \\
&= \lim_{k \to \infty} \sum_{a \in \mathcal{A}} \mathbf{P}\big(\forall m \geq k \quad X_m^{(\cdot)}(1) \equiv a\big).
\end{aligned}$$

Fix $a \in \mathcal{A}$ and let $\mathfrak{N}_k^m$ denote $[X_l^{(\cdot)} \equiv a, l = k, \ldots, m-1]$ for $m > k$. Then by assumption (III.4) under condition $\mathfrak{N}_k^m$ we have $\boldsymbol{W}_l(a; 1) = 1$ for $l = k, \ldots, m-1$. We may then use the recursion of Lemma 8 b) for $y = \diamond, T = 0$ and obtain from Lemma 5 b 2)

$$Q'(a; 1, \diamond, \mathbf{\Pi}_m) \leq 1 - \big(1 - Q'(a; 1, \diamond, \boldsymbol{p}_0)\big) \prod_{l=1}^{m} (1 - \varrho_l^\diamond).$$

This may be used to deduce in (IV.29)

$$\begin{aligned}
&\mathbf{P}\big(\forall m \geq k \quad X_m^{(\cdot)}(1) \equiv a\big) \qquad \text{(IV.30)} \\
&\leq \prod_{m=k+1}^{\infty} \mathbf{P}\big(X_m^{(\cdot)}(1) \equiv a \bigm| \mathfrak{N}_k^m\big) \\
&\leq \prod_{m=k+1}^{\infty} \Big[1 - \ell\Big(\big(1 - Q'(a; 1, \diamond, \boldsymbol{p}_0)\big) \prod_{l=1}^{m} (1 - \varrho_l^\diamond)\Big)\Big]^N
\end{aligned}$$

With (IV.30) and (IV.29), we derive that absorption requires

$$\prod_{m=1}^{\infty} \Big[1 - \ell\Big(\big(1 - Q'(a; 1, \diamond, \boldsymbol{p}_0)\big) \prod_{l=1}^{m} (1 - \varrho_l^\diamond)\Big)\Big]^N > 0$$

for at least one $a \in \mathcal{A}$ which again by Lemma 5 a) and Lemma 6 c) is equivalent to

$$\sum_{m=1}^{\infty} \ell\big(\prod_{l=1}^{m} (1 - \varrho_l^\diamond)\big) < \infty$$

contradicting (IV.28).

■

V. Conclusion

In this paper, we examined the absorption property of an extended MBS algorithm with update rule (I.1) and a learning model as in assumptions (III.3) - (III.5). We showed that the popular setting $\varrho_t = \varrho > 0$ will almost surely lead to a stand-still after finitely many iterations, i.e. the $\boldsymbol{X}_t$-process is absorbed, the $\mathbf{\Pi}_t$-process is convergent and we cannot be sure that we can reach an optimal solution.

We are presently investigating the run time behavior of our algorithm applied to the so-called 'leading one' reward function. This together with a less restrictive learning that allows to include updating from the best-so-far-solutions will be the focus of our future research.

References

[1] M. Zlochin, M. Birattari, N. Meuleau, and M. Dorigo, "Model-based search for combinatorial optimization: A critical survey," *Annals of Operations Research*, vol. 131, no. 1-4, pp. 373–395, 2004.

[2] M. Dorigo and T. Stützle, *Ant colony optimization*. Cambridge, Massachusetts: A Bradford Book, MIT Press, 2004.

[3] R. Y. Rubinstein and D. P. Kroese, *The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning*. Springer, 2004.

[4] M. Hauschild and M. Pelikan, "An introduction and survey of estimation of distribution algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 3, pp. 111–128, 2011.

[5] H. Robbins and S. Monro, "A stochastic approximation method," *The Annals of Mathematical Statistics*, pp. 400–407, 1951.

[6] S. Baluja, "Population-based incremental learning. a method for integrating genetic search based function optimization and competitive learning," DTIC Document, Tech. Rep., 1994.

[7] P. T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of operations research*, vol. 134, no. 1, pp. 19–67, 2005.

[8] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 26, no. 1, pp. 29–41, 1996.

[9] H. Asoh and H. Mühlenbein, "On the mean convergence time of evolutionary algorithms without selection and mutation," in *Parallel Problem Solving from Nature—PPSN III*. Springer, 1994, pp. 88–97.

[10] Z. Wu and M. Kolonko, "Asymptotic properties of a generalized cross entropy optimization algorithm," *submitted, under revision*, 2014.

[11] C. Blum, *Theoretical and practical aspects of ant colony optimization*. IOS Press, 2004, vol. 282.

[12] M. Pelikan, D. E. Goldberg, and F. G. Lobo, "A survey of optimization by building and using probabilistic models," *Computational optimization and applications*, vol. 21, no. 1, pp. 5–20, 2002.

[13] A. Costa, O. D. Jones, and D. Kroese, "Convergence properties of the cross-entropy method for discrete optimization," *Operations Research Letters*, vol. 35, no. 5, pp. 573–580, 2007.