Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 20 (2013) 46 – 51
Procedia
Computer Science

Complex Adaptive Systems, Publication 3
Cihan H. Dagli, Editor in Chief
Conference Organized by Missouri University of Science and Technology
2013- Baltimore, MD

# Employing Learning to Improve the Performance of Meta-RaPS

Fatemah Al-Duoli[a], Ghaith Rabadi[a,b]*

[a]Department of Engineering Management and Systems Engineering, Old Dominion University, Norfolk, Virginia 23529, USA
[b]Visiting Associate Professor, Mechanical and Industrial Engineering Department, Qatar University, Qatar

**Abstract**

In their search for satisfactory solutions to complex combinatorial problems, metaheuristics methods are expected to intelligently explore the solution space. Various forms of memory have been used to achieve this goal and improve the performance of metaheuristics, which warranted the development of the Adaptive Memory Programming (AMP) framework [1]. This paper follows this framework by integrating Machine Learning (ML) concepts into metaheuristics as a way to guide metaheuristics while searching for solutions. The target metaheuristic method is Meta-heuristic for Randomized Priority Search (Meta-RaPS). Similar to most metaheuristics, Meta-RaPS consists of construction and improvement phases. Randomness coupled with a greedy heuristic are typically employed in the construction phase. While a local search heuristic is used in the improvement phase. This paper proposes adding a new learning phase, in which a ML method will be integrated. An Inductive Decision Tree (IDT) will be incorporated into the learning phase in an effort to learn from the information collected during the construction and improvement phases. The proposed approach will be demonstrated using instances for the Capacitated Vehicle Routing Problem (CVRP).

*Keywords:* Vehicle Routing Problem; Metaheuristics; Meta-RaPS; Supervised Learning

## 1. Introduction

Metaheuristic are approximate optimization methods that define search strategies and use heuristics to solve large combinatorial problems in a reasonable time [2 – old 16]. Utilizing memory is an important feature of many metaheuristic methods [1]. With memory comes the possibility of learning and building strategies that aim at finding better solutions more efficiently. The first metaheuristic method to explicitly use memory is Tabu Search [3]. However, Taillard et al (2001) [1] noted that the concept of memory can be found in many metaheuristic methods including the population in Genetic Algorithms and the pheromone trail in Ant Colony Optimization. The various forms of memory used in metaheuristics prompted Taillard et al. [1] to introduce the Adaptive Memory Programming (AMP) framework, which formalized the incorporation of memory in metaheuristics. Memory allows a variety of information to be collected during the process of solving a combinatorial problem. Converting the memory into knowledge via a learning technique helps improve the performance of metaheuristics. Integrating

* Corresponding author. Tel.: +0-757-683-4918; fax: +0-757-683-5640.
*E-mail address*: grabadi@odu.edu.


doi:10.1016/j.procs.2013.09.237

Machine Learning (ML) algorithms into metaheuristics is one of the approaches used to make use of the memory and improve the performance of metaheuristics.

ML algorithms can be categorized according to the training samples used in the learning process. A training sample consists of data instances and/or labels depending on the learning approach, where a training instance is a set of attributes and a label is the desired prediction of that instance (or a class that the instance belongs to) [4]. Unsupervised algorithms accept unlabeled training set. Semi-supervised learning algorithms accept mixed (labeled and unlabeled) instances. Supervised learning algorithms accept labeled data instances[5]. The labeled training samples in supervised learning algorithms represent past experiences. These experiences help a supervised algorithm in making future decisions when faced with new data.

Meta-RaPS (Meta-heuristic for Randomized Priority Search) is one of the latest metaheuristics that fit under the AMP framework. This was demonstrated by Lan and DePuy [6], who incorporated memory into the memory-less Meta-RaPS baseline. Mirza [7] and Arin and Rabadi [8] have also incorporated memory and learning via integrating ML algorithms. Their efforts focused on incorporating unsupervised and semi-supervised ML algorithms. In this paper, the proposed concept aims at incorporating a supervised ML algorithm, namely *Inductive Decision Trees (IDT's)*, into Meta-RaPS and apply it to the Capacitated Vehicle Routing Problem (CVRP).

The CVRP is a variation of the Vehicle Routing Problem (VRP), which is a combinatorial problem that was introduced in the 1960s [9] for discovering optimized routes from a central depot to several customers. The VRP and its variants have been studied and solved using both exact and approximate algorithms [10]. The VRP is considered an NP-hard problem with no exact algorithms for instances that have >75 customers [11].

## 2. Meta-RaPS

Meta-RaPS is a metaheuristic method initially introduced as a modified form of COMSOAL heuristic (Computer Method of Sequencing Operations for Assembly Lines) [12]. Moraga et. al [14] formally defined Meta-RaPS as a "generic, high-level search procedures that introduce randomness to a construction heuristic as a device to avoid getting trapped at a local optimal solution".

Similar to other metaheuristics, Meta-RaPS divides the process of solving a problem into a construction phase and an improvement phase. Throughout these phases, Meta-RaPS uses four parameters: number of iterations (I), priority percentage (p%), restriction percentage (r%), and improvement percentage (i%).

In every iteration, Meta-RaPS solves a combinatorial problem by incrementally building a solution in the construction phase and possibly improving it in the improvement phase. During the construction phase, Meta-RaPS employs a greedy heuristic to identify a set of feasible next moves. The set of feasible next moves are prioritized. The move with the best priority is selected p% of the time. During the remaining times, (100 - p%), the next feasible move is selected randomly from a restricted list and added to the solution. The restricted list contains a set of next moves with priority values that are within the best r% of the best priority move. The construction process is repeated until a solution is complete. Subsequently, the constructed solution may be improved if its value is within i% of the best constructed solution to avoid wasting time on a very inferior solution. A local search heuristic is typically employed in an attempt to improve the constructed solution. The best solution produced from all iterations is reported as the output of Meta-RaPS.

Since its introduction, Meta-RaPS has been successfully used to solve several combinatorial problems including the Resource Constrained Project Scheduling Problem [12], the Vehicle Routing Problem [15], the Traveling Salesman Problem [2], the 0-1 Multidimensional Knapsack Problem [16], the Unrelated Parallel Machine Scheduling Problem with Setup Times [17], the Early/Tardy Single Machine Scheduling Problem [18], the Spatial Scheduling Problem with Release Times [19], and the Aerial Refueling Scheduling Problem with Due Data to Deadline Windows and Release Time [20].

## 3. Inductive Decision Tree (IDT)

Supervised machine learning can be achieved using Inductive Decision Trees (IDT's). The trees are logical trees that generate rules to classify data. The IDT's are constructed using a labeled training sample. Learning is achieved by using the rules (generated by the tree) in classifying new data. Several algorithms are available to construct IDT's. The algorithm used here is the ID3 algorithm introduced by Quinlan [21]. ID3 constructs an IDT recursively