Development of an automatically configurable ant colony optimization framework. State of art.

Aldar Saranov

March 9, 2017

Contents

Abstract

Some animal species show an extreme degree of social organization. Such species (e.g. ants) have pheromone production and detection body parts and therefore seize an ability to communicate between each other in indirect way. This concept has inspired the development of algorithms which are based on social behavior of population called ant colony optimization algorithms (ACO). These algorithms allow to solve NP-hard problems in a very efficient manner. Since these algorithms are considered metaheuristic the development of a ACO framework is the next step of formalizing of this area is to provide tools for resolving general optimization problems. This article gives the brief overview of the current ACO research area state, existing framework description and some tools which can be used for the framework automatic configuration.

1 Introduction (1 page)

Section descriptions. Heuristic information. Pheromones. Constructive heuristics.

2 Combinatorial Optimization Problems and Constructive Heuristics (17 pages)

Definition

Optimization Problem is a tuple (ϕ, ω, f), where

- ϕ is a search space consisting of all possible assignments of discrete variables x_i, with $i = 1, ..., n$
- ω is a set of constraints for the decision variables
- $f : \phi \to R$ is an objective function which has to be optimized

TSP and QAP description. Solution components. Feasible solution.

3 The ACO Algorithmic Framework

ACO algorithm.

```
Algorithm


procedure ACO-Metaheuristic
        repeat
                for each ant do
                        repeat
                                ExtendPartialSolutionProbabilistically()
                        until solution is complete
                for each ant ∈ SelectAntsForLocalSearch() do
                        ApplyLocalSearch(ant)
                EvaporatePheromones()
                DepositPheromones()
        until termination criteria met
end
```