

# On the Finite-Time Dynamics of Ant Colony Optimization

Walter J. Gutjahr  
Department of Statistics and Decision Support Systems  
University of Vienna  
walter.gutjahr@univie.ac.at

**Abstract.** An analytical framework for investigating the finite-time dynamics of ant colony optimization (ACO) under a fitness-proportional pheromone update rule on arbitrary construction graphs is developed. A limit theorem on the approximation of the stochastic ACO process by a deterministic process is demonstrated, and a system of ordinary differential equations governing the process dynamics is identified. As an example for the application of the presented theory, the behavior of ACO on three different construction graphs for subset selection problems is analyzed and compared for some basic test functions. The theory enables first rough theoretical predictions of the convergence speed of ACO.

**Keywords.** Ant colony optimization, convergence speed, probabilistic algorithms, stochastic approximation, stochastic optimization.

**AMS subject classification:** 68W20, 68W40

## 1 Introduction

The aim of this work is to provide an analytical framework for investigating the dynamics of the Ant Colony Optimization (ACO) metaheuristic, and to illustrate the applicability of this framework by comparing different strategies for applying ACO to the special family of subset selection problems. ACO is an optimization technique inspired by observations on the behavior of biological ant colonies; it has been introduced by Dorigo, Maniezzo and Colomi [12], [13], and developed later into a metaheuristic for combinatorial optimization problems [10], [11], [14]. Recently, ACO algorithms have also turned out as competitive in a discrete *stochastic* optimization context [3], [19], [20], [34], [4].

From a more abstract point of view, ACO can be considered as reinforcement learning on a graph that encodes the problem instance, the so-called *construction graph*, as it has been formally defined in [16]. A *solution* to the given combinatorial optimization problem is encoded as a feasible walk on the construction graph.

A feasible walk starts in a fixed initial node of the construction graph and has to satisfy the constraint that each node is visited at most once, i.e., already visited nodes are “tabu”. There may be additional, problem-specific rules defining certain nodes as tabu in certain circumstances. When there is no neighbor anymore that is not tabu, the walk stops and is decoded as a solution to the given problem.

The probability  $p_{kl}$  to go from a node  $k$  to an allowed neighbor node  $l$  is chosen proportional to the so-called *pheromone*  $\tau_{kl}$ , a memory value storing how good transition  $(k, l)$  has been in previous walks. Pheromone is initialized by a constant value. After the walk has finished, pheromone is updated. There are several possible ways to do that. The “classical” update rule is the following: First, set  $\tau_{kl} = (1 - \rho)\tau_{kl}$  for each arc  $(k, l)$ , where  $\rho \in ]0, 1[$  is the so-called *evaporation factor*. Then, increase the pheromone on the arcs of the chosen walk  $x$  by a factor proportional to the fitness  $f(x)$  of the walk. In this way, the partial construction steps of solutions with good objective function values are reinforced. The process

of random walk construction and pheromone update is iterated. Instead of a single walk (“1 ant”), also  $s > 1$  walks may be constructed sequentially (“ $s$  ants”) in each iteration (“round”).

In general, the procedure above needs not to converge. For an alternative pheromone update rule where only the best walk seen so far is reinforced (“Global Best”), results on convergence both of solutions and of pheromone to the global optimum have been obtained ([16] – [18]). Convergence of solutions only can also be shown for the MAX-MIN Ant System developed by Stützle and Hoos [37], as demonstrated by Stützle and Dorigo in [36]. For most other usual update rules, the convergence question is still open. What is even more, the *finite-time* behavior of the ACO algorithm is widely unexplored in theoretical research. As a remarkable exception, we mention the articles by Merkle and Middendorf [26] – [27], who analyze the ACO dynamics by means of the so-called *ACO model* for a specific problem under an update rule where the best walk of each *round* is reinforced.

In the present article, we study the finite-time ACO dynamics under the *classical* update rule described above. For this purpose, the ACO process is approximated in Section 2 by its behavior in the case of a sufficiently large number of ants (the resulting process resembles the ACO model process by Merkle and Middendorf), and the validity of this approximation is demonstrated by a mathematical convergence result. Furthermore, it is shown that in an asymptotic case where  $\rho \rightarrow 0$ , compensated by a suitable re-scaling of the time axis, the approximating process can be described by a system of ordinary differential equations. Section 3 indicates how to compute the dynamics of the expected fitness in the approximating process. In Section 4, three types of construction graphs for the application of ACO to the class of subset problems (including, e.g., knapsack problems or max-clique problems) are introduced. Section 5 compares the behavior of the approximating process on these construction graphs for a few basic test functions. Section 6, finally, gives some concluding remarks.

## 2 The Asymptotic Dynamics and its Basic Law

### 2.1 ACO Algorithm in Formal Terms

We start with a formal presentation of the ACO algorithm investigated in this work. Let us mention that for our purpose, the algorithm has been simplified by omitting *visibility* values (see, e.g., [10]), an additional feature often applied in practice to further improve the performance of ACO. However, since visibility values are based on problem-specific domain knowledge, an analysis of their influence should take into account the specific features of the optimization problem under consideration. The *general* properties of ACO dynamics show in a clearer way if the visibility concept is put aside.

As mentioned in the Introduction, ACO can be considered as a learning procedure working on a *construction graph*  $\mathcal{C}$  which encodes the given instance of the combinatorial optimization problem under consideration. Examples for such encodings will be presented in Section 4.  $\mathcal{C}$  is a directed graph with a distinguished node, the *start node*  $v_0$ . Let  $\bar{S}$  denote the set of directed paths  $x = (v^{(1)} = v_0, v^{(2)}, \dots, v^{(m)})$  in  $\mathcal{C}$  starting in  $v_0$ . The integer  $m$  does not need to be a constant, i.e., different paths in  $\bar{S}$  can have

different lengths. Observe that the property of  $x$  being a path implies that each node of  $\mathcal{C}$  is contained in  $x$  at most once. Optionally, a set of conditions on the path  $x$  can be given, specifying for each possible path  $u = (v^{(1)} = v_0, v^{(2)}, \dots, v^{(i)})$  a subset  $\mathcal{V}(u)$  of  $\mathcal{N}(v^{(i)})$ , where  $\mathcal{N}(v)$  is the set of immediate successor nodes of  $v$  in the directed graph  $\mathcal{C}$ , as the set of feasible continuation nodes of path  $u$ . The set  $\mathcal{V}(u)$  must satisfy the property that it does not contain any node already contained in  $u$ . In the presence of such conditions,  $\bar{S}$  is defined in a more restrictive way as the set of all directed paths  $x = (v^{(1)}, v^{(2)}, \dots, v^{(m)})$  with the property that  $v^{(1)} = v_0$  and that  $v^{(i+1)} \in \mathcal{V}((v^{(1)}, \dots, v^{(i)}))$  for all  $i = 1, \dots, m - 1$ . Also the case where there are no specific conditions on path  $x$  can be captured in this notation by setting in this case  $\mathcal{V}(u)$  equal to the set of all those nodes in  $\mathcal{N}(\text{last}(u))$  that are not already contained in  $u$ , where  $\text{last}(u)$  is the last node of  $u$ .

We define  $S$  as the set of paths in  $\bar{S}$  that cannot be prolonged without violating feasibility: Formally,  $S$  is the subset of  $\bar{S}$  containing all those paths  $x = (v^{(1)}, v^{(2)}, \dots, v^{(m)}) \in \bar{S}$  for which  $\mathcal{V}(x) = \emptyset$ . Throughout the paper, we assume that  $\mathcal{V}(v_0) \neq \emptyset$ , i.e., that the start node has always a feasible successor. (Otherwise, the empty path would be the only feasible path.) As a consequence, each path  $x \in S$  contains at least one node different from the start node.

It is assumed that each path  $x \in S$  encodes a *feasible solution* of the given combinatorial optimization problem. More precisely: There exists a mapping  $\Phi$  which assigns to each  $x \in S$  a unique feasible solution  $y$  of the optimization problem in such a way that to each feasible solution  $y$ , there exists an  $x \in S$  such that  $y = \Phi(x)$ . It is *not* required that  $\Phi$  is one-to-one, so a situation where for a feasible solution  $y$ , two different paths  $x_1 \in S$ ,  $x_2 \in S$  with  $\Phi(x_1) = \Phi(x_2) = y$  exist, is allowed.

In the sequel, the given combinatorial optimization problem will always be formulated as a *maximization* problem with objective function  $f$ . We shall call  $f(y)$  the *fitness* of solution  $y$ . Since the solution space is finite,  $f$  is a bounded function. We restrict ourselves to nonnegative fitness functions. Evidently, each optimization problem on a finite feasible set can be transformed in an equivalent problem of the considered form.

The fitness of a path  $x \in S$  is defined as  $f(\Phi(x))$ . By this definition, we may just as well consider the given problem as an optimization problem on the set  $S$ . For shortness, we abuse notation by simply writing  $f(x)$  instead of  $f(\Phi(x))$  ( $x \in S$ ).

The simplest example of a construction graph encoding is that for a *travelling salesperson problem* (TSP) on  $m$  nodes  $1, \dots, m$ : In this case,  $\mathcal{C}$  is the complete directed graph on node set  $\{1, \dots, m\}$ , the start node  $v_0$  is given as  $v_0 = 1$ , and additional conditions on  $x$  do not exist, such that  $S$  is the set of all permutations  $(x_1, \dots, x_m)$  of the indices  $1, \dots, m$  with  $x_1 = 1$ . The function  $\Phi$  is given as  $\Phi(x_1, \dots, x_m) = (x_1, \dots, x_m, x_1)$ , the last specifying a closed tour on node set  $\{1, \dots, m\}$ .

Based on the definitions above, we can now present a pseudocode formulation of the ACO algorithm, see Fig. 1. Therein, an *ant* is a conceptual unit performing a random walk on the construction graph  $\mathcal{C}$ . Nodes are represented by node indices. A *feasible continuation* of a path  $u = (v^{(1)}, \dots, v^{(i-1)}, v^{(i)} = k)$  is an arc  $(k, l)$  in  $\mathcal{C}$  with the property that  $l \in \mathcal{V}(u)$ ; if this property is not satisfied, arc  $(k, l)$  is called

*infeasible* (as a continuation of path  $u$ ).

The procedures pheromone-initialization and pheromone-update have still to be specified:

*Pheromone-initialization:* Each  $\tau_{kl}(n)$  is set equal to a pre-specified initial value:

$$\tau_{kl}(1) = \tau_{kl}^{(0)} \quad \text{for all arcs } (k, l).$$

(Note that  $\tau_{kl}(n)$  needs not to be defined if  $(k, l)$  is not an arc of  $\mathcal{C}$ .) Considering the possibility that in some construction graphs, not every arc plays the same role, we admit cases where the initial pheromone values are not all equal.

*Pheromone-update:* As stated in the Introduction, we apply the “classical” fitness-proportional update rule: In the case  $s = 1$  of a single ant,

$$\tau_{kl}(n+1) = (1 - \rho) \cdot \tau_{kl}(n) + \rho \cdot I((k, l) \in X_n^{(1)}) \cdot f(X_n^{(1)}), \quad (1)$$

where  $X_n^{(\sigma)}$  is the (random) walk traversed by ant  $\sigma$  in round  $n$ ,

$$I(\text{statement}) = \begin{cases} 1, & \text{if statement is true,} \\ 0, & \text{otherwise} \end{cases}$$

is the indicator function, and  $(k, l) \in X$  means that arc  $(k, l)$  is contained in the path defined by walk  $X$ .

---

#### Procedure ACO

---

```

procedure ACO {
  do pheromone-initialization;
  for round  $n = 1, 2, \dots$  {
    for ant  $\sigma = 1, \dots, s$  {
      set  $k$ , the current position of the ant, equal to the start node  $v_0$  of  $\mathcal{C}$ ;
      set  $u$ , the current walk of the ant, equal to  $(v_0)$ ;
      while (a feasible continuation of walk  $u$  of the ant exists) {
        select successor node  $l$  with probability  $p_{kl}(n)$ , where
          
$$p_{kl}(n) = \begin{cases} 0, & \text{if } (k, l) \text{ is infeasible,} \\ \tau_{kl}(n) / \left( \sum_{(k,r)} \tau_{kr}(n) \right), & \text{otherwise,} \end{cases}$$

          the sum being over all feasible continuations  $(k, r)$  of  $u$ ;
        set  $k = l$ , and append  $l$  to  $u$ ;
      } // end while
    } // end for ant
  do pheromone-update;
} // end for round
} // end procedure

```

---

*Fig. 1.* ACO’s pseudocode.

In the case of  $s > 1$  ants, the update contributions of each ant are averaged:

$$\tau_{kl}(n+1) = (1 - \rho) \cdot \tau_{kl}(n) + \frac{\rho}{s} \sum_{\sigma=1}^s I((k, l) \in X_n^{(\sigma)}) \cdot f(X_n^{(\sigma)}). \quad (2)$$

Taking the average instead of the sum of the contributions (i.e., multiplying the sum by  $\rho/s$  instead of  $\rho$ ) is appropriate since otherwise the process would degenerate with growing number  $s$  of ants.

## 2.2 Asymptotic Approximation of the Process

Let  $\mathcal{A}$  be the set of arcs  $(k, l)$  of  $\mathcal{C}$ . By indexing the elements of  $\mathcal{A}$  in an arbitrary order, the pheromone values  $\tau_{kl}$  can be combined to a vector  $\tau$  with  $|\mathcal{A}|$  nonnegative components,  $\tau \in [0, \infty^{|\mathcal{A}|}]$ . By  $\tau(n)$ , we denote the vector of pheromone values  $\tau_{kl}(n)$  ( $(k, l) \in \mathcal{A}$ ) in round  $n$  of the ACO process. We introduce the following definition:

**Definition 2.1.** The *passage fitness* of arc  $(k, l)$  under a given pheromone vector  $\tau = (\tau_{kl})$  is the random variable

$$I((k, l) \in X) \cdot f(X), \quad (3)$$

where  $X$  is the random walk of a fixed ant under pheromone values  $\tau_{kl}$ . Thus, the passage fitness of  $(k, l)$  is identical to the fitness of the resulting walk, provided that the walk contains arc  $(k, l)$ , and zero otherwise. The *expected passage fitness* of  $(k, l)$  under  $\tau$  is the mathematical expectation of the passage fitness, i.e., the value

$$F_{kl}(\tau) = E(I((k, l) \in X) \cdot f(X)) = \sum_{x \in \mathcal{S}: (k, l) \in x} Pr\{X = x\} \cdot f(x) \quad (4)$$

with  $X$  as defined above. It should be noted that the r.h.s. of this equation implicitly depends on  $\tau$ , as indicated on the l.h.s. Informally, the expected passage fitness of arc  $(k, l)$  describes the expected “gain” obtained from the random walk of an ant on the assumption that a walk  $X$  *not* passing through arc  $(k, l)$  produces no gain at all, and a walk  $X$  passing through  $(k, l)$  produces just the fitness of  $X$  as its gain.

Now we consider the sequence of vectors  $\bar{\tau}(n) = (\bar{\tau}_{kl}(n))$  ( $n \geq 1$ ) given by the difference equations

$$\begin{aligned} \bar{\tau}_{kl}(n+1) &= \bar{\tau}_{kl}(n) + \rho \{-\bar{\tau}_{kl}(n) + F_{kl}(\bar{\tau}(n))\} \quad (n \geq 1), \\ \bar{\tau}_{kl}(1) &= \tau_{kl}(1) = \tau_{kl}^{(0)} \end{aligned} \quad (5)$$

for each arc  $(k, l) \in \mathcal{A}$ , where  $F_{kl}(\bar{\tau}(n))$  is the expected passage fitness under  $\bar{\tau}(n)$ . We call  $(\bar{\tau}(n))$  the *associated discrete deterministic process* (ADDP). Whereas in the original ACO process, the pheromone vectors evolve in a stochastic manner, their evolution is described by the deterministic law (5) in the ADDP.

Our process ADDP is closely related to the so-called *ACO model* process in [26] – [27]. However, we presuppose a fitness-proportional instead of an iteration-best pheromone update rule, and we formulate the ADDP for the very broad range of arbitrary combinatorial optimization problems using arbitrary construction graph encodings.

We shall show now that under mild conditions, the trajectories  $(\tau(n))$  of the ACO process converge in probability to the trajectories  $(\bar{\tau}(n))$  of the ADDP within each finite time interval, as the number  $s$  of ants tends to infinity. The symbol  $\|\cdot\|$  will be used for the Euclidean norm.

**Theorem 2.1.** Let  $D \subseteq [0, \infty^{|\mathcal{A}|}]$  be a domain containing all possible trajectories of  $(\tau(n))$  and  $(\bar{\tau}(n))$ . Furthermore, let  $F(\tau) = (F_{kl}(\tau))$   $((k, l) \in \mathcal{A})$  be the vector of the expected passage fitness values under pheromone vector  $\tau$ . Suppose that  $F(\tau)$  is Lipschitz-continuous on  $D$ , i.e., that there exists a constant  $L > 0$  such that

$$\|F(\tau^{(1)}) - F(\tau^{(2)})\| \leq L \|\tau^{(1)} - \tau^{(2)}\|$$

for all  $\tau^{(1)} \in D$ ,  $\tau^{(2)} \in D$ . Then, for an arbitrary integer  $N^* \geq 1$ , an arbitrary  $\epsilon > 0$ , and an arbitrary  $\delta > 0$ , there exists an integer  $s^* = s^*(N^*, \epsilon, \delta)$  such that for all  $s \geq s^*$ ,

$$\|\tau(n) - \bar{\tau}(n)\| < \delta \quad \text{for all } n = 1, \dots, N^*$$

with a probability of at least  $1 - \epsilon$ . In other words: The trajectory  $\bar{\tau}(n)$  yields, with probability arbitrarily close to one, for any given number of rounds an arbitrarily good approximation to the trajectory  $\tau(n)$ , provided that the number  $s$  of ants is chosen large enough.

**Proof.** Let  $N^*$ ,  $\delta$  and  $\epsilon$  be given. W.l.o.g. we assume  $\epsilon < 1$ . For abbreviation, let us set

$$A_{kl}^{(\sigma)}(n) = I((k, l) \in X_n^{(\sigma)}) \cdot f(X_n^{(\sigma)}).$$

By definition of the expected passage fitness,

$$E(A_{kl}^{(\sigma)}(n)) = F_{kl}(\tau(n)).$$

Combining the random variables  $A_{kl}^{(\sigma)}(n)$  to vectors  $A^{(\sigma)}(n)$ , we re-write eq. (2) in vector form as

$$\tau(n+1) = \tau(n) + \rho \left\{ -\tau(n) + \frac{1}{s} \sum_{\sigma=1}^s A^{(\sigma)}(n) \right\},$$

or alternatively as

$$\tau(n+1) = \tau(n) + \rho \left\{ -\tau(n) + F(\tau(n)) + T^{(s)}(n) \right\},$$

where the noise term  $T^{(s)}(n)$  is given by

$$T^{(s)}(n) = \frac{1}{s} \sum_{\sigma=1}^s A^{(\sigma)}(n) - F(\tau(n))$$

and has expectation  $\mathbf{0} = (0, \dots, 0)$ . The random variables  $A^{(\sigma)}(n)$   $(\sigma = 1, \dots, s)$  are i.i.d. Therefore, by the Law of Large Numbers, to an arbitrary fixed  $\alpha > 0$ , there exists an  $s^*$ , such that

$$Pr\{\|T^{(s)}(n)\| \geq \alpha\} < \frac{\epsilon}{4N^*}$$

for all  $s \geq s^*$ . In other words: For  $s \geq s^*$ ,  $\|T^{(s)}(n)\| < \alpha$  holds with a probability of at least  $1 - \epsilon/(4N^*)$ . Since for each  $n$ , the random variable  $A^{(\sigma)}(n+1)$  is conditionally independent of  $A^{(\sigma)}(n)$ , conditioned

on the pheromone vector  $\tau(n+1)$  in round  $n+1$ , we can multiply the lower bounds  $1 - \epsilon/(4N^*)$  for the probabilities of the events  $\|T^{(s)}(n)\| < \alpha$  to obtain a lower bound for the joint event that  $\|T^{(s)}(n)\| < \alpha$  for all  $n = 1, \dots, N^*$ . This yields the bound

$$\left(1 - \frac{\epsilon}{4N^*}\right)^{N^*} \geq 1 - \epsilon,$$

where the last inequality follows from the assumption  $\epsilon < 1$  and from

$$(1 - y)^k \geq \exp(-4ky) \geq 1 - 4ky$$

for  $0 < y < 1/2$  and  $k \geq 1$ .

Let us now define the function

$$\Psi(\tau) = \tau + \rho \{-\tau + F(\tau)\}.$$

Since  $(1 - \rho)\tau$  and  $F(\tau)$  are Lipschitz-continuous on  $D$ , also  $\Psi(\tau)$  is Lipschitz-continuous, say with a Lipschitz constant  $L_0$ . Hence, with a probability of at least  $1 - \epsilon$ , for all  $n = 2, \dots, N^*$ ,

$$\begin{aligned} \|\tau(n) - \bar{\tau}(n)\| &= \|\Psi(\tau(n-1)) + \rho T^{(s)}(n-1) - \Psi(\bar{\tau}(n-1))\| \\ &\leq \rho \cdot \|T^{(s)}(n-1)\| + L_0 \cdot \|\tau(n-1) - \bar{\tau}(n-1)\| \leq \alpha + L_0 \cdot \|\tau(n-1) - \bar{\tau}(n-1)\|. \end{aligned}$$

Iterating this estimation and using the initial condition  $\tau(1) = \bar{\tau}(1)$  yields that with a probability of at least  $1 - \epsilon$ , for all  $n \leq N^*$ ,

$$\|\tau(n) - \bar{\tau}(n)\| \leq \alpha \cdot \frac{L_0^{n-1} - 1}{L_0 - 1} \leq \alpha \cdot \frac{L_0^{N^*-1} - 1}{L_0 - 1}.$$

By the special choice  $\alpha = \delta(L_0 - 1)/(L_0^{N^*-1} - 1)$ , we obtain the assertion of the theorem. □

For the application of Theorem 2.1, an important question is how the Lipschitz condition can be verified. As it will be seen in the following sections,  $F_{kl}(\tau)$  typically turns out as a continuously differentiable function on

$$D_0 = \{\tau \in [0, \infty^{|\mathcal{A}|}] \mid \sum_{l \in \mathcal{N}(k)} \tau_{kl} > 0 \text{ for all } k \text{ with } \mathcal{N}(k) \neq \emptyset\}. \quad (6)$$

Now, if  $F_{kl}(\tau)$  is continuously differentiable on some *compact* set  $D$ , then the Lipschitz property follows. Thus, the crucial question is whether or not the natural domain  $D_0$  of  $F_{kl}(\tau)$  can be restricted to a compact (i.e., bounded and closed) subset  $D$  containing all possible trajectories of original ACO process and of the ADDP. For the trajectories  $\tau(n)$  and  $\bar{\tau}(n)$  ( $n = 1, \dots, N^*$ ) within a fixed bounded time interval of  $N^*$  rounds, this question can be answered by yes: From (2), it is easy to show by induction that  $\tau_{kl}(n)$  is bounded above by

$$\tau_{max} = \max_{(k,l) \in \mathcal{A}} \tau_{kl}^{(0)} + \max_{x \in S} f(x).$$

Furthermore,  $\tau_{kl}(n)$  is bounded below by

$$\tau_{min}(N^*) = \min_{(k,l) \in \mathcal{A}} \tau_{kl}^{(0)} \cdot (1 - \rho)^{N^*} > 0.$$

as long as  $n \leq N^*$ . Therefore, the compact set

$$D = [\tau_{min}(N^*), \tau_{max}]^{|\mathcal{A}|}$$

satisfies the requirements above. It would be desirable to have a compact restriction  $D$  of  $D_0$  *not* depending on  $N^*$ , but the existence of such a restriction can only be shown for special construction graphs as, e.g., the “chain” graph defined in the next section.

To give a clearer picture of the fundamental dynamics of the ADDP, we consider the asymptotic case of an evaporation factor  $\rho$  near zero. This asymptotics is motivated by the fact that in theoretical convergence results, decreasing  $\rho$  increases the probability that convergence to the *optimal* solution takes place (see [16] – [18]). It is obvious that if  $\rho$  is reduced, such that pheromone changes in each round become small, a comparably larger number of rounds must be executed to get substantial changes. Therefore, in order to be able to plot the asymptotic behavior, it is convenient to re-scale the time axis in such a way that the product of  $\rho$  and the number  $M$  of rounds per time unit remains constant. Without loss of generality, we can assume that it is one, i.e., we assume that  $\rho = 1/M$ . In this scaling, a round takes  $dt = 1/M = \rho$  time units.

We now define the *associated continuous deterministic process* (ACDP) as the limit of the (re-scaled) ADDP as  $dt \rightarrow 0$ : With  $\tilde{\tau}(t)$  denoting the pheromone vector at time  $t$  in the re-scaled process, (5) yields:

$$\tilde{\tau}_{kl}(t + dt) = \tilde{\tau}_{kl}(t) + dt \cdot \{-\tilde{\tau}_{kl}(t) + F_{kl}(\tilde{\tau}(t))\}.$$

Letting  $dt \rightarrow 0$ , we obtain that the ACDP is given by the system

$$\frac{d\tilde{\tau}_{kl}}{dt} = F_{kl}(\tilde{\tau}) - \tilde{\tau}_{kl} \quad ((k, l) \in \mathcal{A}) \quad (7)$$

of ordinary differential equations, where  $\tilde{\tau} = (\tilde{\tau}_{kl})$  is the pheromone vector at time  $t$ , and  $F_{kl}(\tilde{\tau})$  is the expected passage fitness of arc  $(k, l)$  under  $\tilde{\tau}$ . We call (7) the *basic law* of the ACDP.

**Remark 1.** We do not make any assertion on convergence of the trajectories of the ACO process to those of the ACDP. Such a result would require a deeper analysis: In Theorem 2.1, we have assumed that the number of rounds is bounded by some fixed  $N^*$ . For investigating the asymptotic case  $dt \rightarrow 0$ , it would be necessary to estimate the tradeoff between a growing number of rounds on the one hand (for that, the influence of  $N^*$  on the accuracy of the approximation would have to be made explicit), and the diminishing value of  $\rho$  on the other hand. This type of analysis is beyond the scope of the present paper. A sort of “local convergence” could easily be derived from Theorem 2.1, but we conjecture that a stronger convergence result holds without being able to prove it at the moment.

**Remark 2.** The described re-scaling of the time axis allows to parallel the real ACO process with the ACDP. E.g., if  $\rho$  is chosen as 0.01, and a round takes  $10^{-5}$  seconds on a given computer, then a time unit is defined as  $10^{-5} \cdot M = 10^{-5} \cdot (1/\rho) = 10^{-3}$  seconds. If it is possible to compute the process trajectory of the ACDP, it can be used to tentatively *predict the convergence speed* of ACO via this scaling. For an example, see Fig. 5 and the remarks on it in Subsection 5.2.



**Remark 3.** In Theorem 2.1, we have described the ADDP (and hence finally also the ACDP) as a limiting case for a situation where the number  $s$  of ants tends to infinity. A closer consideration shows that for deriving the ACDP, it is probably not necessary to have  $s \rightarrow \infty$ ; it suffices to work with a fixed number  $s$  of ants and to split the number  $M$  of rounds per time unit into two factors  $M_1$  and  $M_2$  by combining  $M_1$  subsequent rounds to a *period* and executing  $M_2$  periods per time unit. By a straightforward analysis, it can be shown that as  $M_2 \rightarrow \infty$ , pheromone changes during each fixed period get infinitesimally small, such that the overall effect achieved in the considered period is in a first-order approximation the same as that of a single round with  $s \cdot M_1$  ants. Letting  $M_1 \rightarrow \infty$  produces then again the difference equation (5). We emphasize that this consideration is “heuristic”; a mathematical convergence result would have to make the indicated line of argumentation more precise.

### 3 Expected Fitness in the Associated Continuous Deterministic Process

In this section, we derive a simple relation allowing to compute the expected fitness of the walk of an ant, given a current pheromone vector  $\tilde{\tau}$ . We do this on the implicit assumption that  $\tilde{\tau}$  evolves over time according to the ACDP, i.e., under the basic law (7). Note that although the pheromone dynamics in the ACDP is deterministic, ants select their walks based on the current pheromone still in a stochastic fashion, except in degenerate cases.

For the sake of a simpler notation, we shall write  $\tau$  instead of  $\tilde{\tau}$  from now on, even though referring to the ACDP. Since the context makes it always clear whether  $\tau$  is considered as a function of round index  $n$  or of time parameter  $t$ , this cannot lead to misunderstandings.

**Proposition 3.1.** Let  $k$  be a node of the construction graph  $\mathcal{C}$  with the property that  $k$  must be traversed in *each* feasible walk, and that  $k$  cannot be the last node of the walk. By  $\mu(\tau) = E_\tau(f(X))$  we denote the *expected fitness* of the walk chosen by a fixed ant under pheromone vector  $\tau$ . Then,

$$\mu(\tau) = \varphi_k(\tau) + \dot{\varphi}_k(\tau), \quad (8)$$

where

$$\varphi_k(\tau) = \sum_{l \in \mathcal{N}(k)} \tau_{kl} \quad (9)$$

is the total pheromone flow out of node  $k$  (recall that  $\mathcal{N}(k)$  denotes the set of immediate successor nodes of  $k$  in the directed graph  $\mathcal{C}$ ), and  $\dot{\varphi}_k(\tau)$  denotes the derivative  $d\varphi_k(\tau(t))/dt$ .

**Proof.** From (7), we obtain by summation over all indices  $l \in \mathcal{N}(k)$  that

$$\frac{d\varphi_k}{dt} = \sum_{l \in \mathcal{N}(k)} F_{kl}(\tau) - \varphi_k(\tau). \quad (10)$$

Since  $k$  is traversed in each walk and must have a successor on each walk, the set of all walks  $x$  can be classified according to the node  $l \in \mathcal{N}(k)$  traversed after  $k$  on  $x$ . Hence by (4),

$$\begin{aligned}\mu(\tau) = E_\tau(f(X)) &= E_\tau \left( \sum_{l \in \mathcal{N}(k)} I((k, l) \in X) \cdot f(X) \right) = \sum_{l \in \mathcal{N}(k)} E_\tau (I((k, l) \in X) \cdot f(X)) \\ &= \sum_{l \in \mathcal{N}(k)} F_{kl}(\tau).\end{aligned}$$

Therefore (10) yields

$$\dot{\varphi}_k(\tau) = \mu(\tau) - \varphi_k(\tau).$$

□

**Remark.** The start node  $v_0$ , shortly denoted as node 0, is guaranteed to be traversed by each feasible walk, and we have assumed that it always has at least one feasible successor. Therefore, the equation

$$\mu(\tau) = \varphi_0(\tau) + \dot{\varphi}_0(\tau) \tag{11}$$

must hold.

**Proposition 3.2.** In a situation where pheromone has become stationary, i.e., where  $\dot{\tau}_{kl} = 0$  for each  $(k, l)$ , one has

$$\mu(\tau) = \varphi_k(\tau) \tag{12}$$

for each node  $k$  with the property that it is traversed by each feasible walk, and that  $k$  cannot be the last node on the walk. In particular, the expected passage fitness in a stationary situation is equal to the total pheromone flow out of the start node  $v_0$ .

**Proof.** Follows immediately from (8) and from the fact that  $\dot{\tau} = 0$  entails  $\dot{\varphi}_k = 0$ . □

**Example 3.1.** Consider the special case of a problem where all feasible paths  $x \in S$  have identical fitness  $f(x) = \mu_0 = \text{const}$ . In this case,  $\mu(\tau) = \mu_0$  for all  $\tau$ , such that (11) yields the inhomogeneous linear differential equation

$$y_0 + \dot{y}_0 = \mu_0 \tag{13}$$

for the function  $y_0 = y_0(t) = \varphi_0(\tau(t))$ . The general solution of (13) is

$$y_0 = \mu_0 + ce^{-t},$$

and for the initial value  $y_0(0) = y_0^{(0)}$ , one gets the special solution

$$y_0 = \mu_0 + (y_0^{(0)} - \mu_0)e^{-t}.$$

Thus, the pheromone flow  $y_0 = \varphi_0(\tau(t))$  out of the start node converges to the constant fitness  $\mu_0$  according to a negative-exponential function.

## 4 Three Types of Construction Graphs for Subset Problems

The theory developed above can be applied for several purposes. For example, the *convergence rate* of ACO for specific test functions, depending on the problem instance size, can be investigated. We leave this issue outside the scope of the present paper (results of this type will be presented in [21]), but focus instead on another possible application: a comparative theoretical evaluation of diverse construction graphs. To outline this application, let us consider the wide class of problems of *subset selection* type, i.e., problems where the feasible set is the set  $\{0, 1\}^m$  of binary vectors of length  $m$ , or, in other terms, the set of all subsets of a given collection of  $m$  items  $1, \dots, m$ . Examples are (linear or nonlinear) knapsack problems, max-clique problems, or binary portfolio selection problems.

Different types of construction graphs can be designed for the application of ACO to such problems. A main criterion for the decision on a suitable construction graph is whether or not there is an a-priori order between the items  $1, \dots, m$  among which a choice has to be made. If there is no such a-priori order, it is arbitrary which item gets which of the labels  $1, \dots, m$ , provided that item characteristics are assigned correctly to the labels. E.g., the three problem types mentioned above have this symmetry property. (An example for a problem lacking it is obtained in the case where the binary vector of length  $m$  is a binary encoding of an integer between 0 and  $2^m - 1$ ; here, bit positions have a pre-defined order.) In this paper, we restrict ourselves to the case where an a-priori order in which the items are arranged does not exist. In this case, also the construction graph encoding should be such that re-arrangements of the items (together with their characteristics) leave the results essentially invariant. More precisely, if item  $i$  is re-named as item  $\pi(i)$  ( $i = 1, \dots, m$ ), where  $\pi$  is some permutation of  $1, \dots, m$ , then the distribution of the states at time  $t$  of the resulting ACO process should differ from that in the original process only by the assignment between item labels and items. In the sequel, we describe three types of construction graphs satisfying this invariance condition. In each case, a solution is represented as a vector  $x = (x_1, \dots, x_m)$  with  $x_i \in \{0, 1\}$  ( $i = 1, \dots, m$ ), and the start node  $v_0$  of  $\mathcal{C}$  is the node  $\underline{0}$ .

1. *Chains*: The node set of  $\mathcal{C}$  consists of the nodes  $1, \dots, m$ , the nodes  $\neg 1, \dots, \neg m$  and auxiliary nodes  $\underline{0}, \dots, \underline{m}$ . The arc set consists of the arcs  $(\underline{i-1}, i)$ ,  $(\underline{i-1}, \neg i)$ ,  $(i, \underline{i})$  and  $(\neg i, \underline{i})$  ( $i = 1, \dots, m$ ). The choice of node  $i$  resp. node  $\neg i$  means that  $x_i = 1$  resp.  $x_i = 0$ . An example is shown in Fig. 2.
2. *Disks*: The node set of  $\mathcal{C}$  consists of the nodes  $\underline{0}, \underline{1}$  and the nodes  $1, \dots, m$ . The arc set consists of the arcs  $(\underline{0}, i)$ , the arcs  $(i, \underline{1})$  ( $i = 1, \dots, m$ ), the arcs  $(i, j)$  ( $i \neq j$ ), and the arc  $(\underline{0}, \underline{1})$ . Choice of node  $i$  means that  $x_i = 1$ . If  $i$  is not chosen during the walk,  $x_i = 0$ . An example is shown in Fig. 3.
3. *Drums*: The node set of  $\mathcal{C}$  consists of the node  $\underline{0}$ , the nodes  $1, \dots, m$  and the nodes  $\neg 1, \dots, \neg m$ . The arc set consists of the arcs  $(\underline{0}, i)$ , the arcs  $(\underline{0}, \neg i)$  ( $i = 1, \dots, m$ ), the arcs  $(i, j)$  ( $i \neq j$ ), the arcs  $(\neg i, \neg j)$  ( $i \neq j$ ), the arcs  $(i, \neg j)$  ( $i \neq j$ ) and the arcs  $(\neg i, j)$  ( $i \neq j$ ). As an additional constraint, both node  $i$  and  $\neg i$  are forbidden after node  $i$  or  $\neg i$  has been visited, i.e.,  $\mathcal{V}(u)$  contains only those nodes  $i$  and  $\neg i$  that do not occur in  $u$ , neither in their positive form  $i$  nor in their negated form  $\neg i$  (cf. Subsection 2.1). Interpretations are as in the *chain* case. An example is shown in Fig. 4.

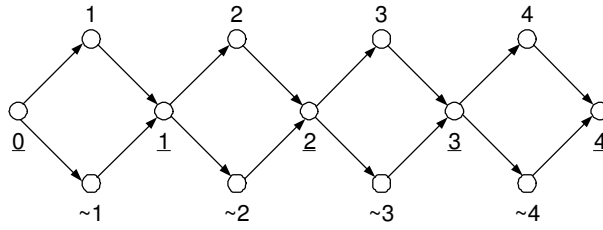


Fig. 2. Chain construction graph for  $m = 4$ .

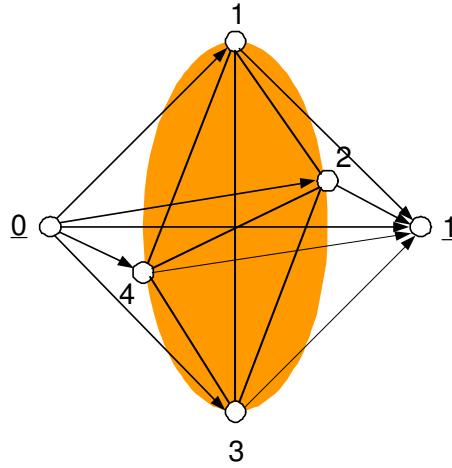


Fig. 3. Disk construction graph for  $m = 4$ . Undirected links are bidirectional, i.e., each of them corresponds to two directed arcs with opposite orientation.

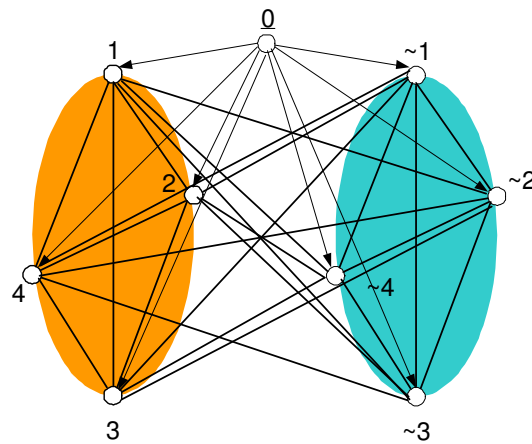


Fig. 4. Drum construction graph for  $m = 4$ . Undirected links are bidirectional, i.e., each of them corresponds to two directed arcs with opposite orientation.

## 5 Comparisons of the Construction Graphs on Test Functions

In this section, we compare the three construction graphs introduced in Section 4 on some basic test functions: indicator functions with one or two optima, and the well-known OneMax test function (see, e.g., [31]). All statements refer to the ACDP.

### 5.1 Indicator Functions

One of the most elementary fitness functions that can be imagined as a test function is an indicator function

$$f(x) = I(x \in S^*)$$

stating whether a solution  $x$  from a given set  $S^* \subseteq S$  of “good solutions” has been found or not. We will derive results on the expected fitness  $\mu(\tau) = E_\tau(f(X))$  for some test functions from this class on the three construction graphs. Let us remark that the emphasis in this context is not on how fast a “good” solution will eventually be *found* (without additional hint, a heuristic can perform this task only by trial and error anyway), but rather on how efficient the heuristic is in *sticking to* already seen good solutions. The reason why the last-mentioned aspect is important lies in the fact that good overall solutions are often composed of good partial solutions: If, e.g.,  $f$  is given as  $f(x) = g(f_1(x|A_1), \dots, f_K(x|A_K))$ , where  $g$  is an increasing function of  $K$  real variables,  $A_1, \dots, A_K$  forms a partition of the set  $\{1, \dots, m\}$  of bit positions,  $x|A_k \in \{0, 1\}^{|A_k|}$  is derived from  $x \in \{0, 1\}^m$  by restriction to the bit positions contained in  $A_k$ , and  $f_k : \{0, 1\}^{|A_k|} \rightarrow [0, \infty[$  is a fitness function on the binary vectors defined on the block  $A_k$  ( $k = 1, \dots, K$ ), then for approaching the optimum of  $f$  quickly, it is not sufficient that the partial optimizers  $(x_k|A_k)^*$  of  $F_k$  ( $k = 1, \dots, K$ ) are found at some times by chance, unless they are *preserved* after having been found. So, ideally, the expected fitness on each block  $A_k$  should continually increase and converge (quickly) to its maximum possible value.

#### 5.1.1 Indicator functions with a single optimizer

In Subsection 5.1.1, we assume that  $|S^*| = 1$ , i.e., that  $S^*$  consists of a single optimal solution  $x^*$ , and choose the test function as  $f(x) = I(x = x^*)$ .

##### (a) Behavior of the chain

Consider a chain with  $m$  links. Because of symmetry between “up moves” and “down moves” on the chain and their mutual independence, we can restrict ourselves without loss of generality to the special case  $x^* = (1, 1, \dots, 1)$ , i.e., we consider the fitness function  $f(x) = I(x_1 = \dots = x_m = 1)$ . Under these circumstances, there is complete symmetry between the  $m$  chain links. Therefore, at each time  $t$ , all pheromone values  $\tau_{\underline{i-1}, i}$  ( $i = 1, \dots, m$ ) will be identical; for abbreviation, we set  $u = u(t) = \tau_{\underline{i-1}, i}$ . Analogously, at each time  $t$ , all all pheromone values  $\tau_{\underline{i-1}, \neg i}$  ( $i = 1, \dots, m$ ) will be identical; we set  $w = w(t) = \tau_{\underline{i-1}, \neg i}$ . We write  $F_u$  resp.  $F_w$  instead of  $F_{\underline{i-1}, i}$  resp.  $F_{\underline{i-1}, \neg i}$ . The expected passage fitness

of arc  $(\underline{0}, 1)$  is immediately obtained as

$$F_u = \left( \frac{u}{u+w} \right)^m,$$

and the expected passage fitness of arc  $(\underline{0}, \neg 1)$  is  $F_w = 0$ . Thus, the ACDP is given by the system

$$\dot{u} = \left( \frac{u}{u+w} \right)^m - u, \quad \dot{w} = -w \quad (14)$$

of ordinary differential equations. The second equation of (14) has solution  $w = w_0 e^{-t}$ , where  $w_0$  is the initial value for  $w$  at time  $t = 0$ , hence the first equation of (14) reduces to a differential equation in one variable,

$$\dot{u} = (1 + w_0 e^{-t} u^{-1})^{-m} - u. \quad (15)$$

By Proposition 3.1, the expected fitness of an ant's solution evolves over time according to

$$\mu = (u+w) + (\dot{u} + \dot{w}) = \left( \frac{u}{u+w} \right)^m = (1 + w_0 e^{-t} u^{-1})^{-m}. \quad (16)$$

By (14),

$$\frac{d}{dt} \frac{w}{u} = -\frac{w}{u^2} \left( \frac{u}{u+w} \right)^m \leq 0,$$

hence (14) yields  $\dot{u} \geq \gamma - u$  with  $\gamma = (1 + w_0/u_0)^{-m} > 0$ , where  $u_0$  is the initial value for  $u$  at time  $t = 0$ . Thus, if  $u \leq \gamma/2$ , we have  $\dot{u} > 0$ , and therefore  $u$  cannot become smaller than  $\gamma/2$ . As a consequence, (16) implies  $\mu \rightarrow 1$  as  $t \rightarrow \infty$ .

We see that the chain is *asymptotically efficient* for the considered test function in the sense that the expected fitness tends to its optimal value.

(b) *Behavior of the disk*

Interestingly, the disk construction graph already fails at the very simple test function  $f(x) = I(x = x^*)$ . To see this, let us consider the special case  $x_1^* = x_2^* = 1$ ,  $x_i^* = 0$  ( $i = 3, \dots, m$ ). (Contrary to the chain, the disk construction graph does not treat the decisions  $x_i = 1$  and  $x_i = 0$  in a symmetric way, so we cannot restrict ourselves here without loss of generality to the special case  $x^* = (1, \dots, 1)$ .) For abbreviation, set  $u = \tau_{\underline{0},1} = \tau_{\underline{0},2}$ ,  $y = \tau_{1,\underline{1}} = \tau_{2,\underline{1}}$ ,  $z = \tau_{1,2} = \tau_{2,1}$ ,  $a = \tau_{\underline{0},i}$  for  $i = 3, \dots, m$ ,  $b = \tau_{i,\underline{1}}$  for  $i = 3, \dots, m$ ,  $c = \tau_{i,j}$  for  $i, j \in \{3, \dots, m\}$  and  $i \neq j$ ,  $d = \tau_{i,j}$  for  $i \in \{1, 2\}$  and  $j \in \{3, \dots, m\}$ ,  $e = \tau_{i,j}$  for  $i \in \{3, \dots, m\}$  and  $j \in \{1, 2\}$ , and  $g = \tau_{\underline{0},\underline{1}}$ . (Therein, equations between  $\tau$ -values hold due to symmetry.) Similarly as for the chain under (a), we abbreviate the corresponding expected passage fitness values by  $F_u, F_y$ , etc.

Since, as it is easily seen,  $F_a = F_b = F_c = F_d = F_e = F_g = 0$ , in a stationary situation,  $a = b = c = d = e = g$  must hold. Therefore, Proposition 3.2 yields that

$$\mu = 2u + (m-2)a + g = 2u \quad (17)$$

in the stationary case. Moreover, the expected passage fitness of arc  $(\underline{0}, 1)$  is easily computed as

$$F_u = \frac{u}{2u + (m-2)a + g} \cdot \frac{z}{z + y + (m-2)d} \cdot \frac{y}{y + (m-2)d},$$

which reduces to

$$F_u = \frac{1}{2} \cdot \frac{z}{z+y} \quad (18)$$

in the stationary case. Similarly, it turns out that in the stationary case, the expected passage fitness values  $F_z$  and  $F_y$  of arc  $(1, 2)$  and of arc  $(1, \underline{1})$ , respectively, become equal to the r.h.s. of (18) as well. (Note that  $(\underline{0}, 2, 1, \underline{1})$  is the only path with positive fitness containing  $(1, \underline{1})$ ). Therefore, by (7), in steady state

$$u = \frac{1}{2} \cdot \frac{z}{z+y}, \quad z = 0 \text{ or } 2(z+y) = 1, \quad \text{and} \quad y = \frac{1}{2} \cdot \frac{z}{z+y}. \quad (19)$$

From the last equation, we get  $z = 2y(z+y)$  which must be equal to  $y$  by the equation before, unless  $z = 0$ . The first equation of (19) and eq. (17) leave  $u = 1/4$  and hence  $\mu = 1/2$  as the only possible stationary values for which  $\mu > 0$ . Therefore, the disk is *asymptotically inefficient* at the considered test problem: the expected fitness needs not to converge to its optimal value.

(b) *Behavior of the drum*

As in the case of the chain, we can assume  $x^* = (1, \dots, 1)$  without loss of generality. Let  $u = \tau_{0,i}$ ,  $w = \tau_{0,-i}$  ( $i = 1, \dots, m$ ),  $x = \tau_{i,j}$ ,  $y = \tau_{i,-j}$ ,  $z = \tau_{-i,j}$  and  $s = \tau_{-i,-j}$  ( $i = 1, \dots, m$ ;  $j = 1, \dots, m$ ;  $i \neq j$ ). One finds

$$F_u = F_x = \frac{u}{m(u+w)} \left( \frac{x}{x+y} \right)^{m-1}, \quad (20)$$

and  $F_w = F_y = F_z = F_s = 0$ . The equation for  $F_x$  follows from the facts that (a) the probability that the current walk has fitness 1 is

$$\frac{u}{u+w} \left( \frac{x}{x+y} \right)^{m-1},$$

and that (b) out of all paths with fitness 1, a fraction of  $(m-1)(m-2)!/m! = 1/m$  contains a *fixed* given arc of type  $(i, j)$ , say arc  $(1, 2)$ . Using  $F_u = F_x$ , which implies  $\dot{u} - \dot{x} = x - u$  and hence  $u - x = (u_0 - x_0)e^{-t}$ , the equations in (20) can be combined to a single differential equation for  $u$  or  $x$ .

In the stationary situation, we have  $w = y = z = s = 0$ , and hence (from  $\dot{u} = 0$  and  $\dot{x} = 0$ )

$$u = \frac{1}{m} \quad \text{and} \quad x = \frac{1}{m}.$$

The stationary value of  $\mu$  is therefore  $\mu = m(u+w) = 1$ . Thus, the drum is asymptotically efficient for the considered test function.

### 5.1.2 Indicator functions with two optimizers

Let us now consider the case  $|S^*| = 2$  where  $S^*$  consists of two (equally good) optimizers  $x^{(1)} = (x_1^{(1)}, \dots, x_m^{(1)})$  and  $x^{(2)} = (x_1^{(2)}, \dots, x_m^{(2)})$ , such that  $f(x) = I(x = x^{(1)} \text{ or } x = x^{(2)})$ . It turns out that in this case, the behavior of the ACDP heavily depends on how similar the two solutions  $x^{(1)}$  and  $x^{(2)}$  are. Measuring similarity by the Hamming distance

$$d(x^{(1)}, x^{(2)}) = \sum_{i=1}^m I(x_i^{(1)} \neq x_i^{(2)}),$$

we focus the analysis on the two extreme cases  $d(x^{(1)}, x^{(2)}) = 1$  and  $d(x^{(1)}, x^{(2)}) = m$ . In view of the results of the previous subsection, we restrict ourselves to chain and drum. It is easy to see that for these types of construction graphs, no loss of generality is caused by the assumption that  $x^{(1)} = (1, 1, 1, \dots, 1)$  and  $x^{(2)} = (0, 1, 1, \dots, 1)$  in the case of Hamming distance 1, resp. by the assumption that  $x^{(1)} = (1, 1, 1, \dots, 1)$  and  $x^{(2)} = (0, 0, 0, \dots, 0)$  in the case of Hamming distance  $m$ . Our main interest will be the question whether the construction graphs are asymptotically efficient or not.

(a) *Behavior of the chain*

(a1) *Minimal Hamming distance*

For  $x^{(1)}$  and  $x^{(2)}$  as defined above for Hamming distance 1, the test function becomes  $f(x) = I(x_2 = x_3 = \dots = x_m = 1)$ . Let  $a = \tau_{0,1}$ ,  $b = \tau_{0,-1}$ ,  $c = \tau_{i,i+1}$ ,  $d = \tau_{i,-(i+1)}$  ( $i = 1, \dots, m-1$ ). One finds

$$\begin{aligned} F_a &= \frac{a}{a+b} \cdot \left( \frac{c}{c+d} \right)^{m-1}, \\ F_b &= \frac{b}{a+b} \cdot \left( \frac{c}{c+d} \right)^{m-1}, \\ F_c &= \left( \frac{c}{c+d} \right)^{m-1}, \end{aligned}$$

and  $F_d = 0$ . Therefore, in steady state,  $d = 0$ ,  $c = 1$ , and  $a + b = 1$ , provided that  $(a, b) \neq (0, 0)$ . Because of  $\dot{a} + \dot{b} = (c/(c+d))^{m-1} - (a+b)$ , the point  $(a, b) = (0, 0)$  is repelling. Thus, in a stable fixed point,  $\mu = a + b = 1$  must hold, which is optimal.

(a2) *Maximal Hamming distance*

For  $x^{(1)}$  and  $x^{(2)}$  as defined above for the Hamming distance  $m$ , the test function becomes  $f(x) = I(x_1 = x_2 = \dots = x_m)$ . With the abbreviations  $a = \tau_{i-1,i}$  and  $b = \tau_{i-1,-i}$  ( $i = 1, \dots, m$ ), one finds

$$F_a = \left( \frac{a}{a+b} \right)^m \quad \text{and} \quad F_b = \left( \frac{b}{a+b} \right)^m,$$

such that for the stationary case  $\dot{a} = \dot{b} = 0$ , the conditions

$$\begin{aligned} a = 0 \quad \text{or} \quad a^{m-1} &= (a+b)^m, \\ b = 0 \quad \text{or} \quad b^{m-1} &= (a+b)^m \end{aligned}$$

have to be satisfied, which imply either  $a = b = 0$  or  $a = b = 1/2^m$ . As a consequence,  $\mu = a+b = 1/2^{m-1}$ , except in the degenerate case  $a = b = 0$ , where  $\mu = 0$ . We see that the chain is here asymptotically (highly) inefficient. The reason is that in the ACDP, the chain does not manage the symmetry break between the two solutions. (Although they refer to a different pheromone update scheme, the results in [18] indicate that the original ACO process could possibly achieve such a symmetry break by means of its random perturbations, albeit at the price of a very slow convergence. We must skip a discussion of this interesting question for the sake of brevity.)



(b) *Behavior of the drum*

(b1) *Minimal Hamming distance*

Again, let  $f(x) = I(x_2 = x_3 = \dots = x_m = 1)$ , and let  $u = \tau_{\underline{0},1} = \tau_{\underline{0},\neg 1}$ ,  $w = \tau_{\underline{0},i}$ ,  $z = \tau_{\underline{0},\neg i}$ ,  $a = \tau_{1,i} = \tau_{\neg 1,i}$ ,  $b = \tau_{1,\neg i} = \tau_{\neg 1,\neg i}$ ,  $a' = \tau_{i,1} = \tau_{i,\neg 1}$ ,  $b' = \tau_{\neg i,1} = \tau_{\neg i,\neg 1}$  ( $i = 2, \dots, m$ ),  $x = \tau_{i,j}$ ,  $y = \tau_{i,\neg j}$ ,  $y' = \tau_{\neg i,j}$ ,  $s = \tau_{\neg i,\neg j}$  ( $i, j \in \{2, \dots, m\}, i \neq j$ ). We show that the drum is asymptotically efficient, i.e., that  $\mu$  must take its optimal value 1 in a stationary situation. To see this, observe that

$$\mu = 2u + (m-1)(w+z).$$

In the stationary case,  $z = b = b' = y = y' = s = 0$  by analogous arguments as in subsection 5.1.1. Distinguish two possible cases for the stationary situation:

1.  $u > 0$ : We compute  $F_u$  by considering (without loss of generality) the arc  $(\underline{0}, 1)$ . Given that node 1 has been reached, the walk is continued with probability one in such a way that the fitness value 1 is reached, since  $b = y = 0$ . Therefore,

$$F_u = \frac{u}{2u + (m-1)(w+z)} \cdot 1 = \frac{u}{\mu}.$$

From  $0 = \dot{u} = F_u - u$ , we get  $\mu = 1$ .

2.  $u = 0$ : In this case, we compute  $F_w$  by considering (without loss of generality) the arc  $(\underline{0}, 2)$ . We obtain

$$F_w = \frac{w}{2u + (m-1)(w+z)} = \frac{1}{m-1}.$$

From  $0 = \dot{w} = F_w - w$ , one gets  $w = 1/(m-1)$  and hence

$$\mu = 2u + (m-1)(w+z) = 1,$$

which shows the assertion.

(b2) *Maximal Hamming distance*

As before, let  $f(x) = I(x_1 = x_2 = \dots = x_m)$ . Setting  $u = \tau_{\underline{0},i} = \tau_{\underline{0},\neg i}$ ,  $x = \tau_{i,j} = \tau_{\neg i,\neg j}$  and  $y = \tau_{i,\neg j} = \tau_{\neg i,j}$ , we obtain

$$F_u = F_x = \frac{1}{2m} \left( \frac{x}{x+y} \right)^{m-1},$$

and  $F_y = 0$ . The equation for  $F_x$  follows from the fact that out of the  $2m!$  equiprobable paths with fitness 1, there are  $(m-1)(m-2)! = (m-1)!$  paths containing a fixed arc  $(i, j)$  or  $(\neg i, \neg j)$ , say arc  $(1, 2)$ , and that the probability that the walk has fitness 1 is  $(x/(x+y))^{m-1}$ .

In the non-stationary case,

$$\dot{x} = \frac{1}{2m} \left( \frac{x}{x+y} \right)^{m-1} - x,$$

and  $\dot{y} = -y$ , such that  $y = y_0 e^{-t}$ , hence  $x$  satisfies the differential equation

$$\dot{x} = \frac{1}{2m} (1 + y_0 e^{-t} x^{-1})^{-m+1} - x$$

similar to (15). Furthermore,

$$\mu = 2m(u + \dot{u}) = 2mF_u = \left( \frac{x}{x+y} \right)^{m-1} = (1 + y_0 e^{-t} x^{-1})^{-m+1}.$$

Analogously as in Subsection 5.1.1 (a), the case  $x \rightarrow 0$  as  $t \rightarrow \infty$  can be excluded, so by the last equation,  $\mu \rightarrow 1$  as  $t \rightarrow \infty$ , i.e., asymptotic efficiency holds. We see that in this setting, the *drum* outperforms the *chain*.

## 5.2 The OneMax Function

In this section, we deal with the so-called OneMax fitness function,

$$f(x) = \sum_{i=1}^m x_i,$$

i.e., the aim is now to maximize the number of 1-bits. Contrary to the test functions in Subsection 5.1, the OneMax function gives “guidance” for an evolution of the search process towards the maximizer  $x^* = (1, \dots, 1)$ .

As in Subsection 5.1.2, we consider only *chain* and *drum*, in view of the elementary inefficiency of the *disk* detected in Subsection 5.1.1.

### (a) Behavior of the chain

The OneMax function preserves symmetry between the  $m$  links of the chain, so it suffices to consider the two values  $u = \tau_{i-1, i}$  and  $w = \tau_{i-1, \neg i}$  ( $i = 1, \dots, m$ ). The passage fitness can be computed as follows: Let us call a bit  $x_i$  with value  $x_i = 1$  a *hit*, such that the fitness is the number of hits. For each  $i$ , the probability of a hit is  $u/(u+w)$ . Hence, the expected number of hits is  $m \cdot u/(u+w)$ , the passage fitness of arc  $(\underline{0}, 1)$  (i.e., the probability that  $(\underline{0}, 1)$  is traversed, multiplied by the expected number of hits in this case) is

$$F_u = \frac{u}{u+w} \cdot \left( \frac{u}{u+w}(m-1) + 1 \right) = \frac{u(mu+w)}{(u+w)^2}, \quad (21)$$

and the passage fitness of arc  $(\underline{0}, \neg 1)$  is

$$F_w = \frac{w}{u+w} \cdot \frac{u}{u+w}(m-1) = \frac{(m-1)uw}{(u+w)^2}. \quad (22)$$

We have

$$\mu = F_u + F_w = \frac{mu}{u+w}.$$

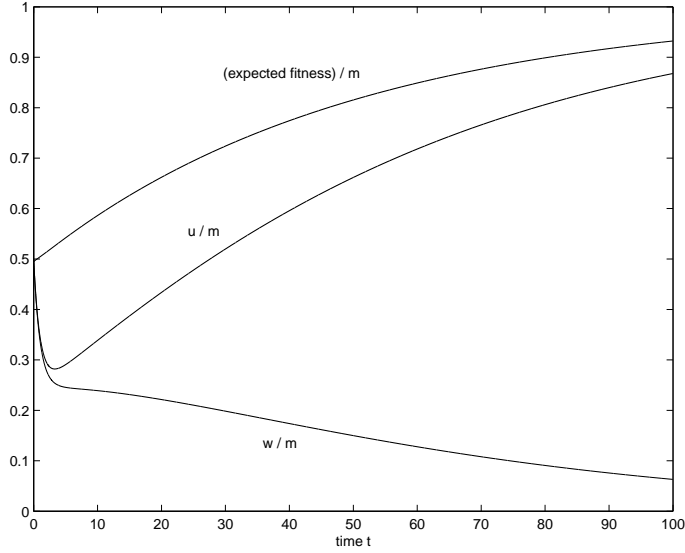
Solving  $\dot{u} = F_u - u = 0$  and  $\dot{w} = F_w - w = 0$  yields either  $u = w = 0$  or  $u = m$  and  $w = 0$ . It is easy to see that the first possible fixed point  $(0, 0)$  is repelling: Straightforward calculation shows that

$(d/dt)(w/u) \leq 0$  and hence  $w/u \leq w_0/u_0$ , where  $u_0$  and  $w_0$  are the initial values of  $u$  and  $w$ , respectively. Therefore, with  $\zeta = u + w$ , we obtain

$$\dot{\zeta} = \dot{u} + \dot{w} = (F_u - F_w) - (u + w) = \frac{m}{1 + w/u} - \zeta \geq \gamma - \zeta$$

with  $\gamma = m/(1 + w_0/u_0) > 0$ . Thus, if  $\zeta \leq \gamma/2$ , we have  $\dot{\zeta} > 0$ , which means that  $\zeta$  cannot reach the value 0. So  $(u, w)$  can only converge to the second possible fixed point  $(m, 0)$ , where  $\mu = m$ . Since  $\max f(x) = m$ , this means that the chain is asymptotically efficient for the OneMax function.

Equations (21) and (22) enable a numerical solution of (7). Fig. 5 shows the resulting dynamics for  $m = 50$  over a period of 100 time units, starting with initial values  $u_0 = w_0 = m/2 = 25$ . (The initial value  $m/2$  for each of the two variables has been chosen since the stationary value of  $u + v$  is  $m$ .) The uppermost curve shows that the expected fitness reaches 90 % of its optimal value in about 80 time units. With the numbers assumed in Remark 2 of Section 2, e.g., these would be 0.08 seconds.



*Fig. 5.* Dynamics of the chain for OneMax,  $m = 50$  and  $u_0 = w_0 = 25$ . The current expected fitness and the variables  $u$  and  $v$  are plotted as relative values, normalized by  $m = 50$ , which is the maximum of the objective function.

(a) *Behavior of the drum*

Again using symmetry, it suffices to consider the values  $u = \tau_{\underline{0},i}$  and  $w = \tau_{\underline{0},\neg i}$  ( $i = 1, \dots, m$ ) together with the values  $x = \tau_{i,j}$ ,  $y = \tau_{i,\neg j}$ ,  $z = \tau_{\neg i,j}$  and  $s = \tau_{\neg i,\neg j}$  ( $i = 1, \dots, m$ ;  $j = 1, \dots, m$ ;  $i \neq j$ ). Omitting the start node  $\underline{0}$ , we can describe a walk  $W$  by a sequence  $(W_1, \dots, W_m)$ , where  $W_i \in \{\pi(i), \neg\pi(i)\}$  with a permutation  $\pi$  of  $\{1, \dots, m\}$ . By the symmetry of the OneMax function, all  $m!$  permutations  $\pi$  occur with the same probabilities in the walks on the drum. Let us fix a pair of different integers from  $\{1, \dots, m\}$ , say  $(1, 2)$ . We classify the possible occurrences of  $(1, 2)$  as two successive elements in  $\pi$  by the position  $i$  of the first element 1 ( $1 \leq i \leq m - 1$ ). Obviously, the probability that  $(\pi(i), \pi(i + 1)) = (1, 2)$

is

$$\frac{(m-2)!}{m!} = \frac{1}{m(m-1)}.$$

Note that in the case  $(\pi(i), \pi(i+1)) = (1, 2)$ , element 1 may occur in original or in negated form (as 1 resp. as  $-1$ ) in the walk  $W$ ; the same holds for element 2.

To compute the expected passage fitness values of  $(1, 2)$ ,  $(1, -2)$ ,  $(-1, 2)$  and  $(-1, -2)$  as well as those of  $(\underline{0}, 1)$  and  $(\underline{0}, -1)$ , we use an *auxiliary graph*  $H = H_m$ , as depicted in Fig. 6. Therein, the nodes of the drum are re-ordered according to the currently chosen permutation  $\pi$  of node indices, such that the (irrelevant) “names” of the items among which the algorithm selects are factored out. Node  $K_0$  in  $H$  corresponds to node  $\underline{0}$  in  $\mathcal{C}$ . Transition from  $K_0$  to  $K_1^{(1)}$  resp.  $K_1^{(0)}$  in  $H$  corresponds to transition from  $\underline{0}$  to  $\pi(1)$  resp. to  $-\pi(1)$  in  $\mathcal{C}$ , transition from  $K_i^{(1)}$  to  $K_{i+1}^{(1)}$  resp. to  $K_{i+1}^{(0)}$  corresponds to transition from  $\pi(i)$  to  $\pi(i+1)$  resp. to  $-\pi(i+1)$ , and transition from  $K_i^{(0)}$  to  $K_{i+1}^{(1)}$  resp. to  $K_{i+1}^{(0)}$  corresponds to transition from  $-\pi(i)$  to  $\pi(i+1)$  resp. to  $-\pi(i+1)$ . To each feasible walk in  $\mathcal{C}$ , there corresponds a complete directed walk from  $K_0$  to  $K_{m+1}$  in  $H$ .

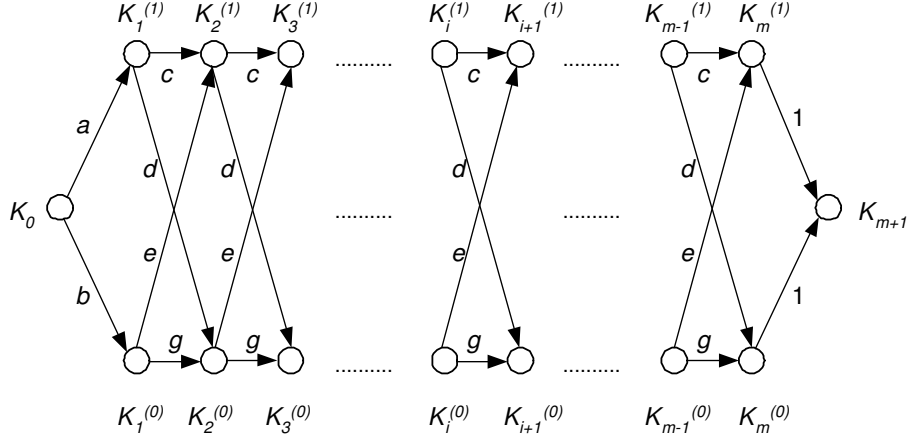


Fig. 6. Auxiliary graph  $H = H_m$ .

We use the following abbreviations for the transition probabilities, computed from the pheromone values  $u, w, x, y, z$  and  $s$  (cf. Fig. 6):

$$\begin{aligned} a &= u/(u+w), & b &= w/(u+w) & (a+b=1), \\ c &= p_{11} = x/(x+y), & d &= p_{10} = y/(x+y) & (c+d=1), \\ e &= p_{01} = z/(z+s), & g &= p_{00} = s/(z+s) & (e+g=1). \end{aligned}$$

Furthermore, we introduce the following notation: For a partial walk  $W$  in  $H$ , let  $U(W)$  denote the number of 1-bits in  $W$ , and let  $P(W)$  denote the probability that the partial walk  $W$  is chosen, given the random walk starts in the first node of  $W$ . For two nodes  $K, K'$  in  $W$ , let

$$\mathcal{A}(K, K') = \text{set of complete walks in } H \text{ that traverse arc } (K, K'),$$

and

$$\mathcal{B}(K, K') = \text{set of partial walks in } H \text{ starting in } K \text{ and ending in } K'.$$

We distinguish six different types of arcs in  $H$  (the two arcs not covered by this classification are not relevant for the analysis):

- type (1): arc  $(K_0, K_1^{(1)})$ ,
- type (0): arc  $(K_0, K_1^{(0)})$ ,
- types  $(k_1, k_2)$  with  $k_1 \in \{0, 1\}$ ,  $k_2 \in \{0, 1\}$ : arcs  $(K_i^{(k_1)}, K_{i+1}^{(k_2)})$ .

The arcs in  $\mathcal{C}$  are classified according to the types of the corresponding arcs in  $H$ , e.g., arc  $(i, -j)$  is of type  $(1, 0)$ .

By  $F_{(1)}$ ,  $F_{(0)}$  and  $F_{(k_1, k_2)}$ , we denote the expected passage fitness (in  $\mathcal{C}$ ) of an arc of type (1), (0) and  $(k_1, k_2)$ , respectively. Finally, for a complete random walk  $W$  in  $H_i$  ( $i \leq m$ ), we set:

$$\begin{aligned} \phi_i^{[k]} &= Pr\{W \text{ contains } K_i^{(k)}\} \quad (k = 1, 0), \\ \psi_i^{[k]} &= E(U(W) \cdot I(W \text{ contains } K_i^{(k)})) \quad (k = 1, 0), \\ \chi_i^{[j]} &= E(U(W) \mid W \text{ contains } K_1^{(j)}) \quad (j = 1, 0), \\ \phi_i^{(j, k)} &= Pr\{W \text{ contains } K_i^{(k)} \mid W \text{ contains } K_1^{(j)}\} \quad (j = 1, 0; k = 1, 0), \\ \psi_i^{(j, k)} &= E(U(W) \cdot I(W \text{ contains } K_i^{(k)}) \mid W \text{ contains } K_1^{(j)}) \quad (j = 1, 0; k = 1, 0). \end{aligned}$$

Obviously,

$$\phi_i^{[k]} = a\phi_i^{(1, k)} + b\phi_i^{(0, k)} \quad (23)$$

and

$$\psi_i^{[k]} = a\psi_i^{(1, k)} + b\psi_i^{(0, k)}. \quad (24)$$

It is immediately seen that

$$F_{(1)} = \frac{1}{m} a\chi_m^{[1]}, \quad F_{(0)} = \frac{1}{m} b\chi_m^{[0]}. \quad (25)$$

The factor  $1/m$  results from the fact that for a fixed arc  $(\underline{0}, i)$  or  $(\underline{0}, -i)$  of type 1 or 0, respectively, the probability that  $\pi(1) = i$  is  $1/m$ .

Somewhat more complicated is the computation of the remaining passage fitness values. We fix two different indices, say 1 and 2, classify the possible occurrences of  $(1, 2)$  in  $\pi$  according to the (possible) position  $i$  of 1 (as described above) and recall that the probability for  $(\pi(i), \pi(i+1)) = (1, 2)$  is  $1/(m(m-1))$ . By this consideration, we obtain

$$F_{(k_1, k_2)} = \sum_{i=1}^{m-1} \frac{1}{m(m-1)} F_{(k_1, k_2)}^{(i)}$$

where

$$\begin{aligned}
F_{(k_1, k_2)}^{(i)} &= E \left( \text{passage fitness of } (K_i^{(k_1)}, K_{i+1}^{(k_2)}) \text{ in } H_m \right) = \sum_{W \in \mathcal{A}(K_i^{(k_1)}, K_{i+1}^{(k_2)})} P(W) \cdot U(W) \\
&= \sum_{W_1 \in \mathcal{B}(K_0, K_i^{(k_1)})} \sum_{W_2 \in \mathcal{B}(K_{i+1}^{(k_2)}, K_{m+1})} P(W_1) \cdot p_{k_1 k_2} \cdot P(W_2) (U(W_1) + U(W_2)) \\
&= p_{k_1 k_2} \cdot \left\{ \sum_{W_1 \in \mathcal{B}(K_0, K_i^{(k_1)})} \sum_{W_2 \in \mathcal{B}(K_{i+1}^{(k_2)}, K_{m+1})} [P(W_1)P(W_2)U(W_1) + P(W_1)P(W_2)U(W_2)] \right\} \\
&= p_{k_1 k_2} \cdot \left\{ \sum_{W_2 \in \mathcal{B}(K_{i+1}^{(k_2)}, K_{m+1})} P(W_2) \cdot \sum_{W_1 \in \mathcal{B}(K_0, K_i^{(k_1)})} P(W_1)U(W_1) \right. \\
&\quad \left. + \sum_{W_1 \in \mathcal{B}(K_0, K_i^{(k_1)})} P(W_1) \cdot \sum_{W_2 \in \mathcal{B}(K_{i+1}^{(k_2)}, K_{m+1})} P(W_2)U(W_2) \right\} \\
&= p_{k_1 k_2} \left( 1 \cdot \psi_1^{[k_1]} + \phi_i^{[k_1]} \cdot \chi_{m-i}^{[k_2]} \right).
\end{aligned}$$

What remains to be done is the computation of the numbers  $\phi_i^{[k]}$ ,  $\psi_i^{[k]}$  and  $\chi_i^{[j]}$ .

*Computation of the numbers  $\phi_i^{[k]}$ :* We use (23). Straightforward considerations show that the recursion

$$\phi_i^{(1,1)} = c\phi_{i-1}^{(1,1)} + d\phi_{i-1}^{(0,1)}, \quad \phi_i^{(0,1)} = e\phi_{i-1}^{(1,1)} + g\phi_{i-1}^{(0,1)}$$

holds for  $i \geq 2$ , and that

$$\phi_1^{(1,1)} = 1, \quad \phi_1^{(0,1)} = 0.$$

By application of the theory of linear difference equations, we obtain the following solution:

$$\begin{aligned}
\phi_i^{(1,1)} &= \frac{e + d(g-d)^{i-1}}{d+e}, & \phi_i^{(0,1)} &= \frac{e - e(g-d)^{i-1}}{d+e}, \\
\phi_i^{(1,0)} &= \frac{d - d(c-e)^{i-1}}{d+e}, & \phi_i^{(0,0)} &= \frac{d + e(c-e)^{i-1}}{d+e}.
\end{aligned}$$

*Computation of the numbers  $\psi_i^{[k]}$ :* We use (24) and the recursion

$$\begin{aligned}
\psi_i^{(1,k)} &= c\psi_{i-1}^{(1,k)} + d\psi_{i-1}^{(0,k)} + \phi_i^{(1,k)}, \\
\psi_i^{(0,k)} &= e\psi_{i-1}^{(1,k)} + g\psi_{i-1}^{(0,k)}
\end{aligned}$$

for  $i \geq 2$  and  $k = 1, 0$ , and

$$\psi_1^{(1,1)} = 1, \quad \psi_1^{(1,0)} = 0, \quad \psi_1^{(0,1)} = 0, \quad \psi_1^{(0,0)} = 0.$$

This inhomogeneous system of difference equations seems to have no closed-form solution, but the values  $\psi_i^{(j,k)}$  can be computed iteratively by means of the formulas for  $\phi_i^{(j,k)}$ .

Computation of the numbers  $\chi_i^{[j]}$ : From

$$\chi_i^{[1]} = c\chi_{i-1}^{[1]} + d\chi_{i-1}^{[0]} + 1, \quad \chi_i^{[0]} = e\chi_{i-1}^{[1]} + g\chi_{i-1}^{[0]}$$

for  $i \geq 2$ , and

$$\chi_1^{[1]} = 1, \quad \chi_1^{[0]} = 0,$$

one obtains, again by using the theory of linear difference equations:

$$\chi_i^{[1]} = \frac{d}{(d+e)^2} [1 - (c-e)^i] + \frac{ei}{d+e},$$

$$\chi_i^{[0]} = \frac{d}{(d+e)^2} + \frac{e}{(d+e)^2} (c-e)^i + \frac{ei-1}{d+e}.$$

In total, the formulas above allow the computation of all passage fitness values and, as a consequence, the numerical solution of (7) for OneMax. Fig. 7 shows the resulting dynamics for  $m = 50$  over a period of 100 time units, starting with a constant initial pheromone value  $\tau_0$  on each arc. To obtain results comparable with Fig. 5, we have chosen  $\tau_0 = 1/2$ , such that the entire pheromone flow out of node  $\underline{0}$  is the same as in the *chain* case.

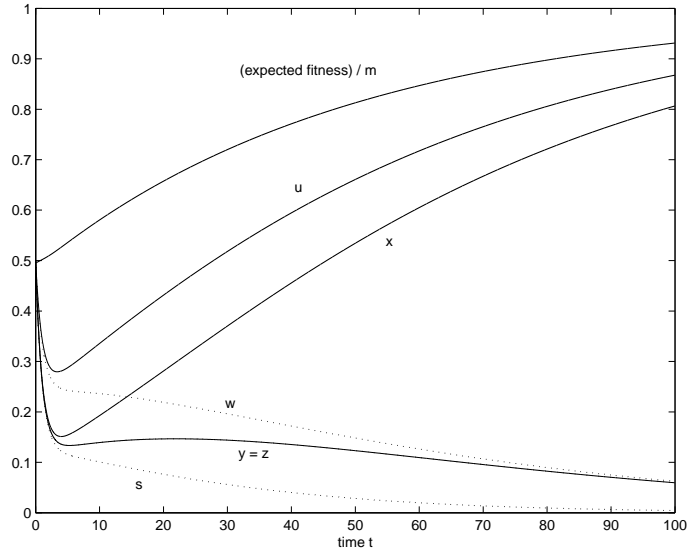


Fig. 7. Dynamics of the drum for OneMax,  $m = 50$  and  $u_0 = w_0 = x_0 = y_0 = z_0 = s_0 = 0.5$ . The current expected fitness is plotted as a relative value, normalized by the maximum  $m = 50$  of the objective function.

Comparison with Fig. 5 leads to some interesting observations:

- The curves for the (relative) expected fitness  $\mu/m$  are nearly identical.
- The curve for  $u$  in the *drum* case is nearly identical to that for  $u/m$  in the *chain* case.
- The curve for  $w$  in the *drum* case is nearly identical to that for  $w/m$  in the *chain* case.

While the corresponding curves almost coincide for  $m = 50$ , decreasing the value of  $m$  leads to growing deviations, which indicates that the congruence is not exact even for larger  $m$ . However, the numerical results suggest the following conjecture, which we are not able to prove formally at the moment:

**Conjecture.** As  $m \rightarrow \infty$ , the expected fitness functions  $\mu_{chain}(t)$  and  $\mu_{drum}(t)$  for OneMax on chain and drum, respectively, with the initial values as given above, are asymptotically equivalent.

To complete our analysis of the three construction graphs investigated in this section, some bibliographical remarks on the origin of them in the ACO literature seem to be in order: Optimization by learning in the *chain* fashion is more typical for *Evolution of Distribution* (EDA) algorithms such as *Population-Based Incremental Learning* (PBIL, see [1]) than for ACO. In the ACO field, *chain*-type construction graphs have been applied rather in their *k*-ary generalization where from each auxiliary node  $\hat{i}$  of the chain, not only two, but  $k$  arcs originate, such that the solution is encoded in a *k*-ary instead of a binary alphabet. Examples of articles where the construction process is organized in this way can be found in [2], [29], [20], [34], or [22]. Construction graphs of *disk* type are the most commonly applied for subset selection problems in the ACO literature, see, e.g., [25], [23], [5], or [15]. It should be noted that our theoretical analysis of the *disk* does not directly address the approaches in these articles, because the problems they investigate have the convenient special property that the considered fitness functions never decrease if a further item is added as long as feasibility is preserved, which provides a “natural” stopping rule for the walks of the ants. In such a context, the main weakness of the *disk*, the inefficiency with respect to the termination condition for the walk, does not occur. A problem where the mentioned monotonicity property is lacking, the *project portfolio selection* problem, has been treated by ACO with the *disk* construction graph in [8], [9]; in these articles, the drawback of the *disk* has been overcome by applying a *lifespan* concept for ants, which, however, requires additional parameters to be tuned. Finally, the *drum* construction graph does not seem to have been applied up to now in the experimental ACO literature.

## 6 Conclusions

We have developed a formal framework for analytical investigations of the finite-time dynamics of ant colony optimization (ACO) under the “classical” pheromone update rule. A limit theorem for the ACO process has been demonstrated, a system of ordinary differential equations has been identified as the “skeleton” of the dynamic process, and it has been shown how the evolution of the expected fitness values can be derived from this system. As an example for the application of the presented theory, the behavior of ACO on three different construction graphs for subset selection problems has been analyzed for some basic test functions.

In particular, it has been shown that the simplest of these construction graphs, the *chain* with storage requirement  $O(m)$ , already turns out as surprisingly efficient for some test functions, but completely fails



on others. The two other investigated types of construction graphs, *disk* and *drum*, use storage of order  $O(m^2)$ . Nevertheless, the disk fails on a very simple test function for which even the chain succeeds. Thus, our theoretical investigation suggests that for diverse subset selection problems, the potential of the *drum* construction graphs might be explored. Of course, the final judgement of the efficiency of an encoding is a matter of empirical research.

Of course, the test functions analyzed here are still rather elementary, and future investigations should turn to more complex types of fitness functions. Let us mention two especially interesting classes: (a) By analyzing ACO on the class of the so-called *Royal Road* fitness functions (see, e.g., [28]), the semi-formal argumentation at the beginning of Subsection 5.1 could possibly be made precise in a quantitative sense. (b) In contrast to “good-natured” functions as OneMax, also *deceptive* fitness functions (see, e.g., [30] or [6]) should be investigated.

As explained at the beginning of Section 4, we have focused on subset selection problems where there is no a-priori order between the items among which a choice has to be made. There are other types of 0-1-problems where such an order exists. Construction graphs that are more suitable for this situation would also deserve to be investigated. Combinatorial problems in sequencing, routing, scheduling etc. need still other construction graphs; the theory presented here includes these cases, but detailed analysis would be necessary to obtain results from which practical conclusions could be drawn.

The pheromone update rule considered here was fitness-proportional. Other rules, such as rank-based update [7] or the update mechanism of MAX-MIN Ant System [37] could be investigated along similar lines. Perhaps this would enable an *analytical* comparison of update rules, which could guide the choice between ACO design variants for new problem types for which experimental know-how is not yet available.

From a theoretical viewpoint, on the other hand, three topics would be of special interest: The first is the question in which sense and under which conditions the trajectories of the real ACO process converge to those of the ACDP. For treating this question, extensive use of the theory of stochastic approximation (see, e.g., Kushner and Yin [24]) could be helpful. A second challenge concerns a refinement of the analysis in Section 2 by finite-population corrections (cf. the related work by Prügel-Benett and Shapiro [32] or Rattray and Shapiro [33] in a genetic algorithm context). Finally, asymptotic approximations of functions describing the process dynamics that can, at the moment, only be computed numerically (cf. the end of Section 5) would be desirable. Mathematical tools as those developed in the analysis of algorithms (see, e.g., Sedgewick and Flajolet [35]) might be applied in this context and could possibly lead to a clearer understanding of the dynamics triggered by particular decisions on problem encoding, design variants and/or parameter choices.

**Acknowledgment.** The author is indebted to Ingo Wegener for suggesting the investigation of “classical” test functions, which led to Subsection 5.2, to Giovanni Sebastiani for a valuable discussion on some aspects of the presented model, and to Marcus Hutter for helpful comments to a previous version of the article.

## References

- [1] Baluja, S., Caruana, R., “Removing the genetics from the standard genetic algorithm”, in: A. Prieditis, S. Russell (eds.), *Proc. 12th Int. Conf. on Machine Learning*, Morgan Kaufmann, San Francisco (1995), pp. 38–46.
- [2] Bauer, A., Bullnheimer, B., Hartl, R.F., Strauss, C., “An ant colony optimization approach for the single machine total tardiness problem”, *Proc. Congress on Evolutionary Computation CEC '99*, vol. 2, Piscataway, NJ (1999), pp. 1445–1450.
- [3] Bianchi, L., Gambardella, L.M., Dorigo, M., “Solving the homogeneous probabilistic travelling salesman problem by the ACO metaheuristic”, *Proc. ANTS '02*, 3rd Int. Workshop on Ant Algorithms (2002), pp. 177–187.
- [4] Birattari, M., Balaprakash, P., Dorigo, M., “ACO/F-Race: Ant colony optimization and racing techniques for combinatorial optimization under uncertainty”, *Proceedings of MIC 2005 – 6th Metaheuristics International Conference* (2005), pp. 107–112.
- [5] Blum, C., Blesa, M., “New metaheuristic approaches for the edge-weighted  $k$ -cardinality tree problem”, *Computers & Operations Research* 32 (2005), pp. 1355 - 1377.
- [6] Blum, C., Dorigo, M., “Search bias in ant colony optimization: On the role of competition-balanced systems”, *IEEE Transactions on Evolutionary Computation* 9 (2005), pp. 159–174.
- [7] Bullnheimer, B., Hartl, R. F., Strauss, C., “A new rank-based version of the Ant System: A computational study”, *Central European Journal for Operations Research and Economics* 7 (1) (1999), pp. 25–38.
- [8] Doerner, K., Gutjahr, W.J., Hartl, R.F., Strauss, C., Stummer, C., “Ant colony optimization in multiobjective portfolio selection”, *Proc. 4th Metaheuristics International Conference*(2001), pp. 243-248 .
- [9] Doerner, K., Gutjahr, W.J., Hartl, R.F., Strauss, C., Stummer, C., “Pareto Ant Colony Optimization: A metaheuristic approach to multiobjective portfolio selection”, *Annals of Operations Research* 131 (2004), pp. 79-99.
- [10] Dorigo, M., Di Caro, G., “The Ant Colony Optimization metaheuristic”, in: *New Ideas in Optimization*, D. Corne, M. Dorigo, F. Glover (eds.), McGraw Hill, London, UK (1999), pp. 11-32.
- [11] Dorigo, M., Di Caro, G., Gambardella, L.M., “Ant algorithms for discrete optimization”, *Artificial Life*, vol. 5, no.2 (1999), pp. 137–172.
- [12] Dorigo, M., Maniezzo, V., Colorni, A., “The Ant System: An autocatalytic optimization process”, Technical Report 91–016, Dept. of Electronics, Politecnico di Milano, Italy (1991).
- [13] Dorigo, M., Maniezzo, V., Colorni, A., “Ant System: Optimization by a colony of cooperating agents”, *IEEE Transactions on Systems, Man, and Cybernetics*, Part B, vol. 26, no. 1 (1996), pp. 29–41.
- [14] Dorigo, M., Stützle, T., *Ant Colony Optimization*, MIT Press, Cambridge, MA (2004).
- [15] Fenet, S., Solnon, C., “Searching for maximum cliques with ant colony optimization”, *Applications of Evolutionary Computing, Proceedings of EvoWorkshops 2003*, Lecture Notes in Computer Science 2611, Springer, Berlin, Germany (2003), pp. 236–245.

- [16] Gutjahr, W.J., “A graph-based Ant System and its convergence”, *Future Generation Computer Systems* 16 (2000), pp. 873–888.
- [17] Gutjahr, W.J., “ACO algorithms with guaranteed convergence to the optimal solution”, *Information Processing Letters* 82 (2002), pp. 145–153.
- [18] Gutjahr, W.J., “A generalized convergence result for the Graph-based Ant System”, *Probability in the Engineering and Informational Sciences* 17 (2003), pp. 545–569.
- [19] Gutjahr, W.J., “A converging ACO algorithm for stochastic combinatorial optimization”, *Proc. SAGA 2003* (Stochastic Algorithms: Foundations and Applications), Eds.: A. Albrecht, K. Steinhoeff, Lecture Notes in Computer Science 2827, Springer, Berlin, Germany (2003), pp. 10 - 25.
- [20] Gutjahr, W.J., “S-ACO: An ant-based approach to combinatorial optimization under uncertainty”, *Ant Colony Optimization and Swarm Intelligence: Proceedings of ANTS 2004 – Fourth International Workshop*, Lecture Notes in Computer Science 3172, Springer, Berlin, Germany (2004), pp. 238–249.
- [21] Gutjahr, W.J., “Convergence rates of ant colony optimization for simple test functions”, in preparation.
- [22] Gutjahr, W.J., Rauner, M., “An ACO algorithm for a dynamic regional nurse-scheduling problem in Austria”, *Computer & Operations Research* (in press).
- [23] Hadji, R., Rahoual, M., Talbi, E., Bachelet, V., “Ant colonies for the set covering problem”, *Abstract Proceedings of ANTS 2000 – From Ant Colonies to Artificial Ants: Second International Workshop on Ant Algorithms*, M. Dorigo, M. Middendorf and T. Stuetzle (eds.), Brussels, Université Libre de Bruxelles (2000), pp. 63–66.
- [24] Kushner, H.J. Yin, G.G., *Stochastic Approximation and Recursive Algorithms and Applications*, Springer, Berlin, Germany (2003).
- [25] Leguizamón, G., Michalewicz, Z., “Ant systems for subset problems”, unpublished manuscript (2000).
- [26] Merkle, D., Middendorf, M., “Modeling the dynamics of ant colony optimization”, *Evolutionary Computation* 10 (2002), pp. 235–262.
- [27] Merkle, D., Middendorf, M., “Competition controlled pheromone update for ant colony optimization”, *Ant Colony Optimization and Swarm Intelligence: Proc. ANTS 2004*, Lecture Notes in Computer Science 3172, Springer, Berlin, Germany (2004), pp. 95–105.
- [28] Mitchell, M., Holland, J.H., Forrest, S., “When will a genetic algorithm outperform hill climbing?”, *Advances in Neural Information Processing Systems* 6 (1994), pp. 51–58.
- [29] Montgomery, J., Randall, M., “Alternative pheromone applications for ant colony optimization”, Bond University, Australia, Technical Report TR02-07 (2002).
- [30] Mühlenbein, H., Mahnig, T., “Evolutionary algorithms: From recombination to search distributions”, in: L. Kallel, B. Naudts and A. Rogers (eds.), *Theoretical Aspects of Evolutionary Computing*, Natural Computing Series, Springer, Berlin, Germany (2001), pp. 134–173.
- [31] Prügel-Bennett, A., Rogers, A., “Modelling genetic algorithm dynamics”, in: L. Kallel, B. Naudts and A. Rogers (eds.), *Theoretical Aspects of Evolutionary Computing*, Natural Computing Series, Springer, Berlin, Germany (2001), pp. 59–85.

- [32] Prügel-Bennett, A., Shapiro, J.L., “The dynamics of a genetic algorithm for simple Ising systems” *Physica D*, 104 (1997), pp. 75–114.
- [33] Rattray, M., Shapiro, J.L., “Noisy fitness evaluations in genetic algorithms and the dynamics of learning”, in: R. K. Belew and M. D. Vose (eds.), *Foundations of Genetic Algorithms 4*, Morgan Kaufmann (1997), pp. 117–139.
- [34] Rauner, M., Brailsford, S.C., Gutjahr, W.J., Zeppelzauer, W., “Optimal screening policies for diabetic retinopathy using a combined discrete-event simulation and ant colony optimization approach”, *Proc. Int. Conference on Health Sciences Simulation, Western MultiConference '05*, Anderson, J.G., Katzper, M. (eds.) (2005), pp. 147–152.
- [35] Sedgewick, R., Flajolet, P., *An Introduction to the Analysis of Algorithms*, AddisonWesley, Reading, Massachusetts (1996).
- [36] Stützle, T., Dorigo, M., “A short convergence proof for a class of ACO algorithms”, *IEEE Transactions on Evolutionary Computation* 6 (2002), pp. 358–365.
- [37] Stützle, T., Hoos, H.H., “MAX-MIN Ant System”, *Future Generation Computer Systems*, 16 (2000), pp. 889–914.