Theory of Swarm Intelligence

Dirk Sudholt

CERCIA, University of Birmingham

Tutorial at GECCO 2011

GECCO'11, July 12-16, 2011, Dublin, Ireland.

Swarm Intelligence

Collective behavior of a "swarm" of agents.

Examples from Nature

- dome construction by termites
- communication of bees
- ant trails
- foraging behavior of fish schools and bird flocks
- swarm robotics

Plenty of inspiration for optimization.

ACO and PSO

Ant colony optimization (ACO)

- inspired by foraging behavior of ants
- artificial ants construct solutions using pheromones
- pheromones indicate attractiveness of solution component

Particle swarm optimization (PSO)

- mimics search of bird flocks and fish schools
- particles "fly" through search space
- each particle is attracted by own best position and best position of neighbors

Theory

What "theory" can mean

- convergence analysis
- analysis of simplified models of algorithms
- empirical studies on test functions
- runtime analysis / computational complexity analysis
- ...

Example Question

How long does it take on average until algorithm A finds a target solution on problem P?

Notion of time: number of iterations, number of function evaluations

Content

What this tutorial is about

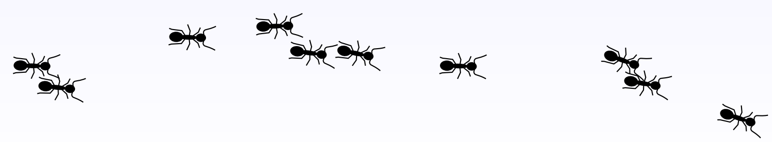
- runtime analysis
- simple variants of swarm intelligence algorithms
- insight into their working principles
- impact of parameters and design choices on performance
- what distinguishes ACO/PSO from evolutionary algorithms?
- performance guarantees for combinatorial optimization
- methods and proof ideas

What this tutorial is not about

- convergence results
- analysis of models of algorithms
- no intend to be exhaustive

Overview

Ant Colony Optimization (ACO)



Main idea: artificial ants communicate via pheromones.

Scheme of ACO

Repeat:

- construct ant solutions guided by pheromones
- update pheromones by reinforcing good solutions

Pseudo-Boolean Optimization

Goal: maximize $f\colon \{0,1\}^n \to \mathbb{R}$.

Often considered in theory of evolutionary algorithms.
Established and well-understood test bed for search heuristics.

Illustrative test functions

$$\textsc{OneMax}(x) = \sum_{i=1}^{n} x_i$$

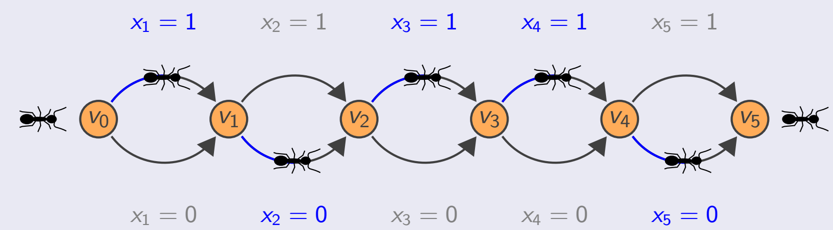
$$\textsc{BinVal}(x) = \sum_{i=1}^{n} 2^{n-i} \cdot x_i$$

$$\textsc{LeadingOnes}(x) = \sum_{i=1}^{n} \prod_{j=1}^{i} x_j$$

$$\textsc{Needle}(x) = \prod_{i=1}^{n} x_i$$

ACO in Pseudo-Boolean Optimization

Solution Construction



Probability of choosing an edge equals pheromone on the edge.

Initial pheromones: $\tau(x_i = 0) = \tau(x_i = 1) = 1/2$.

Note: no linkage between bits.

Pheromones $\tau(x_i = 1)$ suffice to describe all pheromones.

ACO in Pseudo-Boolean Optimization (2)

Pheromone update: reinforce some good solution x.
(x = best-so-far/iteration-best/...)

Strength of update determined by evaporation factor $0 \le \rho \le 1$:

$$\tau'(x_i = 1) = \begin{cases} (1-\rho)\cdot\tau(x_i = 1) & \text{if } x_i = 0 \\ (1-\rho)\cdot\tau(x_i = 1) + \rho & \text{if } x_i = 1 \end{cases}$$

Small ρ: slow adaptation
Large ρ: quick adaptation

Pheromone borders as in MAX-MIN Ant System (Stützle and Hoos, 2000):

$$\tau_{\min} \le \tau' \le 1 - \tau_{\min}$$

Default choice: $\tau_{\min} := 1/n$ (cf. standard mutation in EAs).

Theory of ACO

Analyses performed for:

- illustrative test problems: OneMax, LeadingOnes, ...
- problem classes: unimodal functions, linear functions
- constructed problems
- combinatorial optimization
  - minimum spanning trees
  - TSP
  - shortest path problems
  - stochastic shortest paths
  - minimum cut problem

Focus on simple ACO algorithms

- no heuristic information
- fixed amount of pheromone increase
- one ant in each iteration

One Ant?

Most ACO algorithms analyzed: one ant per iteration.

One ant at a time, many ants over time.

Steady-state GA

- Probabilistic model: Population
- New solutions: selection + variation
- Environmental selection

Ant Colony Optimization

- Probabilistic model: Pheromones
- New solutions: construction graph
- Selection for reinforcement

Evolutionary Algorithms vs. ACO

(1+1) EA

Start with uniform random solution x^* and repeat:

- create x by flipping each bit independently with probability $1/n$
- replace x^* by x if $f(x) \geq f(x^*)$.

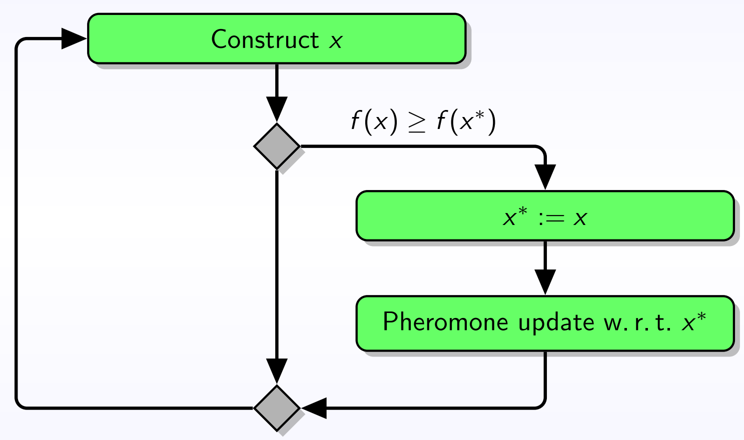
(1+1) EA: Probability of setting bit to 1 is in $\{1/n, 1-1/n\}$.

ACO: Probability of setting bit to 1 is in $[1/n, 1-1/n]$.

Exception: $\rho = 1 \Rightarrow$ ACO = (1+1) EA.
Some ACO algorithms generalize some evolutionary algorithms.

Overview

1-ANT (Neumann and Witt, 2006)

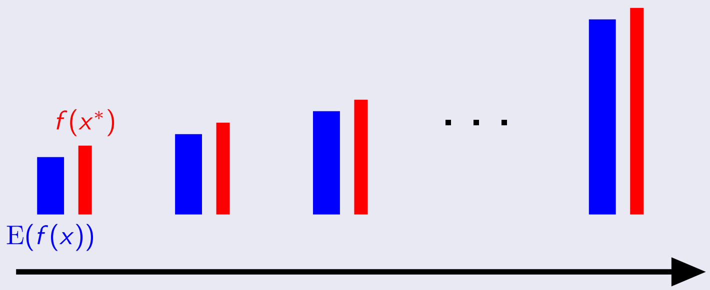


Note: each new x^* is reinforced only once.

1-ANT: Stagnation

Behavior on OneMax (Neumann and Witt, 2006), LeadingOnes and BinVal (Doerr, Neumann, Sudholt, and Witt, 2007):

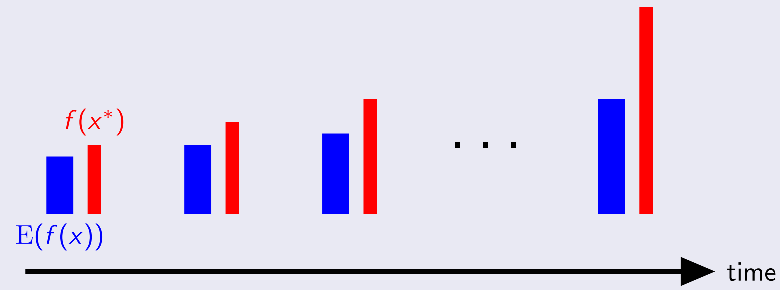
Large ρ (quick adaptation)



Pheromone model follows best solution found so far.

1-ANT: Stagnation

Small ρ (slow adaptation)

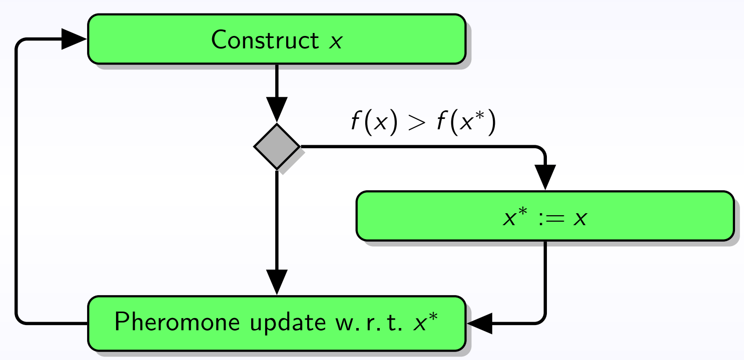


New solutions are not stored in pheromones quickly enough as 1-ANT reinforces each new x^* only once!

Phase transition w. r. t. ρ. Location depends on problem.

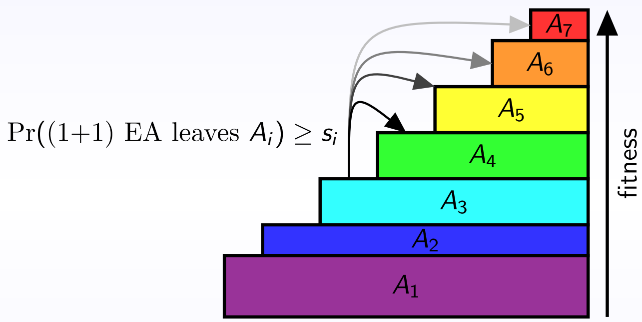
Overview

MMAS* (Gutjahr and Sebastiani, 2008)



Note: best-so-far solution x^* is constantly reinforced.

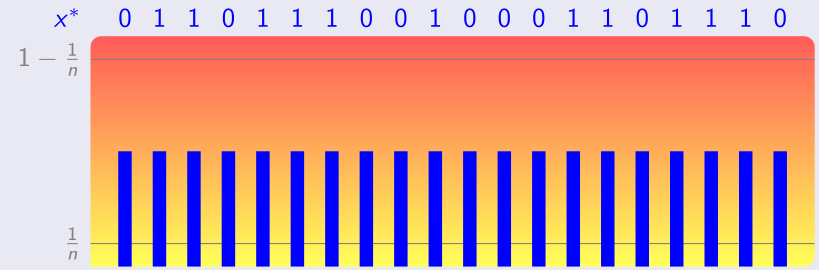
Fitness-level Method for the (1+1) EA



Expected optimization time of (1+1) EA at most $\sum_{i=1}^{m-1} \frac{1}{s_i}$.

MMAS*

Pheromones on 1-edges



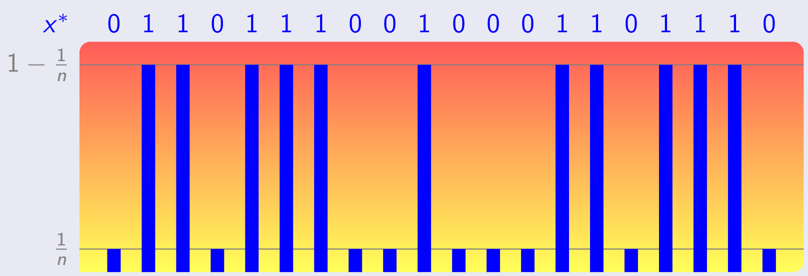
After $(\ln n)/\rho$ reinforcements of x^* MMAS* temporarily behaves like (1+1) EA.

Fitness-Level Method with A_i = search points with i-th fitness value

$$(1+1)\text{ EA}: \quad \sum_{i=1}^{m-1} \frac{1}{s_i} \qquad\qquad \text{MMAS*}: \quad m \cdot \frac{\ln n}{\rho} + \sum_{i=1}^{m-1} \frac{1}{s_i}$$

MMAS*

Pheromones on 1-edges



After $(\ln n)/\rho$ reinforcements of x^* MMAS* temporarily behaves like (1+1) EA.

Fitness-Level Method with A_i = search points with i-th fitness value

$$(1+1)\text{ EA}: \quad \sum_{i=1}^{m-1} \frac{1}{s_i} \qquad\qquad \text{MMAS*}: \quad m \cdot \frac{\ln n}{\rho} + \sum_{i=1}^{m-1} \frac{1}{s_i}$$

Bounds with Fitness Levels

OneMax:

$$s_i \geq (n-i) \cdot \frac{1}{n} \cdot \left(1 - \frac{1}{n}\right)^{n-1} \geq \frac{n-i}{en}$$

Theorem

$$(1+1)\text{ EA}: \quad en \sum_{i=0}^{n-1} \frac{1}{n-i} = O(n \log n)$$

$$\text{MMAS*}: \quad n \cdot \frac{\ln n}{\rho} + en \sum_{i=0}^{m-1} \frac{1}{n-i} = O((n \log n)/\rho)$$

Bounds with Fitness Levels (2)

LeadingOnes

$$s_i \geq \frac{1}{n} \cdot \left(1 - \frac{1}{n}\right)^{n-1} \geq \frac{1}{en}$$

Theorem

(1+1) EA: en^2 MMAS*: $n \cdot \frac{\ln n}{\rho} + en^2 = O(n^2 + (n \log n)/\rho)$

Unimodal functions with d function values:

Theorem

(1+1) EA: end MMAS*: $d \cdot \frac{\ln n}{\rho} + end = O(nd + (d \log n)/\rho)$

Discussion

Q: Does that mean that MMAS* is always worse than the (1+1) EA?

A: No, it only means that we get worse upper bounds!

Remarks

- method relies on MMAS* simulating the (1+1) EA
- neglect effects when pheromones not at their bounds
- real expected running times may differ from upper bounds if many/difficult fitness levels are skipped

Running Times

How to make sense of running times like $O(n^2 + (n \log n)/\rho)$?

$$O(\text{time for improvements}(n) + \text{time for pheromone adaptation}(n, \rho))$$

Time for pheromone adaptation $\widehat{=}$ price for diverse search.

How large is this price for diverse search?

General lower bound (Neumann, Sudholt, and Witt, 2009)

Expected time of MMAS* on any function with unique global optimum is $\Omega((\log n)/\rho)$ if $1/\text{poly}(n) \leq \rho \leq 1/2$.

Conjecture

Can be improved to $\Omega\left(\frac{n}{\rho \log(1/\rho)}\right)$.

Layering of Pheromones

So far: adaptation time of $(\ln n)/\rho$ per fitness level.
Can we argue with smaller adaptation times?

Trade-off in analysis:

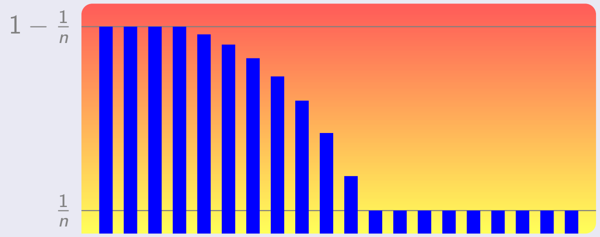
- allow large adaptation time
  $\Rightarrow$ pheromones guaranteed to be well adapted
  $\Rightarrow$ good guarantee to rediscover adapted bit values.
- small adaptation time
  $\Rightarrow$ worse guarantees, pheromones may be not well adapted
  $\Rightarrow$ worse bound for time to rediscover adapted bit values.

Example: improving $O(n^2 + (n \log n)/\rho)$ bound for LeadingOnes.

Layering of Pheromones for LeadingOnes

(Lower bounds on) pheromones on LeadingOnes

best-so-far 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 0 1 0 0 1



Theorem (Neumann, Sudholt, and Witt, 2009)

Bounds for MMAS *and* MMAS* *on* LeadingOnes *of* $O(n^2 + n/\rho)$ *and* $O\left(n^2 \cdot (1/\rho)^\varepsilon + \frac{n/\rho}{\log(1/\rho)}\right)$ *for every constant* $\varepsilon > 0$.

Layering approach also works for BinVal and shortest paths.

Strict Selection

Most ACO algorithms replace x^* only if $f(x) > f(x^*)$.
Danger: algorithm gets stuck on first point of a plateau.

MMAS* on Needle: first solution is 0^n with probability 2^{-n}.
After pheromone freezing, the probability of finding the needle is n^{-n}.

Theorem (Neumann, Sudholt, Witt, 2009)

If $\rho \geq 1/poly(n)$ *the expected optimization time of* MMAS* *on* Needle *is* $\Omega(2^{-n} \cdot n^n) = \Omega((n/2)^n)$.

MMAS on Needle

Define variant MMAS of MMAS* replacing x^* if $f(x) \geq f(x^*)$.

MMAS: pheromones on each bit perform a random walk.

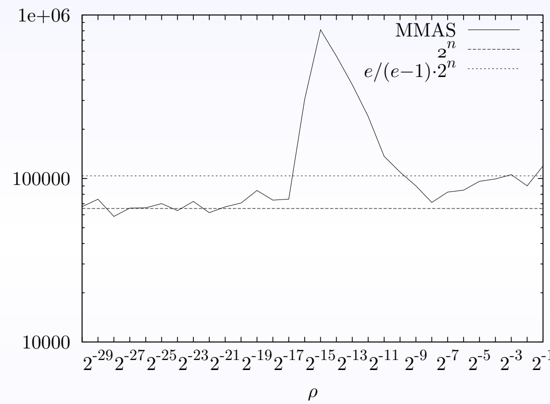
Theorem (Neumann, Sudholt, Witt, 2009 and Sudholt, 2011)

The expected time of MMAS *on* Needle *is* $O(n^2/\rho^2 \log n \cdot 2^n)$.

Proof ideas using tools from Markov Chain Monte Carlo (Sudholt, 2011):

- Consider random walk of MMAS on the constant function.
- Stationary distribution: uniform solution construction.
- After mixing time $O(n^2/\rho^2 \log n)$ MMAS is close to stationarity.
- After every period of $O(n^2/\rho^2 \log n)$ iterations the needle is found with probability $\Omega(2^{-n})$.

MMAS on Needle: Experiments, $n = 16$



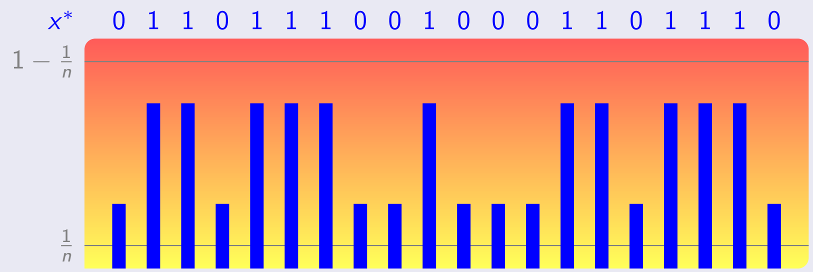
$\rho = 1$: MMAS = (1+1) EA.
ρ very small: MMAS $\approx$ random search.
Intermediate ρ: MMAS tends to resample.

MMAS on unimodal functions

MMAS is better than MMAS* on plateaus.
Does MMAS perform worse on unimodal problems?

Switching between equally fit solutions can prevent freezing.

Pheromones on 1-edges



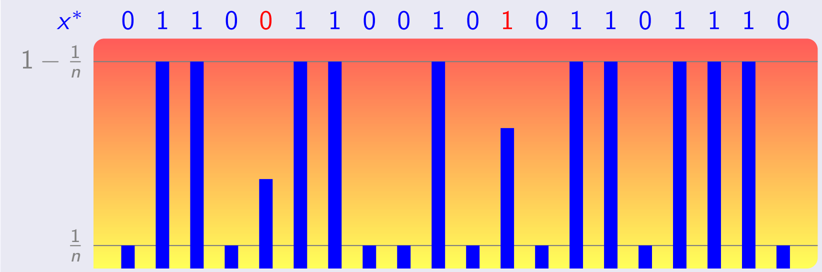
Fitness-level method breaks down!

MMAS on unimodal functions

MMAS is better than MMAS* on plateaus.
Does MMAS perform worse on unimodal problems?

Switching between equally fit solutions can prevent freezing.

Pheromones on 1-edges



Fitness-level method breaks down!

MMAS on unimodal functions

Theorem

The expected optimization time of MMAS *on any unimodal function with d values is $O((dn^2 \log n)/\rho)$.*
(Recall for MMAS**: $O(nd + (d \log n)/\rho)$.)*

- After $(\ln n)/\rho$ steps a solution x with $f(x) \geq f(x^*)$ has been found with good probability.
- Conditioning on $f(x) \geq f(x^*)$, the probability that $f(x) > f(x^*)$ is $\Omega(1/n^2)$.
  - Every non-optimal search point y has a better Hamming neighbor z.
  - Prob(construct z) $\geq 1/n \cdot$ Prob(construct y).
  - A better Hamming neighbor z can be "shared" by up to n search points $y_1, \ldots, y_n$.
- Fitness improvement after expected time $O((n^2 \cdot \log n)/\rho)$.
- Optimum found after d improvements.

MMAS for linear functions

Same idea, with a clever fitness-level partition due to Wegener (2001):

Theorem (Kötzing, Neumann, Sudholt, Wagner, 2011)

The expected optimization time of MMAS* *and* MMAS *on any linear function $f(x) = w_0 + \sum_{i=1}^{n} w_i x_i$ with positive weights is $O((n^3 \log n)/\rho)$.*

Good news

MMAS* and MMAS have polynomial expected optimization time on linear functions and unimodal functions with $d = \text{poly}(n)$ values, if $\rho \geq 1/\text{poly}(n)$.

Bad news

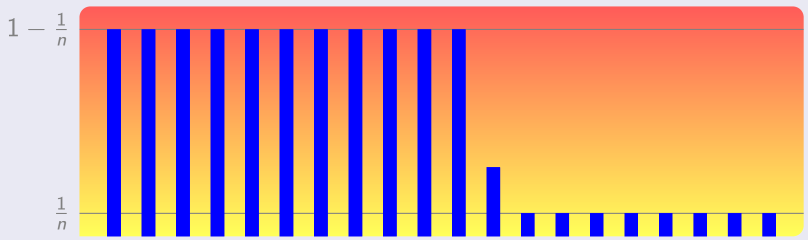
Loose bounds for many functions, including OneMax:
MMAS*: $O((n \log n)/\rho)$ and MMAS: $O((n^3 \log n)/\rho)$.

Pheromone Distributions

Assuming the sum of pheromones is fixed, what is the worst possible distribution?

Solution for OneMax due to Gleser, 1975:

Pheromones on 1-edges

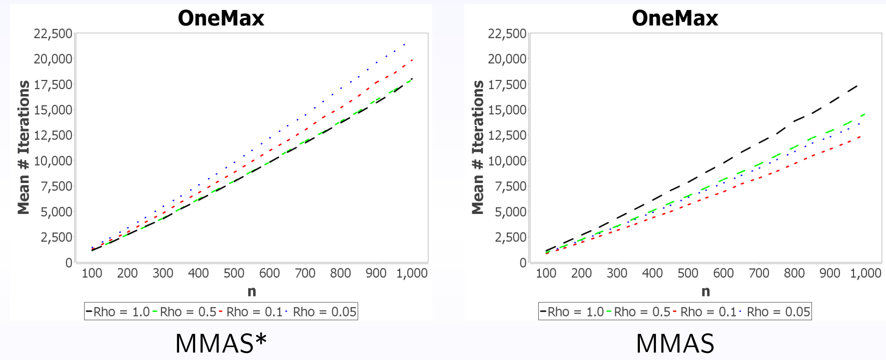


Worst case: all pheromones (but one) at borders.

Theorem (Kötzing, Neumann, Sudholt, and Wagner, 2011)

$O(n \log n + n/\rho)$ *on* OneMax *for both* MMAS* *and* MMAS.

Experiments (Kötzing et al., 2011)

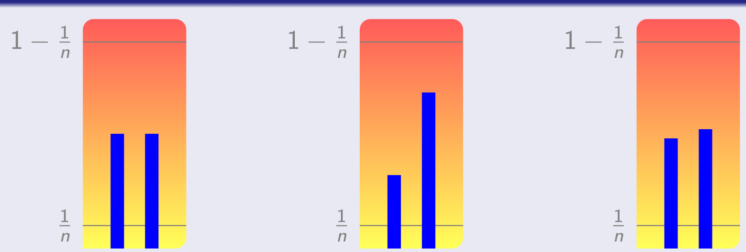


- MMAS better than MMAS*
- MMAS with $\rho = 0.1$ better than (1+1) EA (=MMAS at $\rho = 1$)!
- does not hold for MMAS*

Explanation

Possible explanation: it helps to reward different bits.

Example for two bits and $\rho = 0.2$



Prob(both 1) $= 0.25$ | Prob(both 1) ≈ 0.22 | Prob(both 1) ≈ 0.25

Proper ρ: MMAS remembers past 1-bits.

Open Problem

Prove that MMAS with proper ρ is faster than MMAS* and (1+1) EA.

Overview

ACO with Local Search

Scheme of ACO

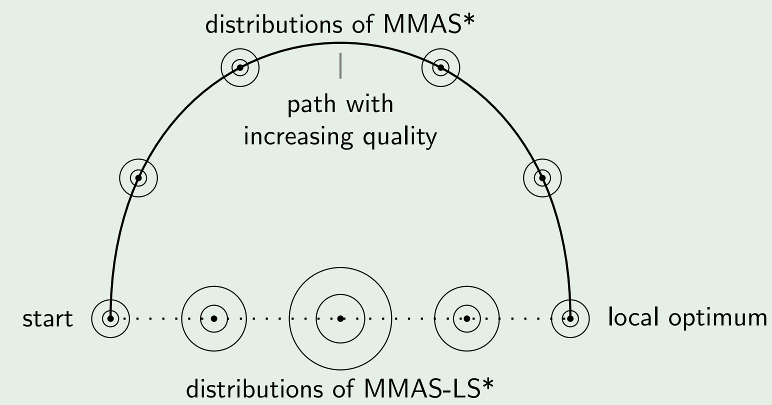
Repeat:

- construct ant solutions guided by pheromones
- local search
- update pheromones by reinforcing good solutions

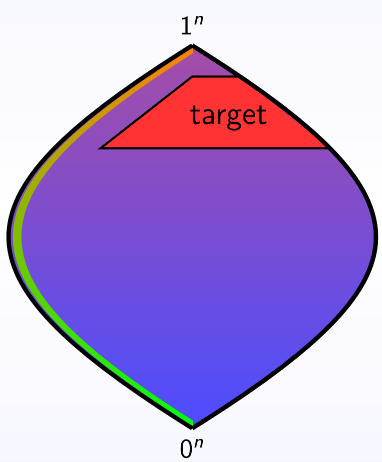
How does the addition of local search affect search dynamics?

ACO with Local Search (2)

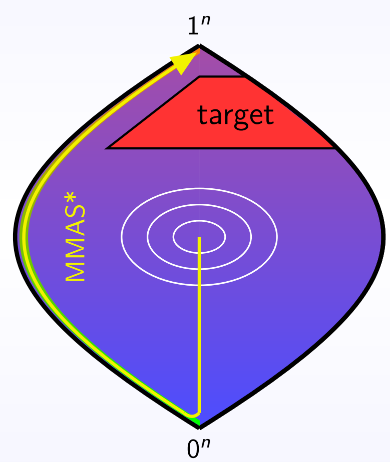
Neumann, Sudholt, Witt, 2008



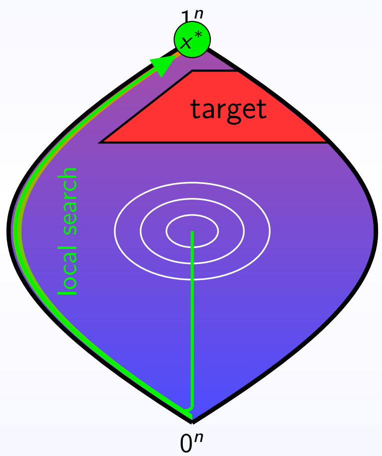
Exponential Performance Gaps



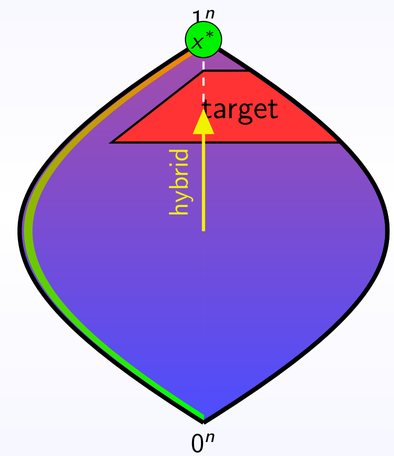
Exponential Performance Gaps



Exponential Performance Gaps



Exponential Performance Gaps



Overview

Iteration-Best Update

λ-MMAS$_{\text{ib}}$

Repeat:

- construct λ ant solutions
- update pheromones by reinforcing the best of these solutions

Advantages:

- can escape from local optima
- inherently parallel
- simpler ants

Iteration-Best vs. Comma Strategies

Jägerskùpper and Storch, 2007

(1,λ) EA: $\lambda \geq c \log n$ necessary, even for OneMax.

If $\lambda \leq c' \log n$ then (1,λ) EA needs exponential time.

Reason: (1,λ) EA moves away from optimum if close and λ too small.

Behavior too chaotic to allow for hill climbing!

Iteration-Best on OneMax

Slow pheromone adaptation effectively eliminates chaotic behavior.

Theorem

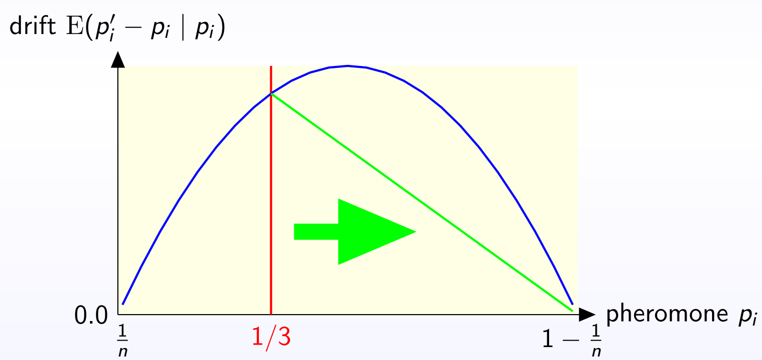
If $\rho \leq 1/(cn^{1/2} \log n)$ for a sufficiently large constant $c > 0$ and $\rho \geq 1/poly(n)$ then 2-MMAS$_{ib}$ optimizes OneMax in expected time $O(\sqrt{n}/\rho)$.
For $\rho = 1/(cn^{1/2} \log n)$ the time bound is $O(n \log n)$.

Two ants are enough!

Proof Ideas

"Local" drift for pheromone on each bit i:

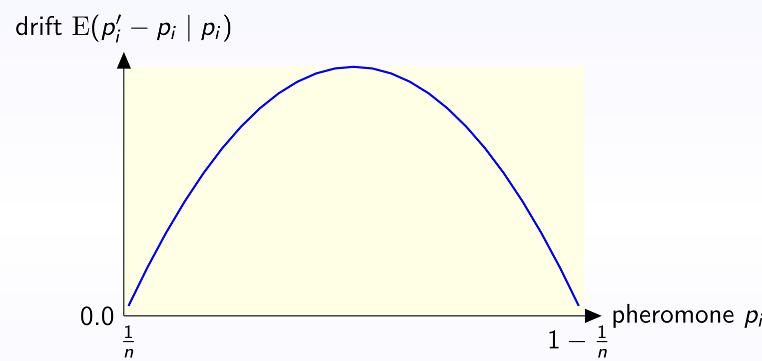
$$\mathrm{E}(p_i' - p_i \mid p_i) \;\geq\; \rho \cdot p_i(1-p_i) \cdot \frac{1}{11}\left(\sum_{j \neq i} p_j(1-p_j)\right)^{-1/2}.$$



"Local" drift implies "global" drift for sum of pheromones.

Lower Bound

λ/ρ small $\Rightarrow$ chance of "Landslide sequence": pheromones go to $1/n$.

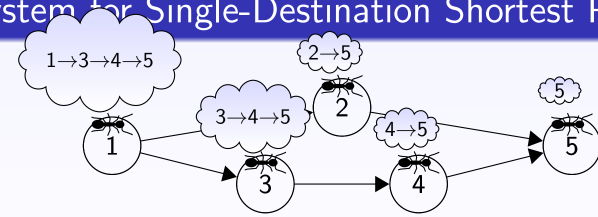


Theorem

Choosing $\lambda/\rho \leq (\ln n)/244$, the expected optimization time of λ-MMAS$_{ib}$ on a function with unique optimum is $2^{\Omega(n^{\varepsilon})}$ for some constant $\varepsilon > 0$ with overwhelming probability.

Overview

ACO System for Single-Destination Shortest Path Problem

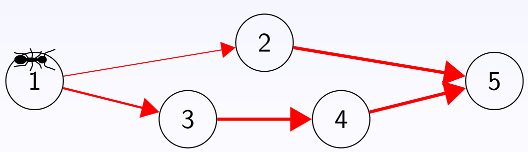


Let $w(p) = \begin{cases} \sum_{e \in p} w(e) & \text{if } p \text{ ends in } n \\ \infty & \text{otherwise.} \end{cases}$

Ant System for Single-Destination Shortest Path Problem

- initialize pheromones τ and best-so-far paths $p_1^*, \dots, p_n^*$
- **for** $u = 1$ to n **do in parallel**
  - let ant $x^{(u)}$ construct a simple path p_u from u to n w. r. t. τ
  - **if** $w(p_u) \le w(p_u^*)$ **then** $p_u^* \leftarrow p_u$
  - update pheromones on edges $(u, \cdot)$ w. r. t. p_u^*
- repeat

ACO System for Single-Destination Shortest Path Problem

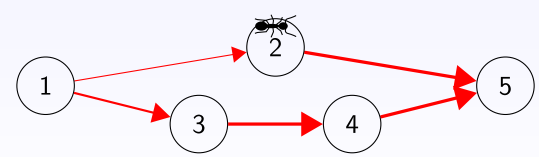


Let $w(p) = \begin{cases} \sum_{e \in p} w(e) & \text{if } p \text{ ends in } n \\ \infty & \text{otherwise.} \end{cases}$

Ant System for Single-Destination Shortest Path Problem

- initialize pheromones τ and best-so-far paths $p_1^*, \dots, p_n^*$
- **for** $u = 1$ to n **do in parallel**
  - let ant $x^{(u)}$ construct a simple path p_u from u to n w. r. t. τ
  - **if** $w(p_u) \le w(p_u^*)$ **then** $p_u^* \leftarrow p_u$
  - update pheromones on edges $(u, \cdot)$ w. r. t. p_u^*
- repeat

ACO System for Single-Destination Shortest Path Problem

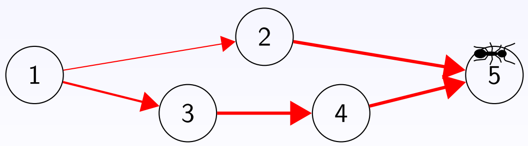


Let $w(p) = \begin{cases} \sum_{e \in p} w(e) & \text{if } p \text{ ends in } n \\ \infty & \text{otherwise.} \end{cases}$

Ant System for Single-Destination Shortest Path Problem

- initialize pheromones τ and best-so-far paths $p_1^*, \dots, p_n^*$
- **for** $u = 1$ to n **do in parallel**
  - let ant $x^{(u)}$ construct a simple path p_u from u to n w. r. t. τ
  - **if** $w(p_u) \le w(p_u^*)$ **then** $p_u^* \leftarrow p_u$
  - update pheromones on edges $(u, \cdot)$ w. r. t. p_u^*
- repeat

ACO System for Single-Destination Shortest Path Problem



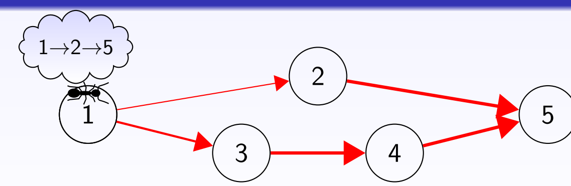
Let $w(p) = \begin{cases} \sum_{e \in p} w(e) & \text{if } p \text{ ends in } n \\ \infty & \text{otherwise.} \end{cases}$

Ant System for Single-Destination Shortest Path Problem

- initialize pheromones τ and best-so-far paths $p_1^*, \dots, p_n^*$
- **for** $u = 1$ to n **do in parallel**
  - let ant $x^{(u)}$ construct a simple path p_u from u to n w. r. t. τ
  - **if** $w(p_u) \le w(p_u^*)$ **then** $p_u^* \leftarrow p_u$
  - update pheromones on edges $(u, \cdot)$ w. r. t. p_u^*
- repeat

Shortest Paths Single-Destination Shortest Paths

ACO System for Single-Destination Shortest Path Problem



Let $w(p) = \begin{cases} \sum_{e \in p} w(e) & \text{if } p \text{ ends in } n \\ \infty & \text{otherwise.} \end{cases}$

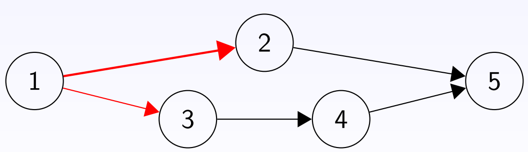
Ant System for Single-Destination Shortest Path Problem

- initialize pheromones τ and best-so-far paths $p_1^*, \dots, p_n^*$
- **for** $u = 1$ to n **do in parallel**
  - let ant $x^{(u)}$ construct a simple path p_u from u to n w. r. t. τ
  - **if** $w(p_u) \le w(p_u^*)$ **then** $p_u^* \leftarrow p_u$
  - update pheromones on edges $(u, \cdot)$ w. r. t. p_u^*
- repeat

Dirk Sudholt (University of Birmingham) Theory of Swarm Intelligence 49 / 107

Shortest Paths Single-Destination Shortest Paths

ACO System for Single-Destination Shortest Path Problem



Let $w(p) = \begin{cases} \sum_{e \in p} w(e) & \text{if } p \text{ ends in } n \\ \infty & \text{otherwise.} \end{cases}$

Ant System for Single-Destination Shortest Path Problem

- initialize pheromones τ and best-so-far paths $p_1^*, \dots, p_n^*$
- **for** $u = 1$ to n **do in parallel**
  - let ant $x^{(u)}$ construct a simple path p_u from u to n w. r. t. τ
  - **if** $w(p_u) \le w(p_u^*)$ **then** $p_u^* \leftarrow p_u$
  - update pheromones on edges $(u, \cdot)$ w. r. t. p_u^*
- repeat

Dirk Sudholt (University of Birmingham) Theory of Swarm Intelligence 49 / 107

Shortest Paths Single-Destination Shortest Paths

Details of Pheromone Update

Initialization

- pheromones $\tau((u, v)) = 1/\deg(u)$ for all $(u, v) \in E$
- and best-so-far paths $p_u^* = ()$ for all $u \in V$

Pheromone Update

Update $\tau\colon E \to \mathbb{R}_0^+$ according to:

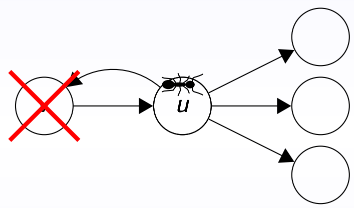
$$\tau(e = (u, v)) \leftarrow \begin{cases} \min\{(1-\rho) \cdot \tau(e) + \rho, \tau_{\max}\} & e \in p_u^* \\ \max\{(1-\rho) \cdot \tau(e), \tau_{\min}\} & e \notin p_u^* \end{cases}$$

where $0 < \rho < 1$ evaporation rate and $0 \le \tau_{\min} \le \tau_{\max}$ bounds for pheromones

Assume $\tau_{\min} + \tau_{\max} = 1$, $\tau_{\min} \le 1/\Delta$, and $\tau_{\min}, \rho \ge 1/\text{poly}(n)$.

Lemma

$$1 \le \sum_{e=(u,\cdot)\in E} \tau(e) \le 1 + \deg(u) \cdot \tau_{\min} \le 2.$$



Corollary

For every edge $e = (u, v)$

$$\frac{1}{2} \cdot \tau(e) \le \mathrm{Prob}\Big(\text{ant } x^{(u)} \text{ chooses edge } e\Big) \le \tau(e).$$

First Upper Bound

Define

- $\Delta := \Delta(G)$: maximum out-degree of any vertex
- $\ell := \ell(G)$: maximum number of edges on any shortest path

Theorem

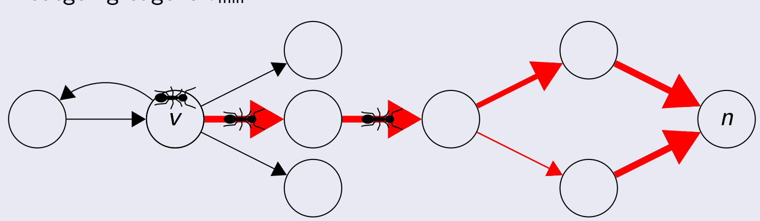
Consider a directed graph G with positive weights.
If $\tau_{\min} \le 1/(\Delta\ell)$, the expected number of iterations is

- $O(n/\tau_{\min} + n\log(1/\tau_{\min})/\rho)$, which for $\tau_{\min} = 1/(\Delta\ell)$ simplifies to
- $O(n\Delta\ell + n\log(\Delta\ell)/\rho)$.

Main proof idea: shortest paths propagate through the graph.

Proof (following Attiratanasunthron and Fakcharoenphol)

- some notions:
  - edge e is correct if it belongs to a shortest path to n
  - vertex u is optimized if $x^{(u)}$ has found a shortest path from u to n
  - vertex u is processed if u is optimized and the pheromone on every incorrect outgoing edge is $\tau_{\min}$



$\tau(e)/2 \ge \tau_{\min}/2$ $\quad$ $(1 - \Delta\tau_{\min})^{\ell-1} \ge \left(1 - \frac{1}{\ell}\right)^{\ell-1} \ge 1/e$

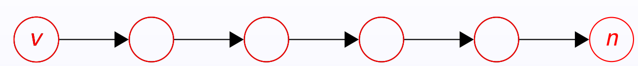
- expected time until v is optimized at most $2e/\tau_{\min}$.
- v becomes processed after further $\ln(\tau_{\max}/\tau_{\min})/\rho$ iterations.
- consider vertices ordered w. r. t. increasing shortest path distance: $n \cdot ((2e/\tau_{\min}) + \ln(\tau_{\max}/\tau_{\min})/\rho) = O(n/\tau_{\min} + n\log(\tau_{\min}/\tau_{\max})/\rho)$

Theorem

Let $\ell^* := \max\{\ell, \ln n\}$. Consider a directed graph G with positive weights where all shortest paths are unique. If $\tau_{\min} \le 1/(\Delta\ell)$, the expected number of iterations is w. h. p. (i. e. $1 - n^{-c}$ for some constant $c > 0$)

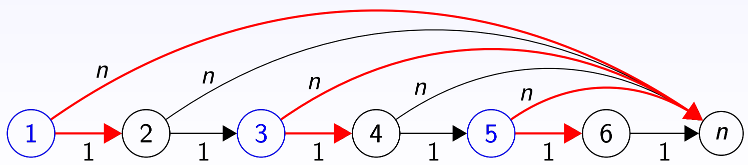
- $O(\ell^*/\tau_{\min} + \ell/\rho)$, which for $\tau_{\min} = 1/(\Delta\ell)$ simplifies to
- $O(\Delta\ell\ell^* + \ell/\rho)$.

Main idea: number of iterations for path with $\Omega(\log n)$ edges is sharply concentrated around its expectation [Doerr et. al, CEC 2007]



$\Rightarrow$ independent coin tosses with success probability $\tau_{\min}/(4e)$.

Is the Upper Bound Tight?



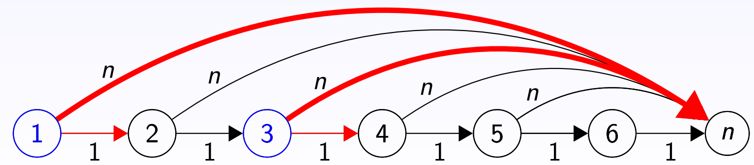
Expected time $O(\ell/\tau_{\min} + \ell/\rho)$ and $\Omega\left(\ell/\tau_{\min} + \frac{\ell}{\rho \log(1/\rho)}\right)$

- #wrong vertices decreases on average by $O(\rho \log(1/\rho))$.
- expected time for decrease of $\Omega(\ell) \Rightarrow \Omega\left(\frac{\ell}{\rho \log(1/\rho)}\right)$.

After pheromone adaptation still $\Omega(\ell)$ wrong vertices left

- #wrong vertices decreases on average by $O(\tau_{\min})$
- expected time for decrease of $\Omega(\ell) \Rightarrow \Omega\left(\frac{\ell}{\tau_{\min}}\right)$.

Is the Upper Bound Tight?



Expected time $O(\ell/\tau_{\min} + \ell/\rho)$ and $\Omega\left(\ell/\tau_{\min} + \frac{\ell}{\rho \log(1/\rho)}\right)$

- #wrong vertices decreases on average by $O(\rho \log(1/\rho))$.
- expected time for decrease of $\Omega(\ell) \Rightarrow \Omega\left(\frac{\ell}{\rho \log(1/\rho)}\right)$.

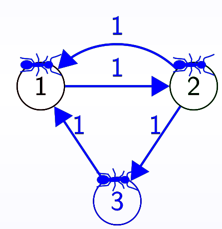
After pheromone adaptation still $\Omega(\ell)$ wrong vertices left

- #wrong vertices decreases on average by $O(\tau_{\min})$
- expected time for decrease of $\Omega(\ell) \Rightarrow \Omega\left(\frac{\ell}{\tau_{\min}}\right)$.

Overview

All-Pairs Shortest Path Problem

Use distinct pheromone function $\tau_v \colon E \to \mathbb{R}_0^+$ for each destination v:



A Simple Interaction Mechanism

Path construction with interaction

For each ant $x^{(u,v)}$

- with prob. 1/2
  - use τ_v to travel from u to v
- with prob. 1/2
  - choose an intermediate destination $w \in V$ uniformly at random
  - uses τ_w to travel from u to w
  - uses τ_v to travel from w to v

Speed-up by Interaction

Theorem

If $\tau_{\min} = 1/(\Delta\ell)$ and $\rho \leq 1/(23\Delta \log n)$ the number of iterations using interaction w. h. p. is $O(n \log n + \log(\ell) \log(\Delta\ell)/\rho)$.

Possible improvement: $O(n^3) \to O(n \log^3 n)$
(with proper ρ and $\Delta, \ell = \Omega(n)$)

Number of function evaluations better than GA by Doerr, Happ, and Klein (2008) but slightly worse than more tailored GA by Doerr, Johannsen, Kötzing, Neumann, and Theile (2010).

Sketch of Proof

- $\rho \leq 1/(23\Delta \log n)$
  → within $\Theta(1/\rho) = \Omega(\Delta \log n)$ iterations almost uniform search
  → all shortest paths with 1 edge found with high probability
- Divide run into phases $1, \dots, \alpha := \lceil \log_{3/2} \ell \rceil$
- Phase i ends when all shortest paths with $\leq (3/2)^i$ edges processed
- after Phase i the probability of finding a shortest path with $(3/2)^i < \ell \leq (3/2)^{i+1}$ edges between fixed vertices at least $\frac{(3/2)^i}{6en}$:
  - $1/2$: ant decides to choose intermediate destination
  - $(\ell/3)/n$: intermediate destination on middle third of shortest path
  - $1/e$: ant follows shortest paths
- w. h. p. Phase $i+1$ takes at most $\frac{6en}{(3/2)^i} \ln(2\alpha n^3)$ iterations.
- expected #iterations (including time for pheromone adaptation):
  $\sum_{i=1}^{\alpha} \left(\frac{6en \ln(2\alpha n^3)}{(3/2)^i} + \frac{\ln(\Delta\ell)}{\rho} \right) = O(n \log n) \cdot \sum_{i=1}^{\alpha} \frac{1}{(3/2)^i} + \frac{\alpha \ln(\Delta\ell)}{\rho}$

Note: slow adaptation helps!

Overview

Stochastic Shortest Paths

Directed acyclic graph $G = (V, E, w)$ with non-negative weights
Family $(\eta(e))_{e \in E}$ of nonnegative random variables

Noise on edge e: $\eta(e) \cdot w(e)$.

For a path $p = (e_1, \ldots, e_\ell)$
- $w(p) := \sum_{i=1}^{\ell} w(e_i)$ is the real length of p.
- $\tilde{w}(p) := \sum_{i=1}^{\ell} (1 + \eta(e_i)) \cdot w(e_i)$ is the noisy length of p.

Goal

Find or approximate real shortest paths despite noise.
α-approximation: all real paths lengths within α of optimum.

Remarks

As η is nonnegative, $w(p) \leq \tilde{w}(p)$.
Noise is independent throughout iterations.
No re-evaluation of stored best-so-far paths.

Results for Arbitrary Noise

Maximum noise $\eta_{\max} := \max_{e \in E} \mathrm{E}(\eta(e))$
Maximum weighted noise $\tilde{w}_{\max} := \max_{e \in E} \mathrm{E}(\eta(e)) \cdot w(e)$

General bounds for arbitrary noise (Horoba and Sudholt, 2010, extended)

In expected time $O((\ell \log n)/\tau_{\min} + \ell(\log n)/\rho)$ MMAS$_{\text{SDSP}}$ finds

multiplicative error:	a $(1 + c \cdot \eta_{\max})^{\ell}$-approximation ($c > 1$ constant),
additive error:	a solution with additive error $O(\ell^2 \cdot \tilde{w}_{\max})$, and
global optimum:	a 1-approximation if every non-optimal path from each vertex v has real length at least $(1 + c \cdot \mathrm{E}(\eta(\text{opt}_v))) \cdot \text{opt}_v$.

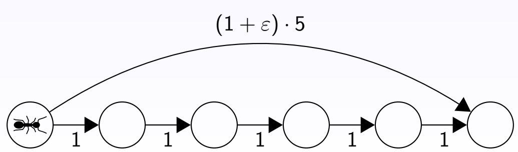
Example where additive error is $\Omega(\ell \cdot \tilde{w}_{\max})$ is necessary.

Open problem

Additive error: close the gap between $O(\ell^2 \cdot \tilde{w}_{\max})$ and $\Omega(\ell \cdot \tilde{w}_{\max})$.

Ants Become Risk-Seeking

Every edge has independent noise $\sim \Gamma(k, \theta)$.



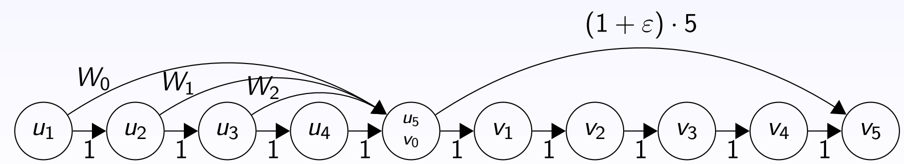
Ant tends to store path with high variance as best-so-far path.

Lemma

With probability $1 - \exp(-\Omega(\sqrt{n}))$ after $n/(6\tau_{\min}) + \sqrt{n}\ln(1/\tau_{\min})/\rho$ iterations

1. *the ant's best-so-far path starts with the upper edge,*
2. *the pheromone on the first lower edge is $\tau_{\min}$, and*
3. *probability of changing best-so-far path is $\exp(-\Omega(n))$.*

Lower Bound for Independent Noise



With probability $1 - \exp(-\Omega(n/\log n))$ MMAS$_{\text{SDSP}}$ does not find a 2-approximation on the left part in time $n/(6\tau_{\min}) + \sqrt{n}\ln(1/\tau_{\min})/\rho$.

Theorem

Let $k = o(\log n)$, $k\theta \leq d$ for some constant $d > e$, and $1/\text{poly}(n) \leq \tau_{\min}, \rho \leq 1/2$. There is a graph where with probability $1 - \exp(-\Omega(\sqrt{n}/\log n))$ MMAS$_{\text{SDSP}}$ does not achieve an approximation ratio better than $(1 + k\theta/d)$ within the first e^{cn} iterations, $c > 0$ a small constant.

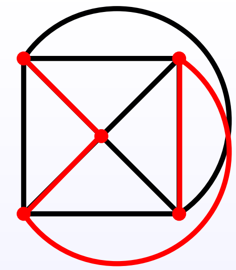
Overview

Broder's Algorithm

Problem: Minimum Spanning Trees

Consider the input graph itself as construction graph.

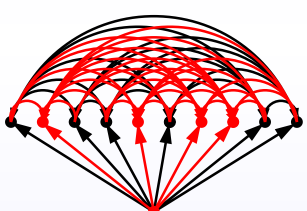
Spanning tree can be chosen uniformly at random using random walk algorithms (e. g. Broder, 1989).



Reward chosen edges ⇒ next solution will be similar to constructed one
But: local improvements are possible

Component-based Construction Graph

- Vertices correspond to edges of the input graph
- Construction graph $C(G) = (N, A)$ satisfies $N = \{0, \dots, m\}$ (start vertex 0) and $A = \{(i, j) \mid 0 \le i \le m, 1 \le j \le m, i \ne j\}$.



For a given path $v_1, \dots, v_k$ select the next edge from its neighborhood $N(v_1, \dots, v_k) := (E \setminus \{v_1, \dots, v_k\}) \setminus \{e \in E \mid (V, \{v_1, \dots, v_k, e\})$ contains a cycle$\}$ (problem-specific aspect of ACO). Reward: all edges, that point to visited vertices (neglect order of chosen edges)

Algorithm

1-ANT: (following Neumann/Witt, 2010)

- two pheromone values
- value h: if edge has been rewarded
- value ℓ: otherwise
- heuristic information η, $\eta(e) = \frac{1}{w(e)}$ (used before for TSP)
- Let v_k the current vertex and N_{v_k} be its neighborhood.
- Prob(to choose neighbor y of v_k) $= \frac{[\tau_{(v_k,y)}]^\alpha \cdot [\eta_{(v_k,y)}]^\beta}{\sum_{y \in N(v_k)} [\tau_{(v_k,y)}]^\alpha \cdot [\eta_{(v_k,y)}]^\beta}$ with $\alpha, \beta \ge 0$.
- Consider special cases where either $\beta = 0$ or $\alpha = 0$.

Results for Pheromone Updates

Case $\alpha = 1$, $\beta = 0$: proportional influence of pheromone values

Theorem (Broder-based construction graph)

Choosing $h/\ell = n^3$, the expected time until the 1-ANT with the Broder-based construction graph has found an MST is $O(n^6(\log n + \log w_{max}))$.

Theorem (Component-based construction graph)

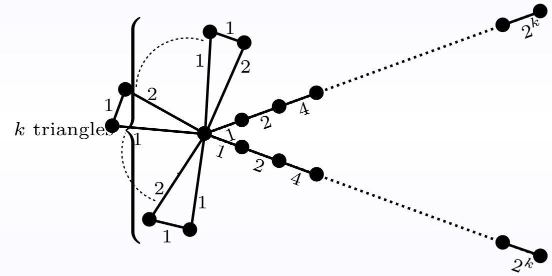
Choosing $h/\ell = (m - n + 1)\log n$, the expected time until the 1-ANT with the component-based construction graph has found an MST is $O(mn(\log n + \log w_{max}))$.

Better than (1+1) EA!

Broder Construction Graph: Heuristic Information

Example graph G^* with $n = 4k + 1$ vertices.

- k triangles of weight profile $(1, 1, 2)$
- two paths of length k with exponentially increasing weights.



Theorem (Broder-based construction graph)

Let $\alpha = 0$ and β be arbitrary, then the probability that the 1-ANT using the Broder construction procedure does not find an MST in polynomial time with probability $1 - 2^{-\Omega(n)}$.

Component-based Construction Graph/Heuristic Information

Theorem (Component-based construction graph)

Choosing $\alpha = 0$ and $\beta \geq 6w_{\max}\log n$, the expected time of the 1-ANT with the component-based construction graph to find an MST is constant.

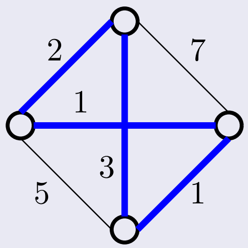
Proof Idea

- Choose edges as Kruskal's algorithm.
- Calculation shows: probability of choosing a lightest edge is at least $1 - 1/n$.
- $n - 1$ steps $\Longrightarrow$ probability for an MST is $\Omega(1)$.

Overview

Traveling Salesman Problem

Traveling Salesman Problem (TSP)



- Input: weighted complete graph $G = (V, E, w)$ with $w : E \to \mathbb{R}$.
- Goal: Find Hamiltonian cycle of minimum weight.

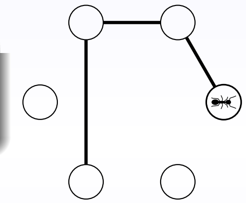
MMAS for the TSP

Best-so-far pheromone update with $\tau_{\min} := 1/n^2$ and $\tau_{\max} := 1 - 1/n$.

Initialization: same pheromone on all edges.

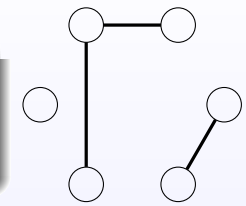
"Ordered" tour construction

Append a feasible edge chosen with probability proportional to pheromones.



"Arbitrary" tour construction

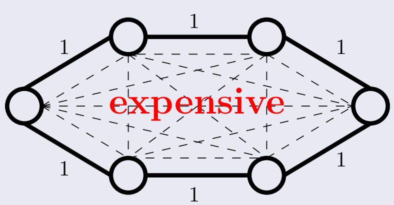
Add an edge chosen with probability proportional to pheromones as long as no cycle is closed or a vertex get degree at least 3.



Previous Work

Theorem [Yuren Zhou 2009]

MMAS* needs $O(n^6)$ iterations in expectation to find optimal solution on the following example:

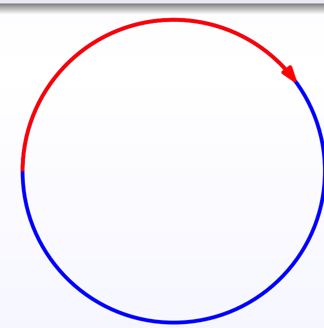


Missing Locality

Pheromones saturated:
$\tau(e) = \tau_{\max}$ for $e \in x^*$ $\quad$ $\tau(e) = \tau_{\min}$ for $e \notin x^*$

Lemma

MMAS* with saturated pheromones exchanges $\Omega(\log(n))$ edges in expectation.



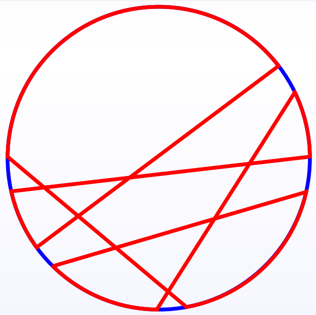
Length of unseen part roughly halves each time.

Missing Locality

Pheromones saturated:
$\tau(e) = \tau_{\max}$ for $e \in x^*$ $\qquad$ $\tau(e) = \tau_{\min}$ for $e \notin x^*$

Lemma

MMAS* with saturated pheromones exchanges $\Omega(\log(n))$ edges in expectation.



Length of unseen part roughly halves each time.

Locality

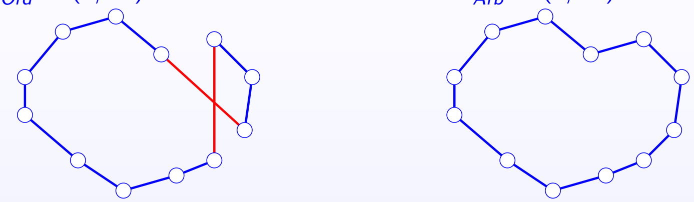
Lemma

For any constant k: MMAS$^*_{Arb}$ with saturated pheromones creates exactly k new edges with probability $\Theta(1)$.

Theorem

MMAS$^*_{Arb}$ needs $O(n^3 \log n)$ iterations in expectation to find optimal solution on Zhou's example.

Probability of particular 2-Opt step (for constant ρ):
MMAS$^*_{Ord}$: $\Theta(1/n^3)$ $\qquad$ MMAS$^*_{Arb}$: $\Theta(1/n^2)$



Average Case Analysis

Assume that n points placed independently, uniformly at random in the unit hypercube $[0,1]^d$.

Theorem [Englert, Röglin, Vöcking 2007]

2-Opt finds after $O(n^{4+1/3} \cdot \log n)$ iterations with probability $1 - o(1)$ a solution with approximation ratio $O(1)$.

Theorem

For $\rho = 1$, MMAS$^*_{Arb}$ finds after $O(n^{6+2/3})$ iterations with probability $1 - o(1)$ a solution with approximation ratio $O(1)$.

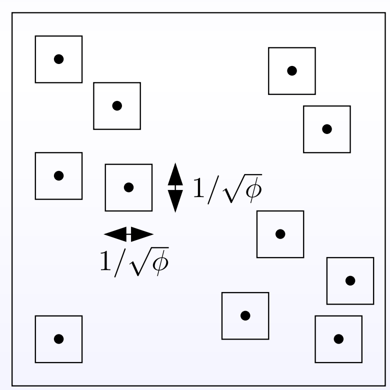
Theorem

For $\rho = 1$, MMAS$^*_{Ord}$ finds after $O(n^{7+2/3})$ iterations with probability $1 - o(1)$ a solution with approximation ratio $O(1)$.

Smoothed Analysis

Smoothed Analysis

Each point $i \in \{1, \dots, n\}$ is chosen independently according to a probability density $f_i : [0,1]^d \to [0, \phi]$.



2-Opt:
$O(\sqrt[d]{\phi})$-approximation in
$O(n^{4+1/3} \cdot \log(n\phi) \cdot \phi^{8/3})$ steps

MMAS$^*_{Ord}$: $O(\sqrt[d]{\phi})$-approximation
in $O(n^{7+2/3} \cdot \phi^3)$ steps

MMAS$^*_{Arb}$: $O(\sqrt[d]{\phi})$-approximation
in $O(n^{6+2/3} \cdot \phi^3)$ steps

TSP: Conclusions and Open Questions

Summary

- $MMAS^*_{Arb}$ has higher locality than $MMAS^*_{Ord}$
- Random and perturbed instances are easy for MMAS* if pheromone update is high.

Open Questions

- Better analysis of random instances for smaller ρ.
- Theoretical analysis of other ACO heuristics.
- Instances on which ACO is better than 2-Opt.

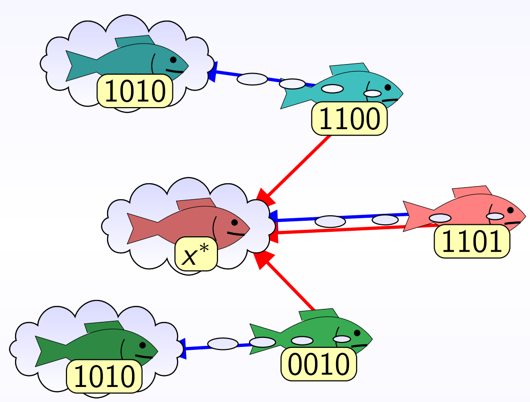
Overview

Particle Swarm Optimization

Particle Swarm Optimization

- Bio-inspired optimization principle developed by Kennedy and Eberhart (1995).
- Mostly applied in continuous spaces.
- Swarm of particles, each moving with its own velocity.
- Velocity is updated according to
  - own best position and
  - position of the best individual in its neighborhood.
- Here: neighborhood = the whole swarm.
- Behavior derived from social-psychology theory.

Particle Swarm Optimization



Binary PSO (Kennedy und Eberhart, 1997)

Binary PSO

Binary PSO

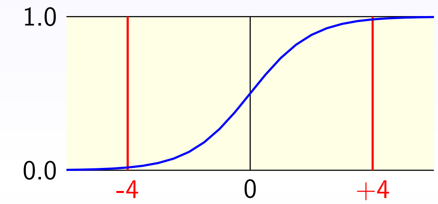
- Developed by Kennedy and Eberhart (1997).
- Goal: optimize pseudo-Boolean function $f\colon \{0,1\}^n \to \mathbb{R}$.
- Swarm contains μ particles.
- Record global best particle x^*.
- The i-th particle maintains triplet
  1. current position $x^{(i)} \in \{0,1\}^n$,
  2. own best position $x^{*(i)} \in \{0,1\}^n$, and
  3. a real-valued velocity $v^{(i)} \in \mathbb{R}$.

What is the meaning of velocity in binary spaces?

Creating New Positions

Probabilistic construction using velocity v and sigmoid function $s(v)$:

$$\text{Prob}(x_j = 1) = s(v_j) = \frac{1}{1+e^{-v_j}}$$



Restrict velocities to $v_j \in [-v_{\max}, +v_{\max}]$.

- Common practice: $v_{\max} = 4$.
- Much better: $v_{\max} := \ln(n-1)$:

$$\frac{1}{n} \leq \text{Prob}(x_j = 1) \leq 1 - \frac{1}{n}.$$

Updating Velocities

Update current velocity vector according to

- cognitive component $\rightarrow$ towards own best: $x^{*(i)} - x^{(i)}$ and
- social component $\rightarrow$ towards global best: $x^* - x^{(i)}$.

Learning rates c_1, c_2 affect weights for the two components.

Random scalars $r_1 \in U[0, c_1]$, $r_2 \in U[0, c_2]$ chosen anew in each generation:

$$v^{(i)} = v^{(i)} + r_1(x^{*(i)} - x^{(i)}) + r_2(x^* - x^{(i)})$$

The Whole Algorithm

Algorithm (Binary PSO)

1. *Initialize velocities with 0^n and all solutions with $\perp$.*
2. *Choose $r_1 \in U[0, c_1]$ and $r_2 \in U[0, c_2]$.*
3. *For $j := 1$ to μ and $i := 1$ to n do*
   Set $x_i^{(j)} := 1$ with probability $s(v_i^{(j)})$, else $x_i^{(j)} := 0$.
4. *For $j := 1$ to μ do*
   If $f(x^{(j)}) > f(x^{*(j)})$ then $x^{*(j)} := x^{(j)}$.
   If $f(x^{*(j)}) > f(x^*)$ then $x^* := x^{*(j)}$.
5. *For $j := 1$ to μ do*
   Set $v^{(j)} := v^{(j)} + r_1(x^{*(j)} - x^{(j)}) + r_2(x^* - x^{(j)})$.
   Restrict each component of $v^{(j)}$ to $[-v_{\max}, v_{\max}]$.
6. *Goto 2.*

The 1-PSO

Special case: 1-PSO with $\mu = 1$, $c_1 = 0$, and $c_2 = 2$ (Sudholt and Witt, 2010).

Algorithm (1-PSO)

1. *Initialize* $v = 0^n$ *and* $x^* = \perp$.
2. *Choose* $r \in U[0, 2]$.
3. *For* $i := 1$ *to* n *do*
   Set $x_i := 1$ *with probability* $s(v_i)$, *else* $x_i := 0$.
4. *If* $f(x) > f(x^*)$ *then* $x^* := x$.
5. *Set* $v := v + r(x^* - x)$.
   Restrict each component of v *to* $[-v_{\max}, v_{\max}]$.
6. *Goto 2.*

Understanding Velocities

1-PSO: update increases velocity by $r(x^* - x)$.

Strange: velocity v_i is changed only if $x_i \neq x_i^*$.

Let $x_i^* = 1$, then probability to increase v_i is

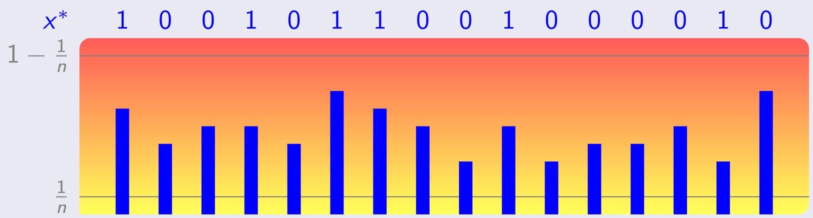
$$1 - s(v_i) = s(-v_i) = \frac{1}{1 + e^{v_i}}.$$

$\Rightarrow$ at least 1/2 for $v_i < 0$, but decreases rapidly with growing v_i.

Velocity Freezing

1-PSO and "social" PSO with $c_1 = 0, c_2 > 0$:

Particle with best-so-far solution



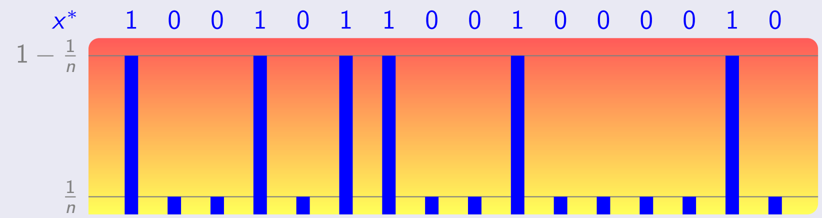
Lemma

Expected freezing time to $v_{\max}$ *or* $-v_{\max}$ *is* $O(n)$ *for single bits and* $O(n \log n)$ *for* n *or* μn *bits if* $\mu = poly(n)$.

Velocity Freezing

1-PSO and "social" PSO with $c_1 = 0, c_2 > 0$:

Particle with best-so-far solution



Lemma

Expected freezing time to $v_{\max}$ *or* $-v_{\max}$ *is* $O(n)$ *for single bits and* $O(n \log n)$ *for* n *or* μn *bits if* $\mu = poly(n)$.

Fitness-Level Method for Binary PSO

Let s_i be the minimum probability of the (1+1) EA to increase the fitness from i-th fitness value.

Upper bound for the (1+1) EA

$$\sum_{i=0}^{m-1} \frac{1}{s_i}$$

Upper bound for the 1-PSO

$$O(m \cdot n \log n) + \sum_{i=0}^{m-1} \frac{1}{s_i}$$

Upper bound for generations of Binary PSO with $c_1 := 0, c_2 := 2$

$$O\left(m \cdot n \log n + \frac{1}{\mu} \sum_{i=0}^{m-1} \frac{1}{s_i}\right)$$

The 1-PSO on OneMax

Fitness level arguments only yield $O(n^2 \log n)$ for the 1-PSO on OneMax.

More careful inspection of the velocities: average adaptation time of $384 \ln n$ is sufficient.

Theorem (Sudholt and Witt, 2010)

The expected optimization time of the 1-PSO *on* OneMax *is* $O(n \log n)$.

Proof uses layering argument and amortized analysis.

Experiments: 1-PSO 15% slower than (1+1) EA on OneMax.

Overview

Continuous PSO

Search space: (bounded subspace of) $\mathbb{R}^n$.

Objective function: $f \colon \mathbb{R}^n \to \mathbb{R}$.

Particles represent positions $x^{(i)}$ in this space.

Particles fly at certain velocity: $x^{(i)} := x^{(i)} + v^{(i)}$.

Velocity update with inertia weight ω:

$$v^{(i)} = \omega v^{(i)} + r_1(x^{*(i)} - x^{(i)}) + r_2(x^* - x^{(i)})$$

Convergence of PSO

Swarm can collapse to points or other low-dimensional subspaces.

Convergence results for standard PSO, $\omega < 1$ (Jiang, Luo, and Yang, 2007)

PSO converges . . . somewhere.

Extensions of standard PSO

- Bare-bones PSO (Kennedy, 2003)
- PSO with mutation (several variants)
- PSO using gradient information (several variants)
- Guaranteed Convergence PSO (GCPSO) (van den Bergh and Engelbrecht, 2002)

Guaranteed Convergence PSO

Van den Bergh and Engelbrecht, 2002:

- Make a cube mutation of a particle's position by adding $p \in U[-\ell, \ell]^n$.
- Adapt "step size" ℓ in the course of the run by doubling or halving it, depending on the number of successes.

Possible step size adaptation (Witt, 2009)

After an observation phase consisting of n steps has elapsed, double ℓ if the total number of successes was at least $n/5$ in the phase and halve it otherwise. Then start a new phase.

$\longrightarrow$ 1/5-rule known from evolution strategies!

Special Case of GCPSO

GCPSO with one particle (for minimization):

GCPSO_1

Repeat:

- $x := x^* + p$, $p \in U[-\ell, \ell]^n$.
- if $f(x) < f(x^*)$ then $x^* := x$.
- Update ℓ.

Basically a (1+1) ES with cube mutation.

Can be analyzed like classical (1+1) EA (Jägersküpper, 2007)

Results

$$\textsc{Sphere}(x) := ||x|| = x_1^2 + x_2^2 + \cdots + x_n^2$$

Theorem (Witt, 2009)

Consider the GCPSO_1 on Sphere. *If $\ell = \Theta(||x^*||/n)$ for the initial solution x^*, the runtime until the distance to the optimum is no more than $\varepsilon||x^*||$ is $O(n \log(1/\varepsilon))$ with probability at least $1 - 2^{-\Omega(n)}$ provided that $2^{-n^{O(1)}} \leq \varepsilon \leq 1$.*

Same result as for (1+1) ES using Gaussian mutations in Jägersküpper, 2007.

Remarks

- Analysis of cube mutations is easier than that of Gaussian mutations for Sphere.
- Runtime result for GCPSO_1 is asymptotically optimal for many black-box heuristics (Jägersküpper, 2007a).
- Populations do not help for Sphere (Jägersküpper and Witt, 2005).

Overview

Conclusions

Summary

- Insight into probabilistic models underlying ACO and PSO
- How design choices and parameters affect (bounds on) running times
- How simple ACO algorithms optimize unimodal functions and plateaus
- Results for ACO in combinatorial optimization
- First analyses of basic PSO algorithms in discrete and continuous spaces

Future Work

- A unified theory of randomized search heuristics?
- More results on multimodal problems
- When and how diversity and slow adaptation help
- ACO: average-case results, possibly with heuristic information
- PSO: swarm dynamics and neighborhood topologies

Selected Literature I

Conference/workshop papers superseded by journal papers are omitted.

N. Attiratanasunthron and J. Fakcharoenphol.
A running time analysis of an ant colony optimization algorithm for shortest paths in directed acyclic graphs.
Information Processing Letters, 105(3):88–92, 2008.

B. Doerr and D. Johannsen.
Refined runtime analysis of a basic ant colony optimization algorithm.
In *Proceedings of the Congress of Evolutionary Computation (CEC '07)*, pages 501–507. IEEE Press, 2007.

B. Doerr, D. Johannsen, and C. H. Tang.
How single ant ACO systems optimize pseudo-Boolean functions.
In *Parallel Problem Solving from Nature (PPSN X)*, pages 378–388. Springer, 2008.

B. Doerr, F. Neumann, D. Sudholt, and C. Witt.
Runtime analysis of the 1-ANT ant colony optimizer.
Theoretical Computer Science, 412(17):1629–1644, 2011.

M. Dorigo and C. Blum.
Ant colony optimization theory: A survey.
Theoretical Computer Science, 344:243–278, 2005.

M. Dorigo and T. Stützle.
Ant Colony Optimization.
MIT Press, 2004.

W. J. Gutjahr.
On the finite-time dynamics of ant colony optimization.
Methodology and Computing in Applied Probability, 8:105–133, 2006.

Selected Literature II

W. J. Gutjahr.
Mathematical runtime analysis of ACO algorithms: Survey on an emerging issue.
Swarm Intelligence, 1:59–79, 2007.

W. J. Gutjahr.
First steps to the runtime complexity analysis of ant colony optimization.
Computers and Operations Research, 35(9):2711–2727, 2008.

W. J. Gutjahr.
Ant colony optimization: recent developments in theoretical analysis.
In *Theory of Randomized Search Heuristics–Foundations and Recent Developments*. World Scientific Publishing, 2011.

W. J. Gutjahr and G. Sebastiani.
Runtime analysis of ant colony optimization with best-so-far reinforcement.
Methodology and Computing in Applied Probability, 10:409–433, 2008.

C. Horoba and D. Sudholt.
Running time analysis of ACO systems for shortest path problems.
In *Proceedings of Engineering Stochastic Local Search Algorithms (SLS '09)*, volume 5752 of *LNCS*, pages 76–91. Springer, 2009.

C. Horoba and D. Sudholt.
Ant colony optimization for stochastic shortest path problems.
In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2010)*, pages 1465–1472, 2010.

J. Kennedy and R. C. Eberhart.
Particle swarm optimization.
In *Proceedings of the IEEE International Conference on Neural Networks*, pages 1942–1948. IEEE Press, 1995.

J. Kennedy and R. C. Eberhart.
A discrete binary version of the particle swarm algorithm.
In *Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics (WMSCI)*, pages 4104–4109, 1997.

Selected Literature III

J. Kennedy, R. C. Eberhart, and Y. Shi.
Swarm Intelligence.
Morgan Kaufmann, 2001.

T. Kötzing, P. K. Lehre, P. S. Oliveto, and F. Neumann.
Ant colony optimization and the minimum cut problem.
In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '10)*, pages 1393–1400. ACM, 2010.

T. Kötzing, F. Neumann, H. Röglin, and C. Witt.
Theoretical properties of two ACO approaches for the traveling salesman problem.
In *Seventh International Conference on Ant Colony Optimization and Swarm Intelligence (ANTS 2010)*, volume 6234 of *LNCS*, pages 324–335. Springer, 2010.

T. Kötzing, F. Neumann, D. Sudholt, and M. Wagner.
Simple Max-Min ant systems and the optimization of linear pseudo-Boolean functions.
In *Proceedings of the eleventh workshop on Foundations of Genetic Algorithms (FOGA 2011)*. ACM Press, 2011, to appear.

F. Neumann, D. Sudholt, and C. Witt.
Rigorous analyses for the combination of ant colony optimization and local search.
In *Proceedings of the Sixth International Conference on Ant Colony Optimization and Swarm Intelligence (ANTS '08)*, volume 5217 of *LNCS*, pages 132–143. Springer, 2008.

F. Neumann, D. Sudholt, and C. Witt.
Analysis of different MMAS ACO algorithms on unimodal functions and plateaus.
Swarm Intelligence, 3(1):35–68, 2009.

F. Neumann, D. Sudholt, and C. Witt.
Computational complexity of ant colony optimization and its hybridization with local search.
In C. P. Lim, L. C. Jain, and S. Dehuri, editors, *Innovations in Swarm Intelligence*, number 248 in SGI. Springer, 2009.

Selected Literature IV

F. Neumann, D. Sudholt, and C. Witt.
A few ants are enough: ACO with iteration-best update.
In *Genetic and Evolutionary Computation Conference (GECCO '10)*, pages 63–70, 2010.

F. Neumann and C. Witt.
Ant Colony Optimization and the minimum spanning tree problem.
In *Proceedings of Learning and Intelligent Optimization (LION '07)*, volume 5313 of *LNCS*, pages 153–166. Springer, 2008.

F. Neumann and C. Witt.
Runtime analysis of a simple ant colony optimization algorithm.
Algorithmica, 54(2):243–255, 2009.

T. Stützle and H. H. Hoos.
MAX-MIN ant system.
Journal of Future Generation Computer Systems, 16:889–914, 2000.

D. Sudholt.
Using Markov-chain mixing time estimates for the analysis of ant colony optimization.
In *Proceedings of the eleventh workshop on Foundations of Genetic Algorithms (FOGA 2011)*. ACM Press, 2011, to appear.

D. Sudholt and C. Witt.
Runtime analysis of a binary particle swarm optimizer.
Theoretical Computer Science, 411(21):2084–2100, 2010.

C. Witt.
Rigorous runtime analysis of swarm intelligence algorithms - an overview.
In *Swarm Intelligence for Multi-objective Problems in Data Mining*, number 242 in Studies in Computational Intelligence (SCI), pages 157–177. Springer, 2009.

Selected Literature V

C. Witt.
Why standard particle swarm optimisers elude a theoretical runtime analysis.
In *Foundations of Genetic Algorithms 10 (FOGA '09)*, pages 13–20. ACM Press, 2009.

C. Witt.
Theory of particle swarm optimization.
In *Theory of Randomized Search Heuristics–Foundations and Recent Developments*. World Scientific Publishing, 2011.

Y. Zhou.
Runtime analysis of an ant colony optimization algorithm for TSP instances.
IEEE Transactions on Evolutionary Computation, 13(5):1083–1092, 2009.

Thank you!

Questions?