Computational Intelligence Modeling of Pharmaceutical Roll Compaction

Hossam Mohammed Zawbaa Ismail ELSAYED

Faculty of Mathematics and Computer Science

Babes-Bolyai University

PhD thesis summary

Doctor of Philosophy (PhD)

May 2016

Contents

Abstract

In this study, I have developed new variants of bio-inspired optimization algorithms such as binary antlion optimization (BALO), binary grey wolf optimization (BGWO), and much more. All the proposed algorithms are compared to two well-known techniques used in feature selection, namely particle swarm optimization (PSO) and genetic algorithms (GA). With the big data captured in the pharmaceutical product development practice, computational intelligence (CI) models, based on machine learning and bio-inspired optimization algorithms, could potentially be used to identify critical quality attributes (CQA) and critical process parameters (CPP), for the formulations and manufacturing processes. The primary objective is to evaluate the robustness of machine learning techniques combined with bio-inspired optimization algorithms in modeling tablet manufacturing processes. More precisely, our effort is focused on the prediction of tablet properties such as porosity and tensile strength from powder and ribbons characteristics. For this purpose, roll compaction experiments were performed with various pharmaceutical excipients, leading to datasets with a wide range of features. The modeling efficiency is evaluated regarding the selected features and the root mean square error. We have remarked that the predicted results were in good agreement with the actual experimental data.

Keywords:
Bio-inspired optimization, Feature selection, Pharmaceutical roll compaction, Antlion optimization, Moth-flame optimization, Grey wolf optimization, Social spider optimization, Flower pollination algorithm, Genetic algorithm, Particle swarm optimization.

Chapter 1

Introduction

An input *feature* is a measurable property of the problem under observation. Over the past years, the domain of features in machine learning and pattern recognition applications have expanded from hundreds to thousands of features (variables). The large amounts of data generated today in biology offer more detailed and useful information on one hand; on the other hand, it makes the process of analyzing these data more difficult because not all the information is relevant. Selecting the relevant characteristics or attributes of the data is a complex problem. *Feature selection* is a technique for solving classification and regression problems, and it identifies a subset of the features and removes the redundant ones. This mechanism is particularly useful when the number of features is large, and not all of them are required for describing the data and for further exploring the data features in experiments (1).

Many studies formulate the feature selection problem as a *combinatorial optimization* problem, in which the selected feature subset leads to the best data fitting (2). In real world applications, feature selection is mandatory due to the abundance of noisy, irrelevant or misleading features (3). These factors can have an adverse impact on the classification performance during the learning and operation processes. Two main criteria are used to differentiate the feature selection methods:

1. *Search strategy*: the method employed to generate feature subsets or feature combinations.

2. *Subset quality (fitness)*: the criteria used to judge the quality of a feature subset.

In general, the feature selection problem is formulated as a *multi-objective* problem with two objectives: *minimize* the size of the selected features and *maximize* the prediction performance. Typically, these two objectives are contradictory, and the optimal solution is a tradeoff between them (4).

Most of the new optimization algorithms are *nature-inspired* that have been inspired from nature (5). There are three main sources of inspiration, namely *biology*, *physics*, and *chemistry*. Therefore, all the new optimization algorithms based on *biology* can be referred to as *bio-inspired* (5).

In general, the size of search space is exponentially increasing with respect to the number of features of a given data set (6). Thence, an exhaustive search for the optimal or near to optimal solution in an enormous search space may be *impracticable* and these exhaustive search techniques still suffer from *stagnation* in local optima (7) (2). It is essential to have a convenient balance between *exploration* (diversification, global search) and *exploitation* (intensification, local search) in all bio-inspired optimization algorithms (8).

The specific objectives are derived from the general ones. In our research, they are given below:

- O_1: utilization of the bio-inspired optimization algorithms for feature selection to solve the classification problem that minimizes the number of selected features and maximize the classification accuracy.

- O_2: in regression problem, the use of bio-inspired optimization to reduce the number of selected features and minimize the prediction error.

- O_3: applying the bio-inspired optimization in the pharmaceutical domain to minimize the number of selected features and minimize the prediction error. Also, we also highlighted the importance of each input variables for a given dataset.

In the pharmaceutical industry, a good understanding of the casual relationship between product quality and attributes of formulations is very useful in developing new products and optimizing the manufacturing processes. During the pharmaceutical drug production, there are four main manufacturing processes namely mixing, roll compaction, milling, and die compaction. Roller compaction is a method of preparing drug granules for capsules or tablet formulations used in the pharmaceutical industry with suitable densification. The most common filler binder excipient used in roller compaction are microcrystalline cellulose (MCC), dibasic calcium phosphate (DCP), and lactose. Also, roller compaction is a particle size enlargement technique that granulated the powder materials to obtain materials of intermediate sizes in tablets production. The use of latest technology facilitates to efficient production of high-quality granules. The selection of the critical roll compaction parameters such as (constant compacting pressure, roller gap, etc.) is a very important process.

The thesis consists of nine chapters including this introductory one. The introduction explores the characteristics of the bio-inspired optimization, the importance of feature selection in classification and regression problems, as well as the impact of feature selection in pharmaceutical domain. Chapter (2) surveys the existing related work on machine learning, bio-inspired optimization, and its applications. Chapter (3) provides background information about the bio-inspired optimization and machine learning algorithms. Chapter (4) presents the different random model generators, the proposed bio-inspired optimization algorithms and performance evaluation metrics. The next three chapters contain our original work: experimental work of using bio-inspired optimization algorithms in classification (Chapter (5)); experimental results of using bio-inspired optimization algorithms in regression (Chapter (6)), and experimental results of bio-inspired optimization algorithms in the pharmaceutical domain (Chapter (7)). Moreover, Chapter (8) demonstrate three different bio-inspired algorithms case studies. All those four chapters discuss the characteristics of datasets used and results analysis by comparison to well-known methods in feature selection. Finally, chapter (9) summarizes the conclusions of this study and directions for future work.

Chapter 2

Related work

2.1 Machine learning

Machine learning (ML) techniques play an important role to solve many complex classification and regression problems. ML techniques are used in classification and regression for constructing the prediction models from of given data. Neural networks, especially single hidden layer feed-forward neural network (SLFN) is considered one of the most common machine learning models used in regression and classification domains (9). The extreme learning machine (ELM) model has been proposed for single hidden layer feed-forward neural networks (SLFNs). In ELM model, the connections between the input layer and the hidden neurons are randomly selected and remain unchanged during the learning process (10).

2.2 Bio-inspired optimization

Various heuristic techniques mimic the behavior of biological and physical systems in nature, and it has been proposed as robust methods for global optimizations. GA was the first evolutionary based algorithm introduced in the literature and developed based on the natural process of evolution through reproduction (11). In PSO, each solution is considered as a particle that is defined by position, fitness, and a speed vector which represents the moving direction of the particle (12). Ant colony optimization (ACO) based wrapper feature selection algorithm applied in network intrusion detection (13). ACO uses Fisher discrimination rate to adopt the heuristic information and rough set theory used for feature selection method with ACO (14). Artificial bee colony (ABC) is numerical optimization algorithm based on the foraging behavior of honeybees. In ABC, the employer bees try to find the food source and advertise the other bees (15). The interactions between these bees achieve the possible solution for the optimization problem (16). Antlion optimization algorithm (ALO) is a comparatively recent EC algorithm that mimics the hunting mechanism of antlions in nature (17).

Chapter 3

Preliminaries and background

3.1 Machine learning

Classification techniques are designed for dependent variables that take a finite number of unordered values, with prediction error measured regarding misclassification cost. Regression techniques are for dependent variables that take continuous or ordered discrete values, with prediction error (mean square error) typically measured by the squared difference between the observed and predicted values. This section gives an introduction to classification and regressions techniques that used during the experiments.

3.1.1 Classification

Machine learning (ML) methods play an important role to solve complex classification problems in different applications. In this subsection, we present an overview of the classification techniques.

3.1.1.1 K-nearest neighbor (KNN)

K-nearest neighbor (KNN) is a very simple classifier based on the nearest neighbor approach. In the classification phase, a new sample is classified based on majority of K-nearest neighbor category (K is predefined integer), given a query point, the algorithm finds K number of objects or training points closest to the query point. Simply it works based on minimum distance from the searching query to the training one to determine the K-nearest neighbors. After we specified the K-nearest neighbor classes, the new sample follows (predicts) as the major class of KNN (18).

3.1.1.2 Random forest (RF)

Random forests (RF) is considered as one of the best machine learning classification and regression techniques. It can classify large data set with high accuracy (19). It consists of a collection of tree-structured classifiers. Each tree depends on the random vector values sampled independently and distribution for all trees in the forest (20). Its input goes into the top of the tree, then traverses down the tree. The original data is randomly sampled, but with the replacement of smaller and smaller sets. The sample class is determined using random forests trees that are based on random number generator (19). The randomizing variable specifies how the cuts are performed successively when constructing the tree by selecting the node and the coordinate to divide and the position of the divided (21).

3.1.2 Regression

3.1.2.1 Artificial neural network (ANN)

Artificial neural networks have been developed as generalizations of mathematical models of biological nervous systems. In a simplified mathematical model of the neuron, the effects of the synapses are represented by connection weights that modulate the effect of the associated input signals, and the nonlinear characteristic exhibited by neurons is represented by a transfer function. There are many transfer functions developed to process the weighted and biased inputs, among which four basic transfer functions widely adopted for multimedia processing. The behavior of the neural network depends largely on the interaction between the different neurons (22).

3.1.2.2 Extreme learning machine (ELM)

The extreme learning machine (ELM) model has been proposed for single hidden layer feed-forward neural networks (SLFNs). In ELM model, the connections between the input layer and the hidden neurons are randomly selected and remain unchanged during the learning process. The output connections are then tuned via minimizing the cost function through a linear system (10). The main objectives of ELM algorithm are the randomly chooses of the SLFNN hidden layer weights and its biases function. ELM both operations (training and prediction) are much faster than the other non-linear techniques. Therefore, extreme learning machine (ELM) algorithm tends to provide a good generalization performance with high learning speed (23).

3.2 Bio-inspired optimization algorithms

3.2.1 Genetic algorithm (GA)

GA were the first evolutionary-based algorithm that is introduced in the literature and has been developed based on the natural process of evolution through reproduction. Holland developed GA during the 1960s and 1970s. GA can solve the complex and non-linear problems (11).

3.2.2 Particle swarm optimization (PSO)

PSO is a heuristic global optimization method originally developed by Kennedy and Eberhart in 1995 (12). PSO is one of the well-known swarm intelligence algorithms that based on the movement behavior of birds (24). PSO is widely used to solve the optimization and feature selection problems (25).

3.2.3 Artificial bee colony (ABC)

The optimization algorithm based on the foraging behavior of honey bees called ABC was proposed by Karaboga in 2007 (15). In ABC, the employer bees try to find a food source and advertise them. The onlooker bees follow their interesting employer and the scout bee fly spontaneously to find better food sources (8).

3.2.4 Firefly algorithm (FFA)

FFA is a biologically stochastic global optimization method that was developed by Yang in 2008 (26). FFA algorithm imitates the mechanism of firefly mating and exchange of information using light flashes. In FFA, the move of firefly is determined mainly by the attractiveness of the other fireflies.

3.2.5 Cuckoo search (CS)

CS is a heuristic search algorithm that has been proposed by Yang in 2009 (27) for solving the continuous optimization problems. Cuckoo birds have aggressive reproduction strategy and lay their eggs in the nests of other host birds that may be of different species. The host bird may discover that the eggs are not their own, they will either throw these alien eggs away or simply abandon its nest and build a new nest elsewhere.

3.2.6 Bat algorithm (BA)

Yang developed BA algorithm in 2010 (28). BA is a meta-heuristic technique that uses the echolocation behavior for seeking the prey and detects or avoids the obstacles. The bats emit a very loud sound pulse and listen for the echo that bounces back from the surrounding objects for navigation.

3.2.7 Flower pollination algorithm (FPA)

FPA is a meta-heuristic optimization algorithm based on the pollination operation of flowering plants that developed by Yang in 2012 (29). The main objective of a flower is ultimately reproduction via pollination. Flower pollination is typically associated with the transfer of pollen, and this process is linked with pollinators such as (insects, birds, bats, etc.) (30). Pollination can be carried out in two ways *self-pollination (abiotic)* or *cross-pollination (biotic)*; local and global search in the artificial algorithm.

3.2.8 Social spider optimization (SSO)

SSO is one of the recent swarm optimization algorithms that was proposed by Cuevas in 2013 (31). SSO algorithm mimics the social behavior of the spider's colony in nature. The social spider optimizer consists of two main components, social members, and communal web. The spider carries out cooperative interaction with the other members of the colony (32).

3.2.9 Grey wolf optimization (GWO)

GWO algorithm was developed by Mirjalili in 2014. GWO is relatively new bio-inspired heuristic optimization algorithm that imitates the way wolves search for food and survive by avoiding their enemies (33). Each grey wolf in the pack chooses its own position, continuously moving to a better spot and watching for potential threats. GWO is prepared with a threat probability, which mimics the incidents of wolves bumping into their enemies.

3.2.10 Hybrid monkey algorithm with krill herd algorithm (MAKHA)

MAKHA algorithm was developed by Khalil in 2015 (34). Hybrid optimization algorithms in which operators from one algorithm are combined with other operators from another one that aim to use the best from each algorithm to produce a better performance. Hybrid MAKHA algorithm uses the best performance operators from monkey algorithm (MA) and krill herd algorithm (KHA) that omit the less performance and high-demanded calculation operators from both algorithms (34).

3.2.11 Dragonfly algorithm (DA)

Mirjalili recently proposed DA in 2015 (35). Dragonflies are amazing insects, and there are about 3000 different species of this fancy insect (36). The life cycle of a dragonfly consists of two main phases: nymph and adult. In the first phase, dragonflies spend the significant portion of their lifespan. Then, they undergo to the second stage to become adult (36). Dragonflies look like small predators that hunt almost all other small insects in nature. DA mimics the static (hunting) and the dynamic (migration) swarming behaviors of dragonflies in nature (37).

3.2.12 Moth-flame optimization (MFO)

Mirjalili developed MFO algorithm in 2015. Moths have been evolved to fly in the night using the moonlight, and they rely on a method called transverse orientation for navigation. In this method, a moth flies by maintaining a fixed angle with respect to the moon (38) (i.e. the light source). This method is considered a very effective technique for traveling long distances in a straight path (39).

3.2.13 Antlion optimization (ALO)

ALO is a bio-inspired optimization algorithm proposed by Mirjalili in 2015 (17). The ALO algorithm mimics the hunting mechanism of antlions in nature. Antlions (doodlebugs) belong to the Myrmeleontidae family and Neuroptera order. They primarily hunt in the larvae stage, and the adulthood period is for reproduction (17).

Chapter 4

The Proposed System and Methodology

All swarm intelligence algorithms usually share information among the multiple agents. Therefore, at each iteration of the optimization all/some agents update/change their position based on position information of other/own position.

Exploration or *global search* can be defined as acquisition of new information throughout searching (26). Exploration is the main concern for all optimizers as it allows for finding new search regions that may contain better solutions. *Exploitation* or *local search* is defined as application of known information. The good sites are exploited via the application by applying local search. The selection process should be balanced between random selection and greedy selection to bias the search towards fitter candidate solutions (exploitation) while promoting useful diversity into the population (exploration) (26).

The proposed bio-inspired optimization algorithms are used to find the *minimum* number of features that *maximize* the prediction performance. The search space represents each feature as an individual dimension, and the span of each dimension ranges from 0 to 1 and hence requires an intelligent searching method to find the optimal feature set in the huge search space that maximizes a given fitness function. In classification case, the general fitness function for the proposed optimization algorithms is to maximize the classification performance over the validation set given the training set, as shown in equation (4.1) while keeping minimum number of features selected:

$$\downarrow Fitness = \alpha(1-P) + \beta\frac{\mid R \mid}{\mid C \mid}, \tag{4.1}$$

where R is the length of selected feature subset, C is the total number of features in the data set, α and β are two parameters corresponding to the importance of classification performance and subset length, $\alpha \in [0,1]$ and $\beta = 1-\alpha$, P is the classification performance measured as in equation (4.2):

$$P = \frac{N_c}{N}, \tag{4.2}$$

where N_c is the number of correctly classified data instances and N is the total number of instances in the data set.

In the case of regression, the general fitness function for the proposed optimization algorithms is to minimize the prediction error over the validation set given the training set as in equation (4.3) while keeping a minimum number of features selected.

$$\downarrow Fitness = \alpha * E + \beta\frac{\sum_i \theta_i}{N}, \tag{4.3}$$

where E is the prediction error, θ represents a vector sized N with 0/1 elements representing unselected/selected features, N is the total number of features in the dataset.

The used features are the same as the number of features in a given data set. All features are limited in the range [0, 1], where the feature value approaches to 1; its corresponding feature is candidate to be selected in the predicition. In individual fitness calculation, the feature is threshold to decide whether a feature will be selected at the evaluation stage.

The random weighting term α is used with a respectively high value so it can accommodate for the feature space with many local minima. This term is used to balance the trade-off between exploration and exploitation and hence should be carefully adapted.

Through the training process, every agent (moth, antlion, grey wolf, ant, bee, etc.) position represents one feature subset. The training set is used to evaluate the classification (KNN) and regression models (ELM or ANN) on the validation set during the optimization to guide the feature selection process. Each data set is divided into three equal parts for *training*, *validation*, and *testing*. The *training* set is used to train the prediction model through optimization and at the final evaluation. The *validation* set is used to assess the performance of the prediction model during the optimization process. The *testing* set is used to evaluate the selected features at the final evaluation. The classification and regression models (KNN or ELM or ANN) is used to ensure the quality of the selected features and is evaluated on a validation set inside the fitness function during the optimization process (24).

4.1 Random distributions

In this section, we talk about the different random models (Gaussian distribution, lèvy flight, and chaotic distribution) and how it's applied with different bio-inspired optimization algorithms. A *random variable* can be considered as an expression whose value is the realization or outcome of events associated with a random process (26). A random variable is a function which maps events to real numbers. The domain of this mapping is called the sample space. Each random variable is represented by a probability density function to express its probability distribution. In this study, we used three different distribution models; the details about each type as shown in the next subsections (40).

4.1.1 Gaussian distribution

Gaussian (normal) distribution is the most popular distributions and many physical variables including light intensity, errors/uncertainty in measurements, and many other processes apply the Gaussian distribution.

4.1.2 Lèvy flight distribution

Many researchers have studied the birds and insects flying behavior that described the typical characteristics of lèvy flights (41). Therefore, lèvy flight distribution has been applied to the optimization problems, and the preliminary results show its promising capability (42).

4.1.3 Chaotic distribution

Chaos means a condition or place of great disorder or confusion (43). Chaotic systems are deterministic systems that exhibit irregular (or even random) behavior and a sensitive dependence on the initial conditions. Chaos is one of the most popular phenomena that exist in nonlinear systems, whose action is complex and similar to that of randomness (44). Chaos theory studies the behavior of systems that follow deterministic laws but appear random and unpredictable, i.e., dynamical systems. Chaotic variables can go through all states in certain ranges according to their own regularity without repetition (44). A *chaotic map* is a map that exhibits some type of chaotic behavior (43). In this work, we applied three different chaotic maps that are common in the literature namely: logistic map, Tent map, and Singer map.

4.2 Applying bio-inspired optimization with different random distributions

4.2.1 Chaotic version of bio-inspired optimization algorithms

Chaotic systems with their interesting properties such as topologically mixing and dense periodic orbits, ergodicity and intrinsic stochasticity, can be used to adapt this parameter and allowing for the required mix between exploration and exploitation. In feature selection, chaos search is more capable of escaping from local optima than random search.

4.2.1.1 Chaotic antlion optimization (CALO)

Iteratively, the antlion algorithm selects an ant for hunting in a roulette wheel manner and performs two different random walks, (a) random walk of ants around the selected ant and (b) random walk around the elite/best antlion. From the previous two random walks, the selected ant adapts its location. Parameter I controls the trade-off between exploration and exploitation in the antlion optimization (ALO) algorithm. I is linearly decreased to allow more exploration at the beginning of the optimization process, while exploitation becomes more important at the end of the optimization. Therefore, half of the optimization resources are consumed in exploration, whereas the remaining time is dedicated to exploitation. Chaotic systems with their interesting properties such as topologically mixing and dense periodic orbits, ergodicity and intrinsic stochasticity, can be used to adapt this parameter and allowing for the required mix between exploration and exploitation. The proposed CALO algorithm is schematically presented in figure (4.1). The search strategy of the wrapper-based approach explores the feature space

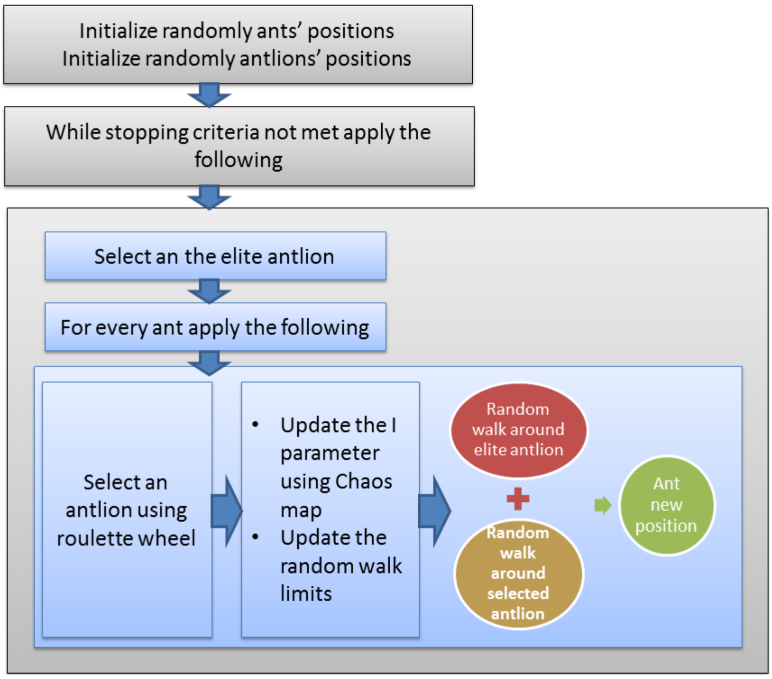

Figure 4.1: The proposed chaotic antlion optimization (CALO)

to find a feature subset guided by the classification performance of individual feature subsets.

4.2.1.2 Chaotic Grey Wolf Optimization (CGWO)

A single parameter, namely $\vec{a}$, was originally proposed to control the trade-off between exploration and exploitation in grey wolf optimization. In the original GWO (33), this parameter was proposed to be linearly decremented to allow for much exploration at the beginning of the optimization while exploitation becomes much more important at the end of optimization. Therefore, that proposal allows us for consuming half of the optimization in exploration while the rest of time is dedicated to exploitation. These problems motivate adapting $\vec{a}$ such as successive periods of exploration are followed by exploitation to allow for exploration distribution all over the optimization times and also to allow for exploitation operations after each exploration stage.

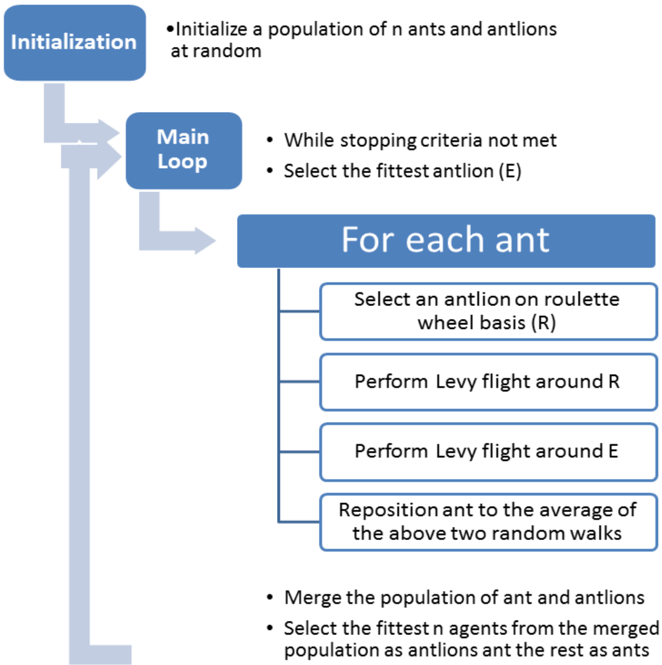

Figure 4.2: The proposed lèvy antlion optimization (LALO) algorithm

4.2.2 Lèvy version of bio-inspired optimization algorithms

Lèvy flight is efficient random walks in exploring the huge search space that can be highlighted from the large abrupt jumps in the walk. A random walking based on lèvy flight is used rather than uniform random distribution to help enhance the convergence speed and convergence to global optima.

4.2.2.1 Lèvy antlion optimization (LALO)

The lèvy antlion optimization (LALO) is comprised of fundamental building phases as described in figure (4.2). Randomization plays an important role in both *exploration* and *exploitation*, the essence of such randomization is the random walk (26). A random walk is a random process that consists of taking a series of consecutive random steps.

4.2.2.2 Lèvy social spider optimization (LSSO)

When the step length obeys lèvy distribution, such a random walk is called a lèvy flight or lèvy walk. Mathematically speaking, a simple version of lèvy flights are more efficient than Brownian random walks in exploring unknown, search large-scale space which can be remarked from the large abrupt jumps in the walk.

4.3 Binary version of bio-inspired optimization algorithms

In some particular problems such as feature selection, the solutions are restricted to the binary {0, 1} values that motivate to propose a binary version of the bio-inspired optimization algorithms. The use of a binary version of an algorithm much limits the search space size and hence eases the task of finding the optimal solutions.

4.3.1 Binary Grey Wolf Optimization (BGWO)

In the continuous grey wolf optimization (GWO) wolves continuously change their positions to whatever point in the space. In some special problems such as feature selection, the solutions are restricted to the binary {0, 1} values which motivate a special version of the CGWO. In this work, a binary version of grey wolf optimization (bGWO) is proposed for the feature selection task (45). The wolves updating equation is a function of three position vectors namely $x_\alpha, x_\beta, x_\delta$ which attracts each wolf towards the first three best solutions. In BGWO algorithm, the pool of solutions is in binary form at any given time; all solutions are on the corner of a hypercube. To update the positions of a given wolf according to the CGWO principle, while keeping the binary restriction.

4.3.2 Binary Antlion Optimization (BALO)

ALO has very competitive results in terms of *improved exploration*, *local optima avoidance*, *exploitation*, and *convergence* (17). These attractive properties motivate using ALO in other applications. Continuous optimization algorithms are commonly used to find feature combinations that maximizing the classifier performance where search agents are positioned in a d-dimensional search space at positions [0, 1]. Binary optimization algorithms; if appropriately used in a similar manner, but the search space is much limited as two values are only allowed for each dimension {0, 1} and hence is expected to perform better. Furthermore, the binary operator is expected to be much simpler than continuous counterparts. In antlion optimization (ALO), antlions and ants continuously change their positions to *whatever* point in the space. Also, the individual ant updates its position as the average of two positions. One of the two positions is obtained by performing random walk with suitable step size around the elite antlion while the other position is obtained by performing random walk around a selected antlion.

4.3.3 Binary moth-flame optimization (BMFO)

In MFO, moths continuously change their positions to *whatever* point in the space depending on the spiral moving (SM). SM is the main component of the algorithm because it decides how the moths are repositioned around flames, which allows the moth to fly around it corresponding flame and not necessarily in the space between them.

4.4 Algorithms used for comparison

In this study, we covered a set of modern numerical swarm-based optimization algorithms. Our comparisons include (ALO, MFO, GWO, SSO, DA, MAKHA, FPA, BAT, CS, and FFA) algorithms. All the optimization algorithms were compared to two common optimization algorithms used in feature selection domain namely PSO and GA, as well as the variants of each of the proposed algorithms.

4.5 Initialization methods

In the proposed feature selection tool, there are four different initialization methods are used to ensure the capability to converge from different initial positions namely *small initialization*, *uniform initialization*, *large initialization*, and *MRMR initialization* (in this study, we only used the uniform initialization). The *small initialization* method is expected to test the global searching capability of a given optimizer as the initial search agents' positions are commonly apart from the optimal. The *large initialization* method is expected to assess the local searching capability of a given optimizer as the search agents' positions are commonly at the optimal solution surround and hence just local searching is usually required to reach the optimal solution. The *uniform initialization* models the common initialization method where search agents are set randomly in the search space using uniform distribution in each dimension. The *MRMR initialization* is a filter-based method that exploits data regardless of the used classifier to select a feature subset.

4.6 Performance metrics

Each algorithm has been applied $K * M$ times with random positioning of the search agents except for the full features selected solution that was forced to be a position for one of the search agents. Forcing the full features solution guarantees that all subsequent feature subsets if selected as the global best solution, are fitter than it. Repeated runs of the optimization algorithms were used to test their convergence capability. The indicators (measures) used to compare the different algorithms such as mean, best, worst, std of fitness values, prediction error, and the selected features.

Chapter 5

Bio-inspired optimization in pharmaceutical processes

In this chapter, we have applied bio-inspired algorithms for different pharmaceutical processes. Extreme learning machine (ELM) and artificial neural network (ANN) are used as regression model fitness function as in equation (4.3). Our third objectives O_3 in the specific research objectives (Chapter 1) has been achieved. Extreme learning machine (ELM) and artificial neural network (ANN) are used as regression model fitness function during the optimization process and evaluating the test set. There are three pharmaceutical data sets (roll compaction, die compaction, and poly-lactic-co-glycolic acid (PLGA)). The primary objective of this chapter is to propose bio-inspired optimization algorithms for feature selection approach that minimize the number of selected features and minimize the prediction error from using all features and conventional feature selection techniques in pharmaceutical domain. Also, we have highlighted the selected features and the corresponding importance of each feature in the prediction model. This chapter summarizes our published paper as in (46), (47), and (48).

Data sets.
We have used three data sets for different pharmaceutical processes, namely, roll compaction, die compaction, and poly-lactic-co-glycolic acid (PLGA). In addition, we use another pharmaceutical data poly-lactic-co-glycolic acid (PLGA) that have a significant effect in much pharmaceutical application.

5.1 Pharmaceutical analysis and discussion

In pharmaceutical data category, the proposed feature selection for computational intelligence (CI) modeling ANN is used as a regression model to evaluate the prediction performance of each algorithm. The aggregate aim of this section is to propose bio-inspired optimization algorithms for feature selection approach that minimize the number of selected features and reduce the prediction error from using all features and conventional feature selection techniques in pharmaceutical domain. Also, we have highlighted the chosen features and the corresponding importance of each feature in the prediction model.

5.1.1 Roll compaction results

In this experiment, we used all the inputs (7) to predict 4 outputs (Granules X10, Granules X50, Granules X90, and Fines). We have applied six bio-inspired optimization algorithms for feature selection combined with extreme learning machine (ELM) for regression (to predict the 4 outputs). We observe that the most significant input features are Proportion MCC, Proportion MCC quadratic, Specific Compaction Force (kN/cm), and Gap Width (mm); as shown in figure (5.1). Moreover, we can conclude that GWO is the best optimization algorithm in this experiment for two outputs (prediction of Percentiles for Granules X50 and fines outputs that make a compromise between RMSE and features reduction.

5.1.2 Die compaction results analysis

The bio-inspired optimization algorithms are used for feature selection, in order to feed good quality data to the ANN. ANN is then used for the prediction of the continuous two outputs (porosity and tensile strength). The results presented in figure (5.2) shows the MSE values for each optimizer for 20 different runs while Figures (5.3) shows the average feature reduction of the two outputs. GWO algorithm was the most accurate in predicting porosity. The most accurate prediction of the tensile strength was achieved by SSO algorithm. GA obtained the highest reduction of features - 60% - with average MSE of 7.2 for predicting porosity and 5.1 for predicting tensile strength. Overall, GWO algorithm obtains the best compromise between MSE and feature reduction for

(a) BALO predictor (b) ALO predictor (c) GWO predictor

(d) CS predictor (e) GA predictor (f) PSO predictor

Figure 5.1: Example of feature importance of percentiles for granules X10 output

predicting both the porosity and the tensile strength. The most importance inputs are "compaction pressure" followed by "material" and "Granule size upper limit" (46).

5.1.3 Poly-lactic-co-glycolic acid (PLGA)

For consistency in comparison of the results for both approaches, original data was extracted from publication and the structure of the data was retained as in Szlęk et al. (47). In brief, the data was gathered after reviewing about 200 scientific publications. The extracted data consisted of release rates of 68 PLGA formulations from 24 publications. Originally, the input vector consisted of 320 variables (molecular descriptors of protein, excipients, formulation characteristics and the experimental conditions) and 745-time points (records). All data were of numeric format with continuous values, except variables such as "Production method" and "Lactide to glycolide ratio" which take discrete values (1, 2, 3, etc.). The amount of the drug substance released (Q) was the only dependent variable, its values ranging from 0 to 100%.

Selection of crucial variables was performed under the assumption that both pre-

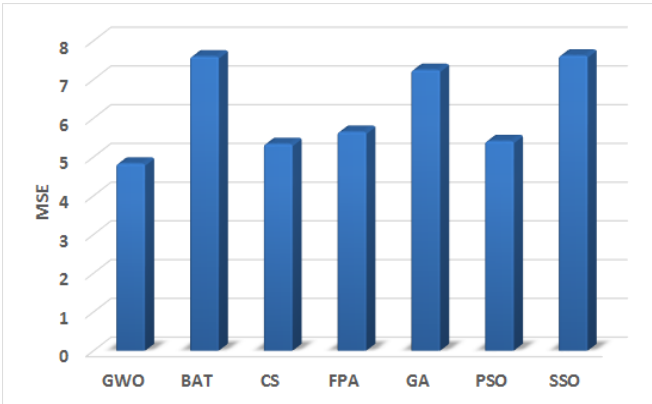

Figure 5.2: Average MSE for porosity output

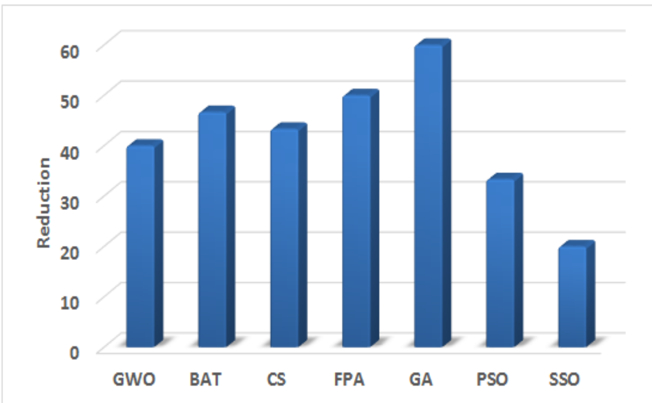

Figure 5.3: Average feature reduction for porosity and tensile strength outputs

dictability and model simplicity are of equal importance to the final result. Four bio-inspired algorithms – ALO, BALO, GWO and SSO – are first used as a feature selection tool. Whenever the feature selection tool proposes a set of input variables, a screening procedure is then employed to find minimum generalization error across different predictive models and their settings/architectures. For predictive modeling, various tools are chosen such as Cubist, RF, ANN (monotonic MLP, deep learning MLP), and FugeR.

5.1.3.1 PLGA results and discussions

The measures described in the following sections are used to quantify the quality of the results obtained by the computational models. RMSE is utilized by both the feature selection and the predictive models and is a measure of how accurate the data is classified. The reduction size is used by the feature selection methods and is an indicator of the dimensionality reduction of the set of all PLGA attributes. Overall, regarding all methods and 10-cv sets, nearly 18,000 models were trained and tested. NRMSE varied from 31.1 to 15.9%. Random Forest algorithm yielded the lowest error; therefore, it was used for selecting optimal inputs vector as in Table (5.1). RF model developed on the nine input data set, 9in(2), selected by BALO algorithm, yielded one of the lowest NRMSE and inputs number as shown in Table (5.2). Comparable results were obtained, 15.97% versus 15.4%, to those by Szlęk et al. (47), but the vector of inputs was smaller, nine versus eleven (48).

Table 5.1: NRMSE for input vectors selected by bio-inspired algorithms

FS method	No. Inputs	Cubist	Mon-mlp	RF	FugeR
ALO	8	22.45	24.55	21.95	-
	12	25.95	25.15	22.19	-
	20	18.73	20.20	16.33	20.15
BALO	9(1)	21.20	20.63	18.81	-
	9(2)	18.26	17.31	15.97	18.09
	9(3)	22.60	21.88	19.79	-
	11	19.40	19.35	18.70	-
	12	17.26	18.17	16.56	18.73
GWO	15	19.30	18.88	16.73	19.10
	18	20.65	18.58	17.63	-
	24	20.30	22.30	17.90	-
	25	20.04	19.29	15.86	19.10
	26	17.32	22.22	16.22	-
SSO	8	30.49	31.12	28.89	-
	13	27.09	25.82	24.86	-

Table 5.2: Results for 9in (2), trained and tested on 10cv data sets.

Algorithm	NRMSE	R2
Cubist	18.26	0.611
Monmlp	17.31	0.652
Deep learning neural nets	16.87	0.655
FugeR	18.09	0.612
RF	15.97	0.692

Chapter 6

Conclusions and future work

6.1 Conclusions

In this work, bio-inspired optimization algorithms were proposed and applied for feature selection in wrapper mode. The most recent bio-inspired optimization algorithms are hired in the feature selection domain for evaluation and results are compared against well-known feature selection methods namely PSO and GA. The evaluation is performed using a set of evaluation criteria to assess different aspects of the proposed algorithms. The feature selection is formulated as a multi-objective optimization task with a fitness function reflecting the prediction performance and feature reduction. A set of evaluation indicators is used to assess different aspects of performance of the optimization algorithms over 21 data sets in classification (10 data sets in regression) problems from the UCI repository.

Bio-inspired feature selection algorithms in modeling tablet manufacturing processes were evaluated, in particular, in the prediction of tablet properties such as porosity and tensile strength of powder characteristics. The modeling efficiency was assessed regarding the average feature reduction and RMSE. GWO algorithm was the most accurate in predicting the porosity while the most accurate prediction of the tensile strength was achieved using SSO. Finally, we can conclude that bio-inspired optimization algorithms are an efficient search algorithm and suitable for feature selection problems.

6.2 Future work

On basis of future performance, we have six different ideas that can be added to the work presented before as the following:

1. The proposed optimization algorithms will be assessed using complex data sets.
2. Add more statistics evaluation measures.
3. Employ bio-inspired optimization algorithms for solving the challenging problems and in different applications.
4. Use more machine learning techniques for wrapper-based fitness assessment such as SVM, SVR, and RF.
5. Propose a multi-objective fitness function that uses bio-inspired algorithms to the find optimal feature subset.
6. Proposes a hybrid of the recently proposed bio-inspired optimization algorithms that applies for feature selection purpose.

References

[1] B. Chizi, L. Rokach, and O. Maimon. **A Survey of Feature Selection Techniques**. *IGI Global*, pages 1888–1895, 2009. 7

[2] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*, **2nd Edition**. Wiley-Interscience, 2000. 7, 8

[3] Y. Chen, D. Miao, and R. Wang. **A rough set approach to feature selection based on ant colony optimization**. *Pattern Recognition Letters*, **31**(3):226–233, 2010. 7

[4] S. Shoghian and M. Kouzehgar. **A Comparison among Wolf Pack Search and Four other Optimization Algorithms**. *Computer, Electrical, Automation, Control and Information Engineering*, **6**(12):1619–1624, 2012. 8

[5] I.F. Jr., X.S. Yang, I. Fister, J. Brest, and D. Fister. **A Brief Review of Nature-Inspired Algorithms for Optimization**. *Elektrotehniski Vestnik*, **80**(3):116–122, 2013. 8

[6] I. Guyon and A. Elisseeff. **An introduction to variable and attribute selection**. *Machine Learning Research*, **3**:1157–1182, 2003. 8

[7] B. Xue, M. Zhang, and W.N. Browne. **Particle swarm optimization for feature selection in classification: a multi-objective approach**. *IEEE transactions on cybernetics*, **43**(6):1656–1671, 2013. 8

[8] R. Akbari, A. Mohammadi, and K. Ziarati. **A novel bee swarm optimization algorithm for numerical function optimization**. *Communications in Nonlinear Science and Numerical Simulation*, **15**:3142–3155, 2010. 8, 14

[9] Y. Miche, A. Sorjamaa, P. Bas, O. Simula, C. Jutten, and A. Lendasse. **OP-ELM: Optimally Pruned Extreme Learning Machine**. *IEEE Transactions Neural Networks*, **21**(1):158–162, 2010. 10

[10] C. Jiuwen and L. Zhiping. **Extreme Learning Machines on High Dimensional and Large Data Applications: A Survey**. *Mathematical Problems in Engineering*, **2015**(1):1–13, 2015. 10, 13

[11] J.H. Holland. *Adaptation in natural and artificial systems.* MIT Press, Cambridge, MA, USA, 1992. 10, 14

[12] R. Eberhart and J. Kennedy. **A New Optimizer Using Particle Swarm Theory**. In *International Symposium on Micro Machine and Human Science*, pages 39–43. IEEE, 1995. 10, 14

[13] H.H. Gao, H.H. Yang, and X.Y. Wang. **Ant colony optimization based network intrusion feature selection and detection**. In *International Conference on Machine Learning and Cybernetics*, pages 3871–3875. IEEE, 2005. 10

[14] H. Ming. **A rough set based hybrid method to feature selection**. In *International Symposium on Knowledge Acquisition and Modeling*, pages 585–588. IEEE, 2008. 10

[15] D. Karaboga and B. Basturk. **A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm**. *Journal of Global Optimization*, **39**:459–471, 2007. 11, 14

[16] X.S. Yang. **Engineering optimizations via nature-inspired virtual bee algorithms**. *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, **3562**:317–323, 2005. 11

[17] S. Mirjalili. **The Ant Lion Optimizer**. *Advances in Engineering Software*, **83**:80–98, 2015. 11, 16, 23

[18] A. Kibriya and E. Frank. **An empirical comparison of exact nearest neighbour algorithms**. *European Conference on Principles and Practice of Knowledge Discovery in Databases*, **4702**:140–151, 2007. 12

[19] V.Y. Kulkarni and P.K. Sinha. **Efficient Learning of Random Forest Classifier using Disjoint Partitioning Approach**. *World Congress on Engineering*, **2**, 2013. 13

[20] L. Breiman. **Random forests**. *Machine learning*, **45**(1):5–32, 2001. 13

[21] G. Biau, L. Devroye, and G. Lugosi. **Consistency of Random Forests and Other Averaging Classifiers**. *Machine Learning Research*, **9**:2015–2033, 2008. 13

[22] D.S. Huang. **A constructive approach for finding arbitrary roots of polynomials by neural networks**. *IEEE Transaction on Neural Networks*, **15**(2):477–491, 2004. 13

[23] G.B. Huang, Q.Y. Zhu, and C.K. Siew. **Extreme Learning Machine: theory and applications**. *Neurocomputing*, **70**(1):489–501, 2006. 14

[24] B. Xue, M. Zhang, and W.N. Browne. **Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms**. *Applied Soft Computing*, **18**:261–276, 2014. 14, 19

[25] B. Chakraborty. **Feature subset selection by particle swarm optimization with fuzzy fitness function**. In *Third International Conference on Intelligent System and Knowledge Engineering*, pages 1038–1042. IEEE, 2008. 14

[26] X.S. Yang. **Nature-Inspired Metaheuristic Algorithms**. *Luniver Press, UK*, **2nd Edition**, 2010. 14, 17, 19, 22

[27] X.S. Yang and S. Deb. **Cuckoo Search via Levy Flights**. *World Congress on Nature and Biologically Inspired Computing*, 2009. 14

[28] X.S. Yang. **A New Metaheuristic Bat-Inspired Algorithm**. *Nature Inspired Cooperative Strategies for Optimization*, **284**:65–74, 2010. 15

[29] X.S. Yang. **Flower pollination algorithm for global optimization**. *Unconventional Computation and Natural Computation, Lecture Notes in Computer Science*, **7445**:240–249, 2012. 15

[30] X.S. Yang, M. karamanoglu, and X. He. **Multi-objective Flower Algorithm for Optimization**. *International Conference on Computational Science, Procedia Computer Science*, **18**:861–868, 2013. 15

[31] E. Cuevas, M. Cienfuegos, D. Zaldivar, and M. Perez-Cisneros. **A swarm optimization algorithm inspired in the behavior of the social-spider**. *Expert Systems with Applications*, **40**(16):6374–6384, 2013. 15

[32] T. Jones and S. Riechert. **Patterns of reproductive success associated with social structure and microclimate in a spider system**. *Animal Behaviour*, **76**(6):2011–2019, 2008. 15

[33] S. Mirjalili, S.M. Mirjalili, and A. Lewis. **Grey Wolf Optimizer**. *Advances in Engineering Software*, **69**:46–61, 2014. 15, 21

[34] A.M.E. Khalil, S.K. Fateen, and A. Bonilla-Petriciolet. **MAKHAŮA New Hybrid Swarm Intelligence Global Optimization Algorithm**. *Algorithms*, **8**(2):336–365, 2015. 16

[35] S. Mirjalili. **Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems**. *Neural Computing and Applications*, **1**(1):1–21, 2015. 16

[36] J.H. Thorp and D.C. Rogers. *Thorp and Covich's freshwater invertebrates*, **4th Edition**. Elsevier, 2014. 16

[37] M. Wikelski, D. Moskowitz, J.S. Adelman, J. Cochran, D.S. Wilcove, and M.L. May. **Simple rules guide dragonfly migration**. *Biology letters*, **2**(1):325–329, 2006. 16

[38] S. Mirjalili. **Moth-Flame Optimization Algorithm: A Novel Nature-inspired Heuristic Paradigm**. *Knowledge-Based Systems*, **89**:228–249, 2015. 16

[39] K.J. Gaston, J. Bennie, T.W. Davies, and J. Hopkins. **The ecological impacts of nighttime light pollution: a mechanistic appraisal**. *Biological reviews*, **88**:912–927, 2013. 16

[40] A.E. Hassanien and E. Emary. **Swarm Intelligence: Principles, Advances, and Applications**. *CRC Press, Taylor & Francis Group*, 2015. 19

[41] I. Pavlyukevich. **Levy flights, non-local search and simulated annealing**. *Computational Physics*, **226**(2):1830–1844, 2007. 19

[42] A.M. Reynolds and M.A. Frye. **Free-flight odor tracking in Drosophila is consistent with an optimal intermittent scale-free search**. *PLoS One*, **2**(4), 2007. 19

[43] R. Vohra and B. Patel. **An Efficient Chaos-Based Optimization Algorithm Approach for Cryptography**. *Communication Network Security*, **1**(4):75–79, 2012. 19, 20

[44] B. Ren and W. Zhong. **Multi-objective optimization using chaos based PSO**. *Information Technology*, **10**(10):1908–1916, 2011. 20

[45] E. Emary, H.M. Zawbaa, and A.E. Hassanien. **Binary Grey Wolf Optimization Approaches for Feature Selection**. *Neurocomputing, Journal indexed in 'Journal Citation Reports' (Thomson Reuters), Impact Factor (2014): 2.083*, **172**:371–381, 2016. 23

[46] H.M. Zawbaa, S. Schiano, L. Perez-Gandarillas, C. Grosan, C.Y. Wu, and A. Michrafy. **An Evaluation of Bio-inspired Feature Selection Techniques for Computational Intelligence Modeling of Die Compaction**. *International congress on Particle Technology (PARTEC)*, 2016. 26, 28

[47] J. Szlęk, A. Paclawski, R. Lau, R. Jachowicz, and A. Mendyk. **Heuristic modeling of macromolecule release from PLGA microspheres**. *Nanomedicine*, **8**:4601–4611, 2013. 26, 28, 30

[48] H.M. Zawbaa, J. Szlek, C. Grosan, A. Mendyk, and R. Jachowicz. **Computational modelling and optimization of the macromolecule release from PLGA microspheres**. *PLOS ONE*, 2016. 26, 30