UNIVERSITÉ LIBRE DE BRUXELLES
Faculté des Sciences
Département d'Informatique

Development of an automatically configurable ant colony optimization framework

Aldar Saranov

Promoter : Prof. Thomas Stützle   Master Thesis in Computer Sciences

*Dedicated to my mother Lena, who
was always sincerely supporting me*

"If one does not know to which port one is sailing, no wind is favorable."

Lucius Annaeus Seneca, 1st century AD

"The general, unable to control his irritation, will launch his men to the assault like swarming ants, with the result that one-third of his men are slain, while the town still remains untaken. Such are the disastrous effects of a siege."

Sun Tzu, "The Art of War", 5th century BC

Acknowledgment

Contents

Chapter 1

Introduction 4-5

Some 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 an indirect way. This concept has inspired the development of algorithms, which are based on the social behavior of the ant colonies called ant colony optimization algorithms. These algorithms allow to solve NP-hard problems in a very efficient manner. These algorithms are considered to be metaheuristics. The development of an ACO framework is the next step of formalizing this area. Such a framework can then be used as a tool to help resolving various optimization problems. This report gives a brief overview of the current state of the ACO research area, existing framework description and some tools which can be used for the automatic configuration of the framework.

In the chapter 2, we describe the theoretical foundations of the framework, that is required to develop, and an efficient configuration software, that allows to obtain high-performing configurations for further framework exploiting.

The development of such framework would also require its application to some particular problem for testing and analysis purposes. Vehicle Routing Problem was chosen as the one. In the chapter 3, we introduce formulation of Vehicle Routing Problem and also mention several variations thereof.

In the chapter 4, we describe thorough implementation details and decisions for the ACO framework, the way we adapt it specifically to VRP-problems and also show line-by-line description of the parameter space defined for the framework.

In the chapter 5, one can see the experimental set-up, tuning results and interpretations that are made.

Chapter 2

Background 20

2.1 Combinatorial Optimization Problems and Constructive Heuristics 5

Combinatorial optimization problems (COPs) are a class of mathematical optimization problems. These problems can be described as a grouping, ordering, assignment or any other operations over a set of discrete objects. In practice, one may need to resolve COPs, which have a large number of extra constraints for the solutions to consider them as admissible. Many of these problems which are still being thoroughly researched at the moment, belong to NP-hard discrete optimization problems. The best performing algorithms known today to solve such problems have a worst-case run-time larger than polynomial (e.g. exponential).

An 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 on the decision variables

- $f : \Phi \to R$ is an objective function, which has to be optimized

---

2.2 Ant Colony Optimization 10

[rip-off the previous year paper, describe only those components that are used]

[ACOTSPQAP brief description]

2.3   Iterated F-Race 5

[description similar to the slides of I-RACE description]

Chapter 3

Vehicle Routing Problem 5

Chapter 4

Implementation 20-25

[framework class-level description, no class diagram due to large size (however no descriptions for every class method. it's not a documentation)]

[mention the back-bone classes that provide generality and maintaining of VRP instance generality]

[framework parameters space]

[mention unit testing???]

Chapter 5

Experimental 20-25

[public samples description, some publicly know plots]

[configuration of i-race]

[obtained result configuration for the framework]

[description of the configuration]

Chapter 6

Conclusion 4-5

6.1 Literature