MODIFIED AND ENSEMBLE INTELLIGENT WATER DROP ALGORITHMS AND THEIR APPLICATIONS

BASEM O. F. ALIJLA

UNIVERSITI SAINS MALAYSIA

2015

MODIFIED AND ENSEMBLE INTELLIGENT WATER DROP ALGORITHMS AND THEIR APPLICATIONS

by

BASEM O. F. ALIJLA

Thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosphy

October 2015

ACKNOWLEDGEMENTS

With the name of Allah, Most Gracious, Most Merciful.

First and foremost, my sincere glorifications and adorations go to almighty Allah for his guidance, protection and strength over me to complete my research study. I would like to express sincere gratitude to my supervisors, Professor Lim Chee Peng, Professor Ahamad Tajudin Khader, and Dr. Wong Li Pei for their insightful guidance throughout the entire period of my research. I will forever remain grateful for the continued help, and wise counseling. Special thanks and appreciations to Professor Lim Chee Peng for his endless help, especially in publishing my research work. I am also grateful to my friend Dr. Mohammad Azmi Al-Betar for his help and useful discussion during the course of this research.

Indeed, without Allah then my parents' prayers, I could not have completed this research. My special thanks to my beloved parents for their continued supports, encouragements, and prayers. Thank you mum, dad, my brothers, and sisters, you are always in my mind and heart. In the same context, love and support my deepest and special thanks go to my wife for her patience, understanding, fervent prayers and for taking care of our kids. Your great support, efforts, and endurance is highly appreciated.

My sincere appreciations are extended to the Islamic Development Bank (IDB) for its financial support under the IDB merit scholarship program. I am also grateful to the School of Computer Sciences, Universiti Sains Malaysia, which has provided all the facilities and equipments needed throughout my research. Last but not least; thank you to all my friends in Malaysia, and those who support me in any respect during my research.

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
ACS	Ant Colony System
AIWD	Adaptive Intelligent Water Drops
CFS	Correlation-based Feature Selection
CLB	Creek Local Best
EC	Evolutionary Computation
EIWD	Enhanced Intelligent Water Drops
ERS-IWD	Exponential Ranking selection Intelligent Water Drops
FCV	Fold Cross Validation
FFT	Fast Fourier Transform
FPS	Fitness Proportionate Selection
FPS-IWD	Fitness Proportionate Selection Intelligent Water Drops
FRFS	Fuzzy Rough Feature subset Selection
FS	Feature Selection
FS-MRMC-IWD	Feature Selection Master River Multiple Creeks Intelligent Water Drops
GA	Genetic Algorithm
GD	Great Deluge
HC	Hill Climbing
HMC	Harmony Memory Consideration
HS	Harmony Search
IILS	Iterative Improvement Local Search

IWD	Intelligent Water Drops
IWD-CO	IWD-Continuous Optimization
LAHC	Late Acceptance Hill Climbing
LRS-IWD	Linear Ranking Selection Intelligent Water Drops
MACO	Mutated Ant Colony Optimization
MHC	Multiple Hill Climbing
MKP	Multiple Knapsack Problem
MLB	Master Local Best
MRMC-IWD	Master River Multiple Creeks Intelligent Water Drops
NB	Naive Bayes
NP-hard	Non-deterministic Polynomial-time hard
PA	Pitch Adjustment
PSO	Particle Swarm Optimization
QoS	Quality of Service
RC	Random Consideration
RMHC	Random Mutation Hill Climbing
RMS	Root Mean Square
RS	Rough Set
RSFS	Rough Set Feature Subset Selection
RST	Rough Set Theory
SA	Simulated Annealing
SI	Swarm Intelligence
SP	Selection Pressure
std	Standard Deviation

SVM	Support Vector Machine
TSP	Travelling Salesman Problem
UCI	University of California Irvine machine learning repository
USM	Universiti Sains Malaysia
VQNN	Vaguely Quantified Nearest Neighbor
WEKA	Waikato Environment for Knowledge Analysis

ALGORITMA TITISAN AIR CERDAS TERUBAH SUAI DAN GABUNGAN SERTA APLIKASINYA

ABSTRAK

Algoritma Titisan Air Cerdas (TAC) ialah model berasaskan kawanan yang sememangnya berguna untuk mengatasi masalah-masalah pengoptimuman. Tujuan utama kajian ini adalah untuk meningkatkan keupayaan algoritma TAC dan mengatasi keterbatasan algoritma tersebut, yang berkaitan dengan kepelbagaian populasi serta mengimbangangi penerokaan dan pengeksploitasian dalam menangani masalah-masalah pengoptimuman. Pertama, algoritma TAC yang diubahsuai, diperkenalkan. Dua kaedah pemilihan berdasarkan kedudukan, iaitu kedudukan linear dan kedudukan eksponen, dicadangkan untuk menggantikan kaedah pemilihan kelekapan yang seimbang. Kedua, algoritma Titisan Air Cerdas yang berdasarkan Sungai Induk Pelbagai Caruk Alir Sungai (SICAS-TAC) dicadangkan untuk mengeksploitasikan keupayaan penerokaan algoritma TAC yang diubahsuai. Di samping itu, model hibrid SICAS-TAC juga dibentangkan. Model hibrid ini menggabungkan algoritma SICAS-TAC dengan peningkatan lelaran carian setempat, untuk meningkatkan keupayaan penjelajahan kepada algoritma SICAS-TAC. Keberkesanan model-model yang dicadangkan dinilai secara sistematik dan menyeluruh dengan menggunakan tiga masalah pengoptimuman kombinatorik iaitu, masalah pemilihan ciri subset berdasarkan set kasar, masalah beg galas berbilang, dan masalah jurujual kembara. Kesesuaian dan keberkesanan model hibrid SICAS-TAC disiasat dengan menyelesaikan masalah pengoptimuman dunia sebenar yang berkaitan dengan pemilihan ciri dan klasifikasi. Beberapa set data tanda aras umum dan dua masalah dunia sebenar, iaitu masalah pengesanan pergerakan manusia dan masalah pengesanan kerosakan motor, telah dikaji. Keputusan kajian telah menunjukkan keberkesanan model-model yang dicadangkan dalam meningkatkan prestasi algoritma TAC yang asal dan juga menyelesaikan masalah-masalah pengoptimuman dunia sebenar.

MODIFIED AND ENSEMBLE INTELLIGENT WATER DROP ALGORITHMS AND THEIR APPLICATIONS

ABSTRACT

The Intelligent Water Drop (IWD) algorithm is a swarm-based model that is useful for undertaking optimization problems. The main aim of this research is to enhance the IWD algorithm and overcome its limitations pertaining to population diversity, as well as balanced exploration and exploitation in handling optimization problems. Firstly, a modified IWD algorithm is introduced. Two ranking-based selection methods, i.e. linear ranking and exponential ranking, are proposed to replace the fitness proportionate selection method. Secondly, the Master River Multiple Creeks Intelligent Water Drops (MRMC-IWD) algorithm is proposed in an attempt to exploit the exploration capability of the modified IWD algorithm. In addition, the hybrid MRMC-IWD model is proposed. It combines MRMC-IWD with the iterated improvement local search method, to empower MRMC-IWD with the exploitation capability. The usefulness of the proposed models is evaluated systematically and comprehensively using three combinatorial optimization problems, i.e., rough set feature subset selection, multiple knapsack problem, and travelling salesman problem. The applicability of the hybrid MRMC-IWD model is investigated to solving real-world optimization problems related to feature selection and classification tasks. A number of publicly available benchmark data sets and two real-world problems, namely human motion detection and motor fault detection, are studied. The results ascertain the effectiveness of the

proposed models in improving the performance of the original IWD algorithm as well as undertaking real-world optimization problems.

CHAPTER 1

INTRODUCTION

1.1 Introduction

Optimization is a process that concerns with finding the best solution of a given problem from among the possible solutions within an affordable time and cost (Weise et al., 2009). The first step in the optimization process is formulating the optimization problem through an objective function and a set of constrains that encompass the problem search space (i.e., regions of feasible solutions). Every alternative (i.e., solution) is represented by a set of decision variables. Each decision variable has a domain, which is a representation of the set of all possible values that the decision variable can take. The second step in optimization starts by utilizing an optimization method (i.e., search method) to find the best candidate solutions. Candidate solution has a configuration of decision variables that satisfies the set of problem constrains, and that maximizes or minimizes the objective function (Boussaid et al., 2013). It converges to the optimal solution (i.e., local or global optimal solution) by reaching the optimal values of the decision variables. Figure 1.1 depicts a 3D-fitness landscape of an optimization problem. It shows the concept of the local and global optima, where the local optimal solution is not necessarily the same as the global one (Weise et al., 2009). Optimization can be applied to many real-world problems in various domains. As an example, mathematicians apply optimization methods to identify the best outcome pertaining to some mathematical functions within a range of variables (Vesterstrom and Thomsen, 2004). In the presence of conflicting criteria, engineers use optimization methods to

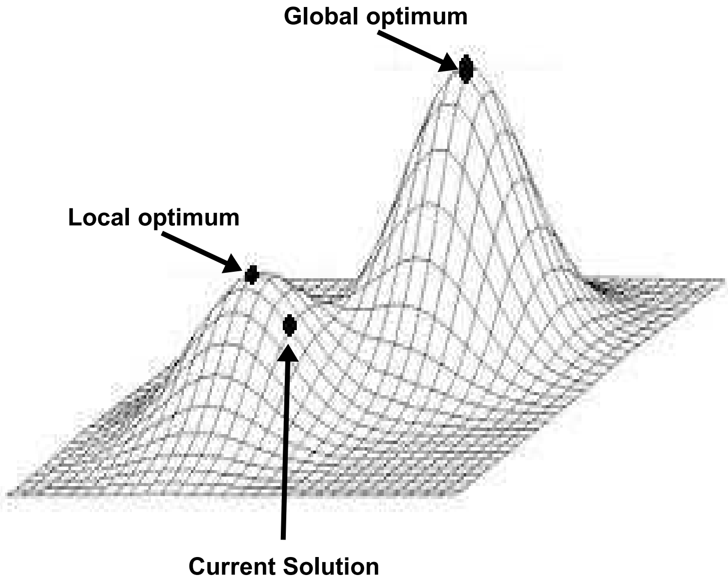

Figure 1.1: A 3D-fitness landscape of an optimization problem (Coppin, 2004).

find the best performance of a model subject to certain criteria, e.g. cost, profit, and quality (Machado et al., 2001; Marler and Arora, 2004; Yildiz, 2009).

In general, optimization problems can be categorized into several categories, depending on whether they are discrete or continuous, single objective or multi-objectives, and constrained or unconstrained (Boussaid et al., 2013). They can be classified into discrete or continuous based on the domain of encoding the solution. Solutions of continuous problems are encoded with *real-valued* variables, while solutions of discrete problems are encoded with *discrete* variables. In this light, discrete optimization problems, which are known as combinatorial optimization problems (COPs), include problems that have a finite set of solutions (Blum and Roli, 2003). Optimization problems can be categorized into constrained and unconstrained based on whether the decision variables are restricted to some limitations (i.e., constrains) or otherwise. The number of objective function is the distinctive property that differentiates between single-objective and multi-objectives optimization problems (Marler and Arora, 2004).

Numerous optimization methods have been devised and successfully applied to solving optimization problems. Generally, they can be classified into two main categories: deterministic (exact) and non-deterministic (stochastic) methods (Lin et al., 2012). Deterministic methods such as linear programming and dynamic programming exhaustively employ the analytical properties of a problem to search for the optimal solution. However, no method can be guaranteed to find the optimal solution especially for NP-hard problems (i.e. problems that have no known solution in polynomial time) (Lin et al., 2012). Non-deterministic methods search with some randomness to solve NP-hard problems to achieve good (near-optimal) solutions in polynomial time. In this regards, meta-heuristic methods play a major role in tackling optimization problems. They utilize heuristic information within a high-level problem-agonistic framework to solve optimization problems. In this context, a branch of meta-heuristic optimization methods that has attracted much attention of researchers is emulating the natural behaviors of real systems in solving optimization problems. These methods are known as nature-inspired meta-heuristics (Yang, 2010). As an example, the genetic algorithm (GA) (Holland, 1975; Goldberg and Holland, 1988) is inspired by biological evolution of organisms, such as inheritance, mutation, crossover, and selection, to solve optimization problems. An innovative family of nature-inspired models known as swarm intelligence (SI) has emerged (Blum and Li, 2008). SI methods are based on the phenomena of different natural swarms, e.g. ant colony optimization (ACO) inspired by the foraging behavior of real ants (Dorigo and Di Caro, 1999; Dorigo and Blum, 2005), particle swarm optimization (PSO) inspired by the social behavior of bird flocking or fish schooling (Shi, 2001; Kennedy, 2010), artificial bee colony (ABC) inspired by the foraging behavior of honey bees in their colony (Karaboga, 2005). A variety of SI-based methods have been successfully used in solving different optimization problems. They are characterized by collaborative learning, i.e., a population of agents collaborates and cooperates among themselves within their environment to solve a problem (Blum and Li, 2008). Furthermore, the interactions among agents enable the model to explore several regions of the search space simultaneously, in order to converge to the global optimum solution in an effective manner (Blum and Li, 2008).

The Intelligent Water Drop (IWD) algorithm (Shah-Hosseini, 2007) is a relatively recent SI model. It is inspired by the natural phenomenon of water drops flowing with soil and velocity along a river. It imitates a number of natural phenomena pertaining to the water drops flowing through an easier path in a river, i.e., a path with less barriers and obstacles. Technically, the IWD algorithm is a constructive-based meta-heuristic algorithm (Shah-Hosseini, 2007) that comprises a set of cooperative computational agents (water drops) iteratively constructing the solution of a problem. The water drop constructs a solution by traversing a path with a finite set of discrete movements. It begins the process with an initial state. Thereafter, it iteratively moves step-by-step passing through several intermediate states (partial solutions) until a final sate (complete solution) is reached. A probabilistic approach is used to control the movements of the water drops. At every iteration of the IWD algorithm, a new complete population (i.e., a set of solutions) is generated. The new generation of solutions benefits from the previous generation through the environment attributes, i.e., soil and velocity. They are used to control the probability distribution of selecting the candidate movements, and to extend the partial solution. The soil level indicates the cumulative proficiency of a particular movement. It represents the communication mechanism that enables the water drops to cooperate among themselves. The velocity is an attribute that influences the dynamics updating process of the soil level based on heuristic information, which is related to the problem under scrutiny.

Although the IWD algorithm has been successfully employed to solve numerous optimization problems (i.e., combinatorial, continuous, and multi-objectives) from different application fields (Siddique and Adeli, 2014), little efforts have been made by researchers in investigating the fundamental algorithmic aspects of IWD. Many researchers focus on the application field of IWD as an optimization method. This research is focused on investigating the algorithmic aspects of the IWD algorithm to tackle optimization problems, i.e., how to preserve population diversity and balance exploration and exploitation of the search process.

The rest of this chapter is organized as follows. Section 1.2 provides the research motivation and problem statement. Sections 1.3 and 1.4, respectively, present the research objectives and contributions. An overview of the research methodology is presented in Section 1.5. Section 1.6 introduces the research scope. Section 1.7 explains thesis structure with indication to the contents of each chapter.

1.2 Motivation and Problem Statement

The IWD algorithm was proposed by Shah-Hosseini (2007), adding a new SI-based nature-inspired optimization method to the literature. It has been shown to be effective in solving COPs, such as travelling salesman problem (TSP), multiple knapsack problem (MKP), and n-queen puzzle problem (Shah-Hosseini, 2007, 2008, 2012a,b). As a new meta-heuristic optimization method, IWD has also been successfully applied to solving numerous optimization problems in different fields (Siddique and Adeli, 2014).

However, research to enhance the performance of IWD in solving COPs is still active. In Niu et al. (2012), five modified schemes that explore three IWD operators (i.e., soil and velocity values, transition rule, and soil update mechanism) were proposed to enhance the IWD performance. These schemes could overcome the early convergence and population diversity problems in IWD. Therefore, the key motivation of this research is to investigate the fundamental algorithmic aspects (i.e., population diversity as well as balance in exploration and exploitation) to enhance the performance of IWD for undertaking optimization problems.

IWD is a constructive-based meta-heuristic algorithm that iteratively constructs new solutions at every iteration. The process of solution construction is influenced by a probabilistic procedure, i.e., fitness proportionate selection (FPS), which is based on two parameters i.e., soil and velocity. They are updated throughout the solution construction process at every iteration, in order to guide the search process toward the optimal solution

The published results by Shah-Hosseini (2007) indicated that good results could be achieved at the early stage of the IWD optimization process (i.e., the first few iterations). However, all the water drops could stuck at a local solution, and unable to achieve further improvements. This problem is known as search stagnation (Stützle and Dorigo, 1999). It is a common problem in constructive, swarm-based optimization methods, which include IWD (Niu et al., 2012).

Swarm-based optimization methods depend on global optimal solutions found thus far to generate new solutions. While this technique could lead to good solutions, other sub-optimal solutions could also contribute towards generating better solutions. As the swarm-based methods inject a strong selection pressure to the global-optimal solutions found thus far, the search process could converge prematurely at a rapid pace. Conversely, a weak selection pressure could diverse the search to unfavorable regions, resulting in a slow convergence. Therefore, the Darwinian's survival of the fittest principle should be observed to control the balance between diversification and intensification during the search process.

The soil update mechanism and FPS are the main factors affecting the selection pressure in IWD (Niu et al., 2012). After certain number of IWD iterations, it is possible for lower soil levels to be assigned to the components of the local optimal solutions. As such, in successive iterations, the water drops are likely to combine these components in the solution, causing the water drops to be stuck in local optima, therefore unable to escape and explore another region of the search space.

Furthermore, IWD works with single large population of water drops. Many findings in the literature indicate that re-running IWD with different random initialization and using the best solution found among all runs could allow IWD to escape from stagnation (Ahmed and Glasgow, 2012). Splitting the large population into several small sub-populations and running IWD in an asynchronous way could also maintain diversity in a good way (Reimann et al., 2004). In addition, the divide-and-conquer technique can be considered to maintain interaction among the sub-populations.

1.3 Research Objectives

The aim of this research is to develop enhanced IWD algorithms, which can be used to tackle combinatorial optimization problems effectively. The ultimate goal is to show that enhanced IWD algorithms perform better than the original IWD algorithm and other state-of-the-art methods in solving COPs.

The primary objectives of this research are as follows:

- to utilize a suitable selection mechanism in the solution construction phase of the IWD algorithm to enhance its population diversity;
- to modify the IWD algorithm by utilizing the divide-and-conquer and multi-population strategies to empowering its exploration capability;
- to hybridize the modified IWD algorithm with a local based search method to enhance its exploitation capability;
- to assess the usefulness of the enhanced IWD algorithms using benchmark COPs and demonstrating its applicability to real-world problems.

1.4 Research Contributions

In this research, the objectives mentioned in Section 1.3 lead to the following tangible contributions.

- The original IWD algorithm is modified by replacing the fitness proportionate selection method (FPS) in the solution construction phase with two ranking-based selection methods i.e. the exponential and linear ranking selection methods. This proposed modification results in a model called Modified IWD. It is proposed to avoid the search stagnation problem by enhancing population diversity.

- An ensemble model of the Modified IWD algorithm is proposed to improve the exploration capability of the Modified IWD algorithm. The resulting model is known as the Master River Multiple Creeks IWD model, and is denoted as MRMC-IWD.

- The MRMC-IWD model is hybridized with a local search algorithm, which enhances local exploitation of the search space; therefore achieving a balance between exploration and exploitation in the resulting model, which known as hybrid MRMC-IWD.

- The applicability of the hybrid MRMC-IWD model is comprehensively assessed using benchmark and real-world optimization problems. The problems include UCI (University of California Irvine machine learning repository) benchmark data sets (Bache and Lichman, 2013) and two real-world problems, namely human motion detection and motor fault detection.

1.5 Research Methodology

Figure 1.2 depicts a three-stage methodology, which has been employed to achieve the research objectives mentioned in Section 1.3. The first stage modifies the original IWD algorithm to improve its performance for solving COPs. It includes two subsequent steps: (i) modifies the original IWD algorithm by replacing the original selection (i.e., FPS) method in the solution construction phase by two ranking-based selection methods (i.e., linear and exponential ranking); (ii) modifies the fundamental algorithmic aspect (i.e., exploration) of the IWD by proposing an ensemble model of the modified IWD algorithms called MRMC-IWD. In each step, benchmark data sets are used in the experimental study to evaluate the usefulness of the proposed modification. The second stage combines the MRMC-IWD model with a local based method. Again, evaluation is conducted to assess the effectiveness of the proposed models. The last stage assess the applicability of the proposed model (i.e., Hybrid MRMC-IWD) to real-world problems.

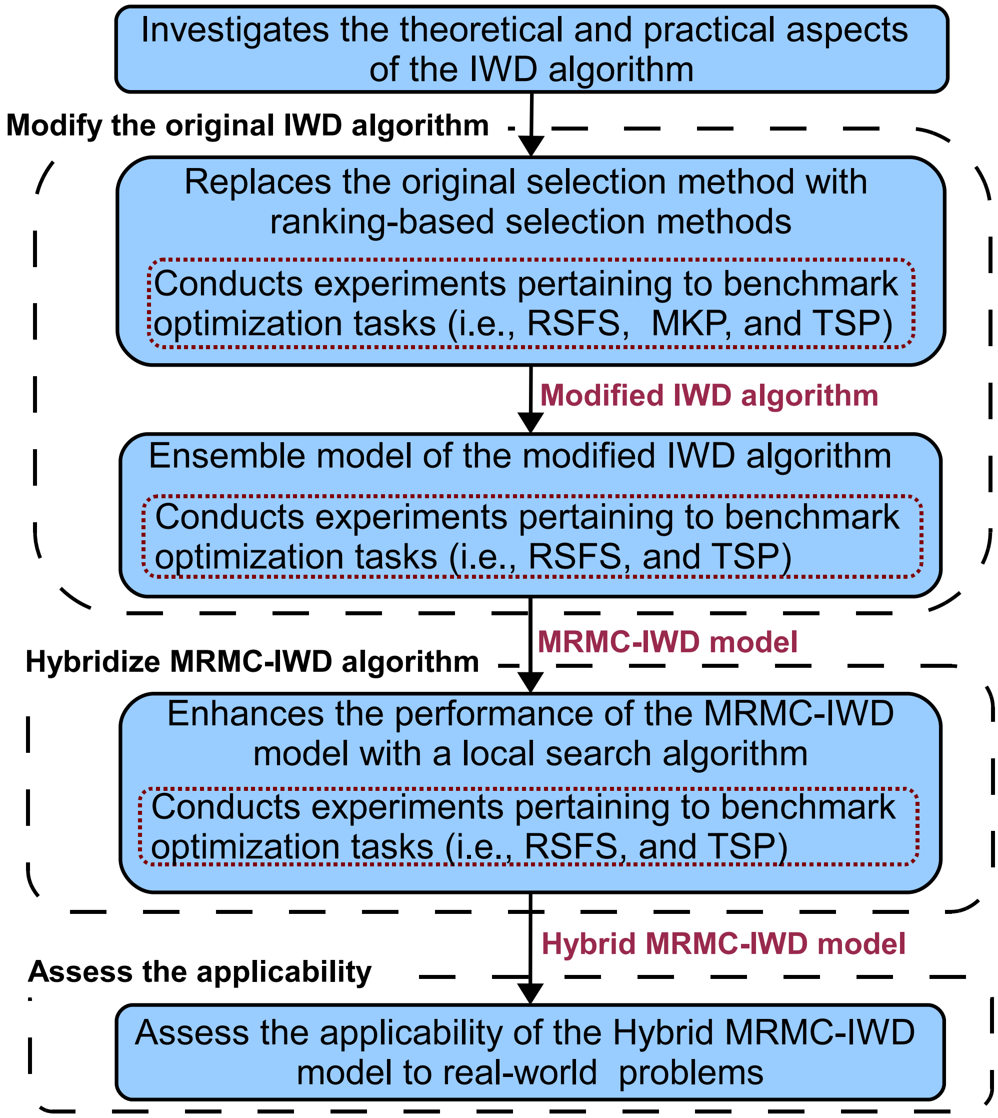

Figure 1.2: The main stages of the research methodology.

To evaluate the proposed models (i.e. MRMC-IWD and hybrid MRMC-IWD) and to facilitate performance comparison with other state-of-the-art methods, three case studies are carried out, namely TSP, MKP, and rough set features subset selection (RSFS) that are widely used in the literature (Chu and Beasley, 1998; Yu and Liu, 2004; Matthias et al., 2007; Shah-Hosseini, 2008; Xie and Liu, 2009; Smith-Miles and Lopes, 2012; Azad et al., 2014). These problems are selected because they are NP-hard, and have different level of difficulties. The problem complexity (i.e., the number of alternatives) grows exponentially with respect to the size of the problem (Helsgaun, 2000). Since TSP and MKP have known bounds, they are useful to ascertain the effectiveness of the solutions produced by the proposed models. On the other hand, RSFS is crucial in pattern recognition applications. Contrary to TSP, RSFS presents strong inter-dependency among the decision variables (i.e., features). The feature sequences within the subset are not important, and the optimal solutions are normally unknown (Yu and Liu, 2004). Contrarily to both TSP and RSFS, the MKP is a constrain based optimization problem (Shah-Hosseini, 2008; Azad et al., 2014). Therefore, TSP, MKP, and RSFS problems are selected as case studies to evaluate the usefulness of the proposed models and to benchmark the results against those published in the literature. As a result, the effectiveness of the proposed models for undertaking general optimization problems can be validated.

1.6 Research Scope

This research focuses on enhancing the performance of the IWD algorithm to tackle COPs. Three main approaches, namely the selection mechanism in the solution construction phase of the IWD algorithm, the ensemble model of the IWD algorithm with a novel problem decomposition technique, and the hybrid IWD model have been investigated to overcome the search stagnation problem; therefore improving original IWD performance. In this context, this research is limited to the use ranking-based selection methods (i.e., linear and exponential), as well as it is limited to *k*-means clustering algorithm to decompose the entire problem into few simple sub-problems. To assess the effectiveness of the proposed models, a series of experiments pertaining to three COPs (i.e., TSP, RSFS, and MKP) with performance comparison against other state-of-the-art methods is conducted. In this regards, this research is limited to the combinatorial single-objective optimization problems. The applicability of the proposed models to two real-world problems related to feature selection and classification task, namely human motion detection and motor fault detection, is investigated.

1.7 Thesis Structure

The rest of this thesis is organized as follows:

Chapter 2 *(Background and Literature Review)*: In this chapter, a detailed descriptions of the IWD algorithm, including learning mechanisms, fundamental steps and mathematical formulation. A review pertaining to optimization and the associated approaches (i.e., selection methods, multi-populations, and hybridization), which are used to enhance the performance of the IWD algorithm in solving COPs is presented. An overview of the evaluation problems (i.e., RSFS, TSP, and MKP) and the associated data sets used in the experiments are also presented.

Chapter 3 *(Modified Intelligent Water Drops Algorithm)*: This chapter introduces the first contribution, i.e., the modified IWD algorithm. The effectiveness of the selection mechanism in the solution construction phase of the IWD algorithm is investigated. Two ranking-based methods are proposed to replace the FPS method in the solutions construction phase of the original IWD algorithm. The experimental results pertaining to benchmark COPs and evaluation of the proposed ranking-based selection methods are presented.

Chapter 4 *(An Ensemble of Intelligent Water Drops Algorithms)*: In this chapter two new contributions are presented. Firstly, the Master-River Multiple-Creek IWD (MRMC-IWD) model is introduced. The proposed model is motivated by a multi-population scheme with the divide-and-conquer strategy to simplify the search process, and to exploit the exploration capability of the modified IWD algorithm. Secondly, the hybrid MRMC-IWD model is proposed by hybridizing MRMC-IWD with a local search method, i.e., Iterative Improvement Local search (IILS). The aim is to empower MRMC-IWD with local exploitation capabilities, therefore achieving a balance between exploration and exploitation. The effectiveness of the proposed models is investigated using a series of experiments pertaining to the benchmark COPs.

Chapter 5 *(Applications of the hybrid MRMC-IWD model)*: The hybrid MRMC-IWD model devised in Chapter 5 is applied to UCI benchmark feature selection and classification problems. Comparative studies against other state-of-the-art methods are presented. In addition, two real-world problems, namely human motion detection as well as motor fault detection are examined, to assess and demonstrate the applicability of the hybrid MRMC-IWD model.

Chapter 6 *(Conclusion)*: Concluding remarks and a summary of the key findings are presented in this chapter. A discussion of future researches that can be carried out to further investigate the enhanced IWD models to handle different optimization problems is presented.

Appendices *(Appendix A)*: A detail description to feature selection methods is presented. It is mainly organized into two part, First part provides a succinct review of categories of feature selection methods. In the second part, a detailed description of rough set and fuzzy rough set for subset feature selection and illustrative example are provided.

CHAPTER 2

BACKGROUND AND LITERATURE REVIEW

2.1 Introduction

The main focus of this research is to enhance the performance of the IWD algorithm to tackle COPs. This chapter is mainly organized into three sections. In section 2.2 a detailed description on the IWD algorithm is presented. Section 2.3 reviews optimization methods and literatures related to the IWD method to solve COPs. Section 2.4 presents an overview of the three COPs (i.e., RSFS, TSP, MKP) as well as the characteristics of the data sets, which are employed in the experimental studies to evaluate and validate the usefulness of the proposed model and to benchmark the results against those published in the literature.

2.2 Background to the Intelligent Water Drops Algorithm

In nature, water in a river follows an easier path with fewer barriers and obstacles. Water flows with a particular speed. Water stream changes the environmental properties of the river, and subsequently changes the direction of water flow to create an optimal path between the upstream and downstream of a river. The IWD algorithm is a constructive-based SI optimization method introduced by Shah-Hosseini (2007). It is inspired by the natural phenomena of water drops moving along the river bed. The IWD algorithm computationally realizes some of the natural phenomena and uses them as a computational mechanism to solve COPs. It comprises a number of computational agents (i.e., water drops). At each iteration, water drops construct a solution based on a finite set of discrete movements. Two key properties of natural water drops are imitated by the IWD algorithm, i.e. velocity and soil, which are changed during a series of transitions pertaining to the movement of water drops. Each water drop iteratively moves step-by-step from one location to the next until a complete solution is produced. It begins with an initial state, i.e. an initial velocity, and carries zero amount of soil.

Water drops cooperate with each other to update the environmental properties, i.e. soil and velocity. Changes in the soil and velocity parameters have an influential role on the selection probability of the flow direction. When a water drop moves from one location to the next, its velocity and soil level are updated. The velocity is changed non-linearly, and is proportional to the inverse of the amount of soil between two locations. Therefore, water drops in a path with less soil move faster. The water drop carries an amount of soil in each movement, which is non-linearly proportional to the inverse of the time needed by the water drop to move from the current location to the next. On the other hand, the time taken by a water drop to move from one location to another is proportional to its velocity and inversely proportional to the distance between two locations.

Figure 2.1 depicts a flowchart of the fundamental of the IWD algorithm, as presented in Algorithm 2.1. In the following sub-sections, a detailed description of the problem formulation as well as the main phases of the IWD algorithm i.e. initialization, solution construction, reinforcement, and termination are presented.

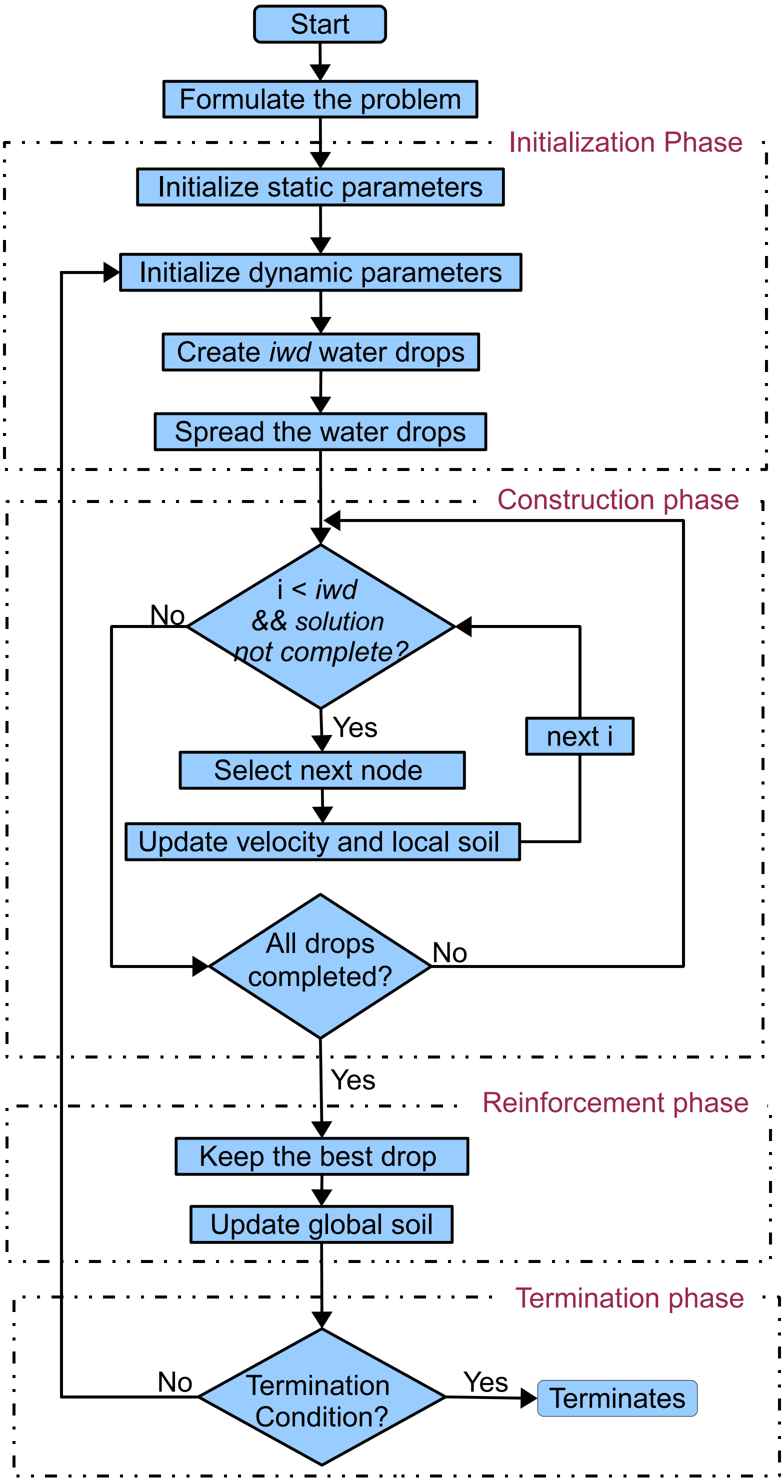

Figure 2.1: A flowchart shows the fundamental of the IWD algorithm (Alijla et al., 2014).

Algorithm 2.1 : The main steps of the IWD algorithm (Alijla et al., 2014).

Input: Data instances.
Output: Subset of features.
Formulate the optimization problem as fully connected graph.
Initialize the static parameters i.e. parameters are not changed during the search process.
while algorithm termination condition is not met **do**
  Initialize the dynamic parameters i.e. parameters changed during the search process.
  Spread *iwd* number of water drops randomly on a construction graph.
  Update the list of visited vertex $(V_{visited})$, to include the source vertex.
  while construction termination condition is not met **do**
    for $k = 1$ to *iwd* **do**
      $i =$ the current vertex for drop k.
      $j =$ selected next vertex, which does not violate problem constrains.
      move drop k from vertex i to vertex j.
      update the following parameters.
      (a). Velocity of the drop k.
      (b). Soil value within the drop k.
      (c). Soil value within the edge e(i,j).
    end for
  end while
  Select the best solution in the iteration population (T^{IB})
  Update the soil value of all edges included in the (T^{IB})
  Update the global best solution (T^{TB})
  if quality of $T^{TB} <$ quality of T^{IB} **then**
    $T^{TB} = T^{IB}$
  end if
end while
return T^{TB}

2.2.1 Problem formulation

As shown in Algorithm 2.1 (line 3), formulating an appropriate problem is a preliminary step for solving any optimization problem using the IWD algorithm. The IWD algorithm uses a fully connected weighted graph called the construction graph i.e. *G(V, E)*, to represent an optimization problem, where $V = \{v_i | i = 1...N\}$ denotes the set of vertices in the graph, $E = \{(i, j) | (i, j) \in V \times V, i \neq j,\ i, j = 1...N\}.$ denotes a set of edges, and N is the total number of decision variables. Consider a solution, $\pi_k = \{a_{kj} | k = 1...iwd,\ j = 1...|D_k|\}$, where k denotes the index of a solution within the population, *iwd* is the total number of solutions in the population (i.e., the number of water drops), and $a_{kj} \in A$ is a set of all possible components of the solution. As an example, solution $\pi_k = \{a_{k1}, a_{k2}, ..., a_{k|D_k|}\}$, where $|D_k| \leq N$ is the dimension of the solution k, which is one of the possible permutations constructed from the possible components of A.

2.2.2 Initialization phase

As show in Algorithm 2.1 (line 4), the initialization phase is used to initialize a set of static and dynamic parameters of the IWD algorithm. Thereafter, the water drops are spread randomly.

2.2.2(a) Static parameters

The static parameters are initialized with static values, and they remain unchanged during the search process. They are:

- ***iwd***: is the number of water drops, which denotes a set of agents that forms the solution population.

- **Velocity updating parameters (a_v, b_v, c_v)**: a set of parameters used to control the velocity update function, as defined in Eq. (2.4)

- **Soil updating parameters (a_s, b_s, c_s)**: a set of parameters used to control the soil update function, as defined in Eq. (2.7)

- ***Max_iter***: the maximum number of iterations before terminating the IWD algorithm.

- ***initSoil***: the initial value of the local soil.

2.2.2(b) Dynamic parameters

The dynamic parameters are initialized before search begins, and are updated during the search process. They are reverted to their initial values at the beginning of each iteration. The dynamic parameters are:

- $\mathbf{V^k_{visited}}$: a list of vertices visited by water drop *k*.

- $\mathbf{intiVel^k}$: the initial velocity of water drop *k*.

- $\boldsymbol{Soil^k}$: the initial soil loaded on water drop *k*.

At the beginning, water drops are spread randomly at the vertices of the construction graph, and $\mathbf{V^k_{visited}}$ is updated to include the initial state (i.e., vertex).

2.2.3 Solution construction phase

The main aim of this phase is to construct a population of *iwd* solutions. A solution comprises a finite set of components, $\pi_k = \{a_{kj} | k = 1...iwd, j = 1...|D_k|\}$, D_k is the dimension of solution k. Water drop k starts with an empty set of solution components, $\pi_k = \{\}$. The first vertex of the tour is added to π_k whenever the water drop is spread. Then, at each step of the construction phase, the water drop extends the partial solution by traversing a new vertex, i.e., a feasible component that does not violate any constraints of the problem. The construction phase is completed by the transition of all water drops through the graph until the stopping criteria for constructing a complete population is met (see Algorithm 2.1, lines 9-16). The construction phase is composed of the following steps:

2.2.3(a) Edge selection mechanism

Consider water drop k residing at the current vertex i intends to move to the next vertex j through an edge, $e(i, j)$, where $e \in E$. The probability of selecting $e(i, j)$ is determined by $p_i^k(j)$, as defined in Eqs. (2.1) and (2.2). Then, the water drop visits vertex j by adding it to $V_{visited}^k$.

$$p_i^k(j) = \frac{f(soil(i,j))}{\sum\limits_{\forall l \notin V_{visited}^k} f(soil(i,l))} \tag{2.1}$$

$$f(soil(i,j)) = \frac{1}{\varepsilon + g(soil(i,j))} \tag{2.2}$$

where ε is a small positive number used to prevent division by zero in function $f(.)$

$$g(soil(i,j)) = \begin{cases} soil(i,j) & if \min_{\forall l \notin V^k_{visited}} soil(i,l) \geqslant 0, \\ soil(i,j) - \min_{\forall l \notin V^k_{visited}} soil(i,l) & Otherwise. \end{cases} \tag{2.3}$$

where $soil(i,l)$ refers to the amount of soil within the local path between vertices i, and j.

2.2.3(b) Velocity and local soil Update

The velocity of water drop k at time $t+1$ is denoted by $vel^k(t+1)$. It is updated every time it moves from vertex i to vertex j using Eq. (2.4).

$$vel^k(t+1) = vel^k(t) + \frac{a_v}{b_v + c_v * soil^2(i,j)} \tag{2.4}$$

where a_v, b_v, and c_v are the static parameters used to represent the non-linear relationship between the velocity of water drop k (i.e., vel^k) and the inverse of the amount of soil in the local path (i.e., $soil(i,j)$). When water drop k moves from vertex i to vertex j, both $soil^k$ (i.e., the soil within water drop k)and $soil(i,j)$ are updated using Eqs. (2.6) and (2.5) respectively.

$$soil^k = soil^k + \Delta soil(i, j) \tag{2.5}$$

$$soil(i,j) = (1 - \rho_n) * soil(i,j) - \rho_n * \Delta soil(i, j) \tag{2.6}$$

where ρ_n is a small positive constant between zero and one, (i.e., $0 < \rho_n < 1$); $\Delta soil(i,j)$ is the amount of soil removed from the local path and carried by the water drop. Note that $\Delta soil(i,j)$ is non-linearly proportional to the inverse of the time needed for a water drop to travel from the current vertex to the next, as defined in Eq. (2.7).

$$\Delta soil(i, j) = \frac{a_s}{b_s + c_s * time^2(i,j : vel^k(t+1))} \tag{2.7}$$

where a_s, b_s, and c_s are the static parameters used to represent the non-linear relationship between $\Delta soil(i,j)$ and the inverse of the time. Note that $time(i,j : vel^k(t+1))$ refers to the time needed for water drop k to transit from vertex i to vertex j at time $t+1$. It is proportional to the distance between the two vertices as well as is proportional to the inverse of the $vel^k(t+1)$ as is defined as shown in Eq. (2.8).

$$time(i,j : vel^k(t+1)) = \frac{HUD(i, j)}{vel^k(t+1)} \tag{2.8}$$

where $HUD(i,j)$ refers to a heuristic desirability degree between vertices i and j.

The processes of selecting a vertex to visit as well as updating the velocity and local soil are iterated subject to the stopping criteria for obtaining a complete solution.

2.2.4 Reinforcement phase

As shown in Algorithm 2.1 (lines 17-21), the fittest solution of each population is known as the iteration-best solution, and is denoted as T^{IB}. It is determined using Eq. (2.9).

$$T^{TB} = arg \min_{\forall l \in T^{IWB}} q(x) \tag{2.9}$$

where $q(.)$ is the fitness function which is used to evaluate the quality of the solutions, and T^{IWB} is the population of the solutions. To reinforce the water drops in the subsequent iterations to follow T^{IB} and achieve the fittest solution over the iterations, the soil of all edges in T^{IB} is updated using Eq. (2.10). This update process is known as global soil update.

$$soil(i,\ j) = (1+\rho_{iwd}) * soil(i,j) - \rho_{iwd} * \frac{1}{q(T^{IB})} \tag{2.10}$$

where ρ_{iwd} is a small positive constant between zero and one, (i.e., $0 < \rho_{iwd} < 1$).

In each iteration, the best solution (global best), (i.e.,T^{TB}), is either replaced by T^{IB} or maintained, as defined in Eq. (2.11).

$$T^{TB} = \begin{cases} T^{IB} & if q(T^{IB}) < q(T^{TB}) \\ T^{TB} & Otherwise. \end{cases} \tag{2.11}$$