ADAPTIVE AND COOPERATIVE HARMONY SEARCH MODELS FOR RNA SECONDARY STRUCTURE PREDICTION

by

ABDULQADER MOHAMMED A. MOHSEN

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

February 2011

ACKNOWLEDGEMENTS

This thesis would not have been possible without help of others who I would to thank. First, and for most, I thank Allah for all his blessings and guidance.

I would like to thank my supervisor Associate Prof. Dr. Ahamad Tajudin bin Khader for his supervision, encouragements, guidance, constructive, and for all of his help in different aspects of the research, giving valuable suggestions and providing constructive criticism throughout my research work and preparation of this thesis. Much of appreciation to my co-supervisor Dr. Dhanesh Ramachandram for his help and support during my Ph.D. study.

I would like to acknowledge School of Computer Sciences for all the facilities and support during this research.

Thanks to members of the Computational Intelligence Research Group who were always there to discuss any research-related and unrelated issues. Abdullatif Ghallab and Farhad Nadi who brought fond memories and friendships throughout the years. I also thank the other friends in School of Computer Science, in particular Waleed Shaher.

In addition, special thanks and profound gratitude should go to Waleed Ahmed, Abdul-Hameed Al-Namshah and Yahya Al-Marrani for helping me in proofreading the thesis.

Last but not least, many thanks also go out to my parents, and my wife and children. They have always shown their faithful support during the long hours I have spent in studying and writing. I appreciate their patience during the trying time of doctoral research. Finally, I am thankful for the prayer and encouragement of my family and friends.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

LIST OF ABBREVIATIONS

ACO Ant Colony Optimization.

AHSRNAFold Adaptive Harmony Search for RNA Folding.

BW Band width.

CHSRNAFold Cooperative Harmony Search for RNA Folding.

CI Computational Intelligence.

CO Combinatorial Optimization.

COP combinatorial Optimization Problem.

DHDP Decreasing HMCR and PAR.

DHIP Decreasing HMCR and Increasing PAR.

DNA Deoxyribonucleic Acid.

DP Dynamic Programming.

EA Evolutionary Algorithms.

EC Evolutionary Computation.

Fixed Fixed values of HMCR and PAR.

FM F-measure.

FN False Negative base pairs.

FP False Positive base pairs.

GA Genetic Algorithm.

GHS Global-best Harmony Search.

GO Global Optimization.

HM Harmony Memory.

HMCR Harmony Memory Consideration Rate.

HMS Harmony Memory Size.

HS Harmony Search.

HSRNAFold Harmony Search for RNA Folding.

HSV Harmony Search Version.

IHDP Increasing HMCR and Decreasing PAR.

IHIP Increasing HMCR and PAR.

IHS Improved Harmony Search.

INN Individual Nearest Neighbour.

INN-HB Individual Nearest Neighbour Hydrogen Bonding Model.

MFE Minimum Free Energy.

mRNA Messenger Ribonucleic Acid.

NMR Nuclear Magnetic Resonance.

nt nucleotides.

PAR Pitch Adjustment Rate.

PSO Particle Swarm Optimizer.

RNA Ribonucleic Acid.

ROC Receiver Operating Characteristic.

rRNA Ribosomal Ribonucleic Acid.

RSR Random Selection Rate.

SA Simulated Annealing.

SE Sensitivity.

SetPSO Set Particle Swarm Optimizer.

SGHS A Self-Adaptive Global Best Harmony Search Algorithm for continuous optimization problems.

snRNA Small Nuclear RNA.

SP Specificity.

TP True Positive base pairs.

tRNA Transfer Ribonucleic Acid.

LIST OF SYMBOLS

$\triangle G$ Delta MFE

μ The overall HM

α Statistical α value

ρ Statistical ρ value

MODEL MUDAH SUAI DAN KERJASAMA GELINTARAN HARMONI BAGI RAMALAN STRUKTUR SEKUNDER RNA

ABSTRAK

Penentuan fungsi molekul RNA amat bergantung kepada struktur sekundernya. Kaedah fizikal yang sedia ada untuk penentuan struktur sekunder adalah mahal dan memakan masa. Beberapa algoritma telah dicadangkan untuk peramalan struktur sekunder RNA, termasuk pengaturcaraan dinamik dan algoritma metaheuristik. Gelintaran harmoni (GH) merupakan suatu algoritma metaheuristik baru yang berjaya dalam penyelesaian berbagai jenis masalah pengoptimuman. Penyelidikan ini mengusulkan tiga varian baru algoritma GH untuk menyelesaikan masalah peramalan struktur sekunder RNA. Varian pertama yang dikenali sebagai HSRNAFold adalah berdasarkan GH asas dan merupakan algoritma GH pertama untuk masalah ramalan struktur sekunder RNA. Varian kedua, AHSRNAFold, memperbaiki HSRNAFold dengan kawalan parameter mudah suai. Varian ketiga pula memperbaiki HSRNAFold dengan menggunakan model GH kerjasama dengan berbilang ingatan harmoni, dikenali sebagai CHSRNAFold. Kelakuan varian-varian GH yang baru itu dikaji dan impak penalaan parameter yang berlainan bagi varian-varian ini dinilai. Eksperimen dijalankan ke atas 20 individu dengan struktur yang diketahui dari empat kelas RNA. Kejituan peramalan ditentusahkan dengan menggunakan struktur natif dan algoritma terkini yang lain. Hasil penyelidikan ini menunjukkan bahawa CHSRNAFold memberikan keputusan yang lebih baik berbanding dengan keputusan beberapa algoritma terkini dari segi kejituan peramalan.

ADAPTIVE AND COOPERATIVE HARMONY SEARCH MODELS FOR RNA SECONDARY STRUCTURE PREDICTION

ABSTRACT

Determining the function of RNA molecules relies heavily on its secondary structure. The current physical methods for secondary structure determination are expensive and time consuming. Several algorithms have been proposed for the RNA secondary structure prediction, including dynamic programming and metaheuristic algorithms. Harmony search (HS) is a new metaheuristic algorithm which succeeded in solving many different types of optimization problems. This research proposes three new variants of HS algorithm to address the RNA secondary structure prediction problem. The first variant is called HSRNAFold as a first application of HS for RNA secondary structure prediction. The second variant, AHSRNAFold, improves HSRNAFold by using adaptive parameter control. The third variant, CHSRNAFold, improves HSRNAFold by using a cooperative multiple harmony memories model. The behavior of the new HS variants is investigated and the impact of tuning the different parameters of these variants is evaluated. The experiments were conducted on 20 individuals with known structures from four RNA classes. The prediction accuracy was verified with native structures and other state-of-the-art algorithms. The results demonstrate that CHSRNAFold outperformed several state-of-the-art algorithms in terms of prediction accuracy.

CHAPTER 1

INTRODUCTION

RNA is a nucleic acid which consists of a long linear polymer of nucleotide units found in the nucleus. RNA is similar to DNA, but usually it is single stranded instead of double-stranded, containing ribose rather than deoxyribose bases. It has uracil (U) in place of thymine (T).

The discovered biological functions of RNA have increased in recent times. The scope of understanding has expanded and RNA is no longer viewed as only a passive messenger of genetic information from DNA to proteins manufacturers as had been thought before. These new discoveries have motivated RNA research in many aspects.

RNA has been found to play important roles in all molecular biology such as carrying genetic information (messenger RNA), interpreting the code (ribosomal RNA) and transferring genetic code (transfer RNA). It also performs different functions including catalyzing chemical reactions (Doudna and Cech, 2002; Hansen et al., 2002), directing site specific modification of RNA nucleotides, controlling gene expression, modulating protein expression and serving in protein localization (Bachellerie et al., 2002; Meister and Tuschl, 2004). These functions of RNA molecules determine many diseases caused by RNA viruses. Understanding of the biological functions of an RNA molecule is fundamentally based on identifying its 3D structure (Tsang and Wiese, 2007a; Neethling and Engelbrecht, 2006). The primary structure of RNA is the easiest structure to be determined in the laboratory using gene sequencing techniques. It does not contain additional information about the functional structure. On the other hand, the tertiary structures are much more difficult to model where the secondary structure bonds

are stronger and can be formed faster than that of the tertiary structure (Zuker et al., 1999). Therefore, the computational approaches used to predict the structure of RNA have paid more attention to the secondary structure.

Since RNA structure and function are closely related, it is important to understand the common structure of homologous RNAs in order to discover their functional signatures. However, due to the exponential number of possible structures, RNA structure prediction is a complex problem. As such, it is still an open problem in bioinformatics.

1.1 Problem statement and Motivations

Physical methods used to determine the RNA secondary structure such as X-ray diffraction and nuclear magnetic resonance (NMR) spectroscopy are difficult, time consuming and expensive. Therefore, computational approaches to predict the secondary structure of RNA molecule can be considered as an appropriate alternative (Tsang, 2007).

RNA secondary structure prediction is not a trivial problem. It has been estimated that the number of secondary structures modeled from the input of n nucleotides is greater than 1.8^n (Doshi et al., 2004). For example, *Saccharomyces cerevisiae (X67579) 5S rRNA* with 118 nucleotides in length has an estimated 1.3×10^{30} secondary structure models whereas a larger RNA such as the *Sulfolobus acidocaldarius (D14876) 16S rRNA*, with 1493 nucleotides, has an estimated total of 1.3×10^{381} possible secondary structure models.

Two different computational approaches are currently in use to address the RNA secondary structure prediction problem. The first approach is called the comparative sequence analysis approach (Gardner and Giegerich, 2004; Gotoh, 1999). It is an iterative process performed on a set of homologous related RNA sequences. Briefly, sequence alignment works on the RNA

sequences similarities. This alignment is achieved by adding and removing gap characters (Deschenes, 2005). The purpose is to correlate sequence and function across genomes. The second approach is the single sequence approach, which predicts the secondary structure by searching for the minimum free energy.

Most of the methods were developed based on free energy minimization either by applying dynamic programming algorithms (DP) or metaheuristics. Based on free energy minimization of a single RNA sequence, dynamic programming algorithms have been studied since the early 1970s. Mathews (2006b) provided a review of the revolutions which occurred in the development of a number of these algorithms.

Nussinov et al. (1978) predicted the RNA secondary structure using the DP method by maximizing the number of base pairs. In 1980, they further adapted their original method to enhance the results using a simple nearest-neighbor energy model (Nussinov and Jacobson, 1980). Zuker and Stiegler (1981) proposed a slightly refined DP approach which models the nearest neighbor energy interactions and directly incorporated stacking into the prediction. Later, Zuker (2003) proposed the DP algorithm, mfold. It is still a popular algorithm used to find the minimum free energy (MFE) pseudoknot-free secondary structure of an RNA molecule. Furthermore, it has become the benchmark for predicting the RNA secondary structure. mfold uses a complex thermodynamic model to evaluate the free energy of the structures by seeking the pseudoknot-free secondary structure with MFE (Zuker, 1994, 2003). Later, RNAFold from the *ViennaRNA* (Hofacker et al., 1994) package was proposed as a dynamic programming algorithm to predict the RNA secondary structure through energy minimization.

Dynamic programming algorithm, as a mathematical technique, can hit the global optima in solving small problems. Nevertheless, in real world problems, there are some drawbacks. For example, when the number of variables increases, the number of evaluations increases

exponentially due to recursive nature of dynamic programming. For RNA secondary structure prediction, the huge number of structure alternatives makes it difficult to determine the most correct one (Tsang, 2007).

In another development, many metaheuristics algorithms were proposed such as genetic algorithms (GAs), simulated annealing (SA) and particle swarm optimization (PSO). GAs was shown to achieve higher base pairs prediction rates than DP (Gultyaev et al., 1998). The most recent GAs studies in this area are RnaPredict and its parallel version (P-RnaPredict) which were proposed by Wiese and his students (Wiese et al., 2007; Wiese and Hendriks, 2006). The results of both algorithms showed that their quality is comparable to mfold. SARNA-Predict which is an SA algorithm was introduced by Tsang and Wiese (2007a,b). It attempted to predict the RNA secondary structures with a low free energy. SARNA-Predict showed good results, with high number of correctly predicted base pairs, in comparison to known native structures and to other algorithms in the literature. Recently, two versions of PSO, setPSO and HelixPSO, were also proposed by Neethling and Engelbrecht (2006) and Geis and Middendorf (2007) respectively. Both algorithms were used to find secondary structures with low free energy.

The main drawback of local based metaheuristics approaches like SA is that they may get stuck in the local optimal solution. In addition, there is no guarantee that the value of the objective function at any local optimum is close to the optimum value (Aarts and Lenstra, 1997). On the other hand, the population-based metaheuristics approaches such as GA, ant colony and PSO have their drawbacks of premature convergence and stagnation (Qin et al., 2006).

1.2 Harmony Search Algorithm

Harmony Search (HS) algorithm is an optimization technique developed by Geem (Geem et al., 2001). HS mimics the musicians' improvisation process.

Researchers (Alatas, 2010; Lee and Geem, 2004; Geem et al., 2001; Mahdavi et al., 2007) summarized the features of HS over the other traditional optimization techniques: i) HS imposes less mathematical requirements, and as such it can be easily used for various types of engineering problems; ii) it does not require initial value settings of the decision variables, and thus, it may escape from local optima; iii) derivative information is not necessary due to stochastic random searches which HS uses; iv) HS can work with both discrete and continuous optimization problems; v) it can overcome the drawback of building block theory of GA by taking into account the relationship between the decision variables using its ensemble operation; and vi) HS algorithm generates a new vector by considering all of the existing vectors, rather than considering only two parents as in GA.

These features increase the flexibility of HS algorithm in solving a wide variety of optimization problems in several fields. Ingram and Zhang (2009) provided an overview of applications and developments using HS algorithm. These applications include continuous engineering optimization, vehicle routing, combined heat and power economic dispatch, water pump switching problem, optimal scheduling of multiple dam system and transport energy modeling (Fesanghary et al., 2009, 2008; Ayvaz, 2007; Lee and Geem, 2004, 2005; Mahdavi et al., 2007; Mohsen et al., 2008, 2009a,b; Saka, 2009; VASEBI et al., 2007; Ceylan et al., 2008; dos Santos Coelho and Mariani, 2009; Jaberipour and Khorram, 2010; Kaveh and Talatahari, 2009; Kaveh and Abadi, 2010; Mun and Geem, 2009a,b; Pan et al., 2010c; dos Santos Coelho and de Andrade Bernert, 2009; Zou et al., 2010).

Harmony search has three parameters which contribute interactively to the creation of new solution. The interaction between various components is the important factor to consider for the success of HS algorithm over other algorithms. As such, this interaction may guarantee a good balance between the intensification and diversification. Such a balance may prevent premature convergence and overcome the stagnation. For algorithm parameters, some evidences are available to suggest that HS is sensitive to chosen parameters (Mahdavi et al., 2007). This means that these parameters may need to be fine-tuned to obtain quality solutions. Furthermore, a group of multiple harmony memories can be used in parallel modeling. This parallel model may increase both the efficiency and effectiveness of the algorithm (Yang, 2009).

1.3 Objectives

The current algorithms used in RNA secondary structure prediction have some limitations and drawbacks. The aim of this thesis is not only to develop efficient HS variants for RNA secondary structure prediction, but also to show that the proposed variants are able to overcome the state-of-the-art algorithms in terms of performance.

The primary objectives of this thesis are summarized as follows:

- To adapt HS algorithm to address RNA secondary structure prediction problem.
  - To speed up the prediction process by refining the existing helix generation algorithm.
  - To evaluate the effect of using different settings for HS parameters such as harmony memory size (HMS), harmony memory consideration rate (HMCR), pitch adjustment rate (PAR) and bandwidth (BW) on the solution quality and the convergence behavior.

- To further enhance the accuracy of prediction and performance of HS for RNA secondary structure prediction problem by
  - Applying adaptive parameters for HS parameters.
  - Using a new cooperative model with multiple harmony memories.

1.4 Scope and Limitations

This thesis focuses on solving the RNA secondary structure prediction problem. Based on the assumption that the correct structure is a low energy structure, the RNA folding is subject to the laws of thermodynamics (Deschenes, 2005). The stability of the secondary structure depends on the amount of free energy released to form the base pairs. Therefore, the more negative the free energy of a structure is, the more stable a particular sequence is formed. This structure is called the MFE secondary structure (Layton and Bundschuh, 2005).

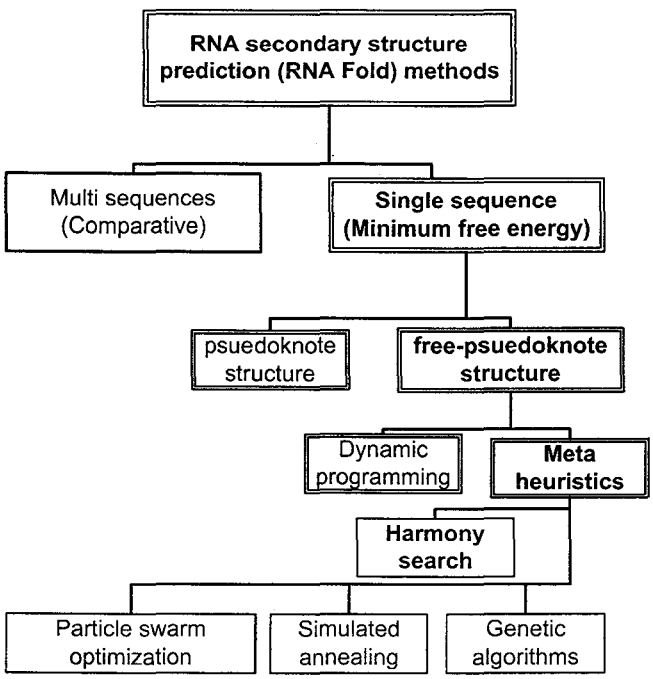

Figure 1.1: The scope of the research

In a case where only a single sequence of a given RNA molecule is known or the number of available sequences with high similarity is low, the *ab initio* methods are used to perform RNA secondary structure prediction as an energy minimization problem. These methods are either dynamic programming or metaheuristics. This research proposes new metaheuristic methods based on HS algorithm to enhance the accuracy of the prediction. As shown in Figure 1.1, the scope of study focuses on the prediction of pseudoknots-free RNA secondary structure. The prediction of the pseudoknots RNA secondary structure does not fall within the scope of this research due to the following reasons: i) the computational complexity (Hendriks, 2005); ii) the inability of the adopted thermodynamic models to deal with pseudoknot motifs (Neethling, 2008); and iii) infrequent occurrence of pseudoknots in nature (Deschenes, 2005). In the future, if corresponding thermodynamic models support the calculation of pseudoknot energy contributions, it will be easy to extend the proposed variants to enable prediction of pseudoknots.

1.5 Research Approach

This research work is divided into three processes: preprediction, prediction and postprediction as shown in Figure 1.2.

In the preprediction process, the set of all feasible helices are generated with the calculation of the free energy for each helix using the particular thermodynamic model. In the prediction process, three variants are proposed based on HS for RNA secondary structure prediction. The first proposal is to predict the structure using basic HS without modification. The second proposal is an enhanced version of first proposal by using adaptive parameters control. Two parameters- harmony memory consideration rate (HMCR) and pitch adjustment rate (PAR)- are affected by this adaptation. The third proposal is the new cooperative model of the basic HS with multiple HMs. In all the proposals, the same mechanism is used for RNA secondary structure encoding, decoding, the thermodynamic models and the harmony representation.

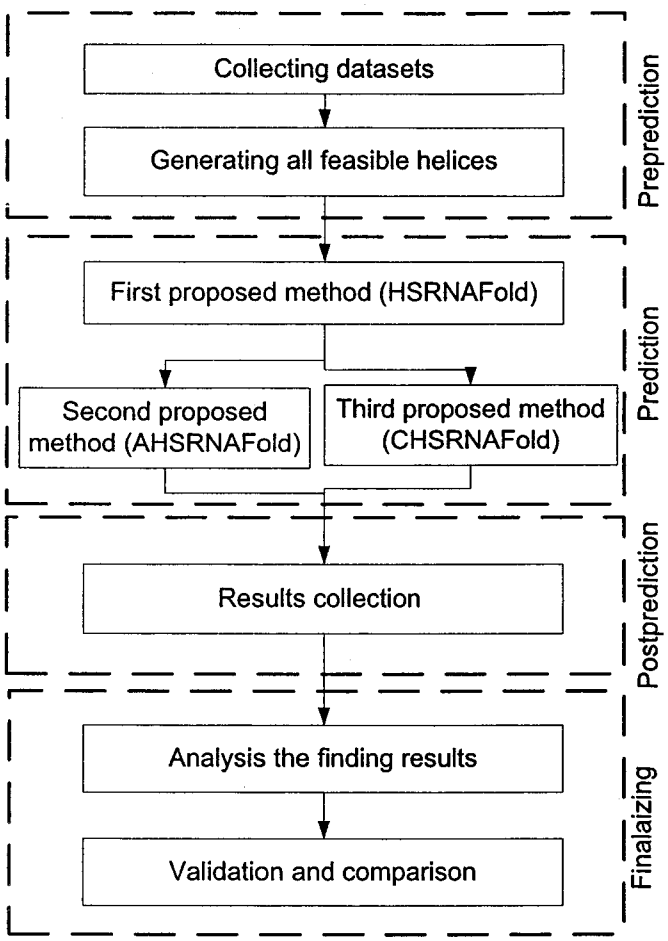

Figure 1.2: The Methodology.

The third process includes the collection of the results of each proposal for analysis and discussion. In this step, the best predicted secondary structure is also generated.

Comprehensive study and discussion are performed on a variety of RNA classes (5S rRNA, Group I intron 16S rRNA, Group I intron 23S rRNA and 16S rRNA) to study the performance of the three proposed variants of HS and the accuracy of prediction. For the performance, the convergence behavior of the three variants is examined and various parameters' setting are investigated. In terms of prediction accuracy, an evaluation of the performance of the new variants is performed via the comparison to other RNA secondary structure prediction algorithms such as mfold (Zuker, 1994, 2003), RNAFold (Hofacker et al., 1994), RnaPredict

and P-RnaPredict (Wiese et al., 2007; Wiese and Hendriks, 2006; Hendriks, 2005; Deschenes, 2005), setPSO (Neethling and Engelbrecht, 2006), HilexPSO (Geis and Middendorf, 2007) and SARNA-Predict (Tsang and Wiese, 2007b,a; Tsang, 2007)).

1.6 List of Contributions

This research investigates ideas in the direction of improving HS performance for RNA secondary structure prediction. There seem to be several exciting research issues connected with parameter control as well as the cooperative model of HS, which are investigated in this research. The main contributions include the following:

- A new variant of HS algorithm called HSRNAFold as the first application of HS for RNA secondary structure prediction (Geem, 2010).
- An improved variant of HSRNAFold based on adaptive parameters called AHSRNAFold.
- A new variant called CHSRNAFold algorithm which differed from the original HSRNAFold by operating on cooperative multiple HMs model to enhance both algorithm performance and accuracy of prediction.
- A comprehensive study of the influence of the main parameters of HSRNAFold and its subsequent variants which may affect the algorithm's performance when used in a real world optimization problem.

A comparative study of the three proposed HS variants was performed amongst themselves and against the state of the art algorithms for RNA secondary structure prediction, and then to the native structures.

1.7 Thesis Outline and Organization

The organization of the remaining chapters of this thesis is as follows:

- Chapter 2 provides an overview of the original HS algorithm.
- Chapter 3 introduces the RNA structures, RNA secondary structure prediction problem and related work, and outlines the shortcomings of existing methods.
- Chapter 4 presents the three variants of HS algorithm: HSRNAFold, AHSRNAFold and CHSRNAFold. In addition, the major modified and enhanced components of the three methods are also presented.
- Chapter 5 presents the experimental setup and a comprehensive investigation of the performance of the three variants of HS with different parameter setting.
- Chapter 6 presents the experimental results and evaluates the validity of the proposed algorithm.
- Chapter 7 presents the concluding remarks, suggestions and some possible directions for future research.

CHAPTER 2

LITERATURE REVIEW

This chapter provides a description on the HS fundamentals. Section 2.1 gives a brief overview of the HS procedures. Section 2.1.1 describes the HS procedure. Section 2.1.2 reviews the main HS optimization steps. Section 2.1.3 and 2.1.4 provide a summary of the related work that has been done in adaptive parameters and multiple harmony memories.

A review of the related literature pertaining to RNA secondary structure prediction will be provided as well. Section 2.2 gives a quick review of the RNA secondary structure. In Section 2.3, Two physical methods for determining RNA structure are presented. Section 2.4 provides information on the related work of the two major computational methods for RNA secondary structure prediction: multiple sequences and single sequence methods. Finally, Section 2.5 summarizes and concludes the chapter.

2.1 Harmony Search Algorithm

Harmony search was initiated by Geem and his colleagues in 2001 (Geem et al., 2001) as a relatively new metaheuristic for hard combinatorial optimization problems (for more details about metaheuristic and combinatorial optimization see Appendix A). Harmony search is a stochastic search technique based on the mechanism of improvisation process to find fantastic harmony. It has received a great deal of attention regarding its potential as an optimization technique for solving discrete and continuous optimization problems (for more details about discrete and continuous optimization see Appendix A).

2.1.1 Fundamental Procedures of HS

In HS, harmony parameters are usually used to create new harmony in each improvisation. The main role of these parameters is to direct the search toward the most favorable areas of the search space. These parameters are:

- Harmony memory size (HMS) representing the total number of harmonies in the HM.
- Harmony memory consideration rate (HMCR) which represents the probability of picking up values from HM to the variables in the solution vector.
- Random selection rate (RSR) representing the probability of randomly chosen feasible values from the range of all possible values to the variables in the solution vector, formally, $RSR = 1 - HMCR$.
- Pitch adjusting rate (PAR) representing the probability of further adjusting the pitch with neighboring pitches.
- Number of improvisations (NI) representing the number of iterations to be used during the solution process, or stopping criterion.

To explain the fundamental procedures of HS, consider a harmony memory that consists of N harmonies representing potential solutions to a problem. In HS, a harmony in harmony memory is represented by a string S of length n as follows: $S = S_1, S_2, \ldots, S_j \ldots, S_n$.

The string S is regarded as a harmony that consists of n musical instruments. The character S_j is a musical instrument at the j^{th} locus, and the different values of a musical instrument are called notes. The harmony is a potential solution to a problem corresponding to a string S called the solution vector. In minimization problems, the string with a smaller objective function value has a higher fitness.

HS starts with an initial HM of n harmonies generated randomly. Each harmony in the harmony memory represents a potential solution of the problem under consideration. Each harmony in the harmony memory is evaluated using an objective function. The harmonies evolve through successive iterations, called improvisations. During each improvisation, a new harmony is created through harmony operators. After that, the harmony memory is updated if the new harmony is better than its worst one. The procedure continues until the termination condition is satisfied. When the termination condition is satisfied, the best harmony obtained is regarded as an optimal or approximate optimal solution to the problem.

When applying HS to solve particular optimization problems, further detailed considerations are required: i) representation for potential solutions, ii) a way to create an initial harmony memory, iii) an evaluation process in terms of their objective function, iv) harmony parameters, v) constraint-handling techniques, vi) tuning for various parameters in HS such as HMS, HMCR and PAR and vii) termination conditions.

2.1.2 HS Optimization Steps

Figure 2.1 shows the optimization steps of HS which is presented in detail in the next subsections.

2.1.2(a) Initialize the Problem and Algorithm Parameters

Mathematically, the general form of optimization problem can be specified as follows:

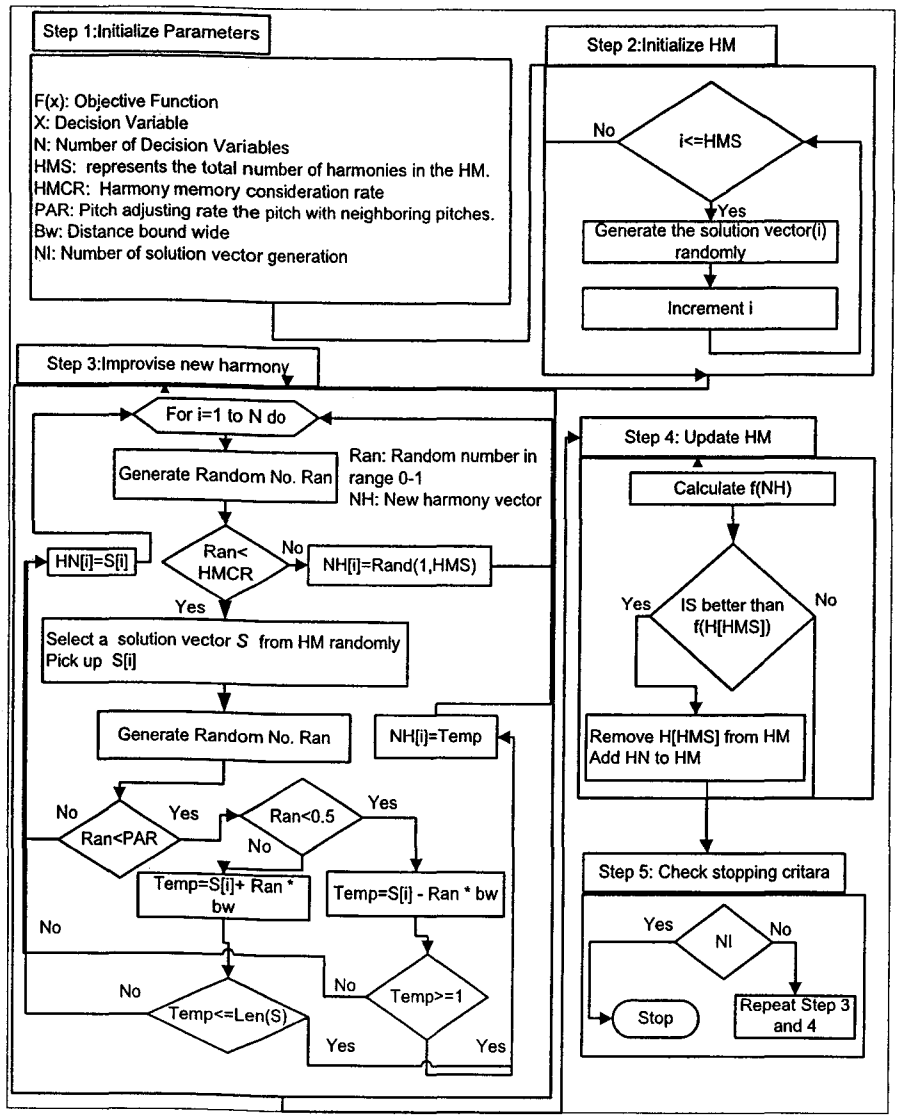

Figure 2.1: Optimization procedure of the simple HS algorithm (Mahdavi et al., 2007).

$$\begin{cases} Minimize\, f(x) \\ Subject\, to\, g(x) > 0, x = \{x_1, x_2, \ldots, x_n\} \\ h(x) = 0 \end{cases} \tag{2.1}$$

Where $f(x)$ is the objective function; $g(x)$ and $h(x)$ are the inequality and equality con-

straint functions respectively; x is the set of each decision variable x_i; and n is the number of decision variables (music instruments). HS algorithm parameters that are required to solve the optimization problem (i.e., HMS, HMCR, PAR, BW and NI) are also specified in this step. These parameters are used to improve the solution vector.

2.1.2(b) Initialize the Harmony Memory

Initialize the HM matrix($N \times HMS$) where N is the number of decision variables and M is HMS. Then fill the HM randomly by generating the feasible solution vectors. Formally, HM and the corresponding fitness function values are shown as follows:

$$HM = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_{N-1}^1 & x_N^1 \\ x_1^2 & x_2^2 & \dots & x_{N-1}^2 & x_N^2 \\ \vdots & \dots & \dots & \dots & \dots \\ x_1^{HMS-1} & x_2^{HMS-1} & \dots & x_{N-1}^{HMS-1} & x_N^{HMS-1} \\ x_1^{HMS} & x_2^{HMS} & \dots & x_{N-1}^{HMS} & x_N^{HMS} \end{bmatrix} \begin{matrix} \Rightarrow f(x^1) \\ \Rightarrow f(x^2) \\ \Rightarrow \vdots \\ \Rightarrow f(x^{HMS-1}) \\ \Rightarrow f(x^{HMS}) \end{matrix} \quad (2.2)$$

Where each $x' = (x_1^1 x_2^1 \dots x_N^1)$ and $f(x^1)$ represents a feasible solution vector and it's corresponding objective function respectively.

2.1.2(c) Improvise a New Harmony

A new harmony vector $x' = (x_1' x_2' \dots x_N')$, is generated based on three parameters: memory consideration, pitch adjustment and random selection as follows (Geem et al., 2001):

i) for each component x_i', pick up the corresponding component of x_i' randomly from any of

the values in the specified HM range $(x_i^{'1} - x_i^{'HMS})$ with the probability of P_{hmcr}.

$$x_i^{'} \leftarrow \begin{cases} x_i^{'} \in \{x^{'} = (x_i^1, x_i^2, \ldots : x_i^{HMS}\} & with\ probability\ HMCR \\ x_i^{'} \in Xi & with\ probability\ (1 - HMCR) \end{cases} \tag{2.3}$$

ii) the rest of the components of $x_i^{'}$ are picked randomly from the range of allowed values with the probability of $1 - P_{hmcr}$. For example, HMCR of 0.95 indicates that the probability of HS algorithm to choose the decision variable values from historically stored values in the HM is 95% and the probability of choosing a new random value from the allowed range is (100-95)%.

iii) change $x_i^{'}$ with the probability of P_{par}. The pitch adjustment is applied only if the value is chosen from the HM.

$$Pitch\ adjusting\ decision\ for\ x_i^{'} \leftarrow \begin{cases} Yes\ with\ probability\ PAR, \\ No\ with\ probability\ (1 - PAR). \end{cases} \tag{2.4}$$

If the pitch adjustment decision for $x_i^{'}$ is yes, then small amount (bw) of changes takes place for pitch adjustments:

$$x_i^{'} \leftarrow x_i^{'} \pm bw * rand(). \tag{2.5}$$

2.1.2(d) Harmony Memory Update

Evaluate the new harmony $x^{'} = (x_1^{'} x_2^{'} \ldots x_N^{'})$ by calculating it's objective function. If the value of its objective function is better than that of the objective function of the worst harmony in the

HM, the new harmony is included in the HM and the existing worst harmony is excluded from the HM. Subsequently, the vectors are sorted out based-on their objective function values.

2.1.2(e) Termination Criterion Check

Stop the search process if a maximum number of iterations (number of improvisations) is reached. Otherwise, repeat steps three and four.

2.1.3 Adaptive Parameters

The study of the adaptive parameters in HS started in 2007 by Mahdavi et al. (2007) with his algorithm called Improved Harmony Search (IHS). Many subsequent studies were inspired by IHS. However, some of these studies disagreed with IHS. In IHS the fine-tuning was done for two parameters, PAR and BW. These two parameters control the convergence rate of HS. Figure 2.2 shows the fine-tuning that has been applied dynamically by increasing and decreasing the values of PAR and BW respectively. They claimed that IHS overcomes the drawbacks of using fixed values of PAR and BW in the simple HS algorithm. Formally, PAR and BW are updated dynamically according to Equation 2.6 and Equation 2.7 respectively.

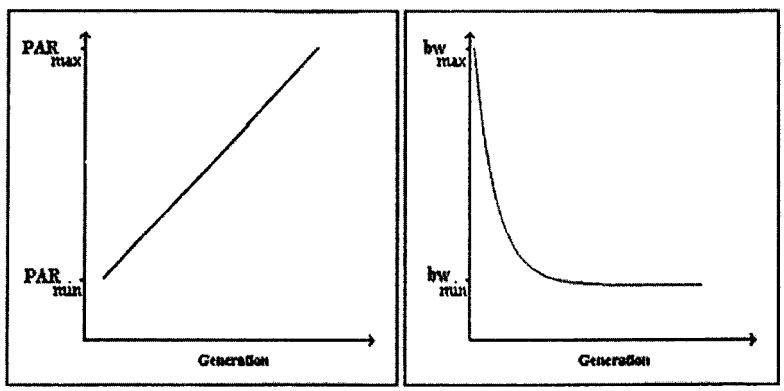

Figure 2.2: Variation of PAR and bw versus generation number (Mahdavi et al., 2007).

$$PAR(g) = PAR_{min} + \frac{PAR_{max} - PAR_{min}}{NI} \times g \tag{2.6}$$

$$BW(g) = BW_{max} \exp\left(\frac{\ln \frac{BW_{min}}{BW_{max}}}{NI} \times g\right), \tag{2.7}$$

where PAR(g) and BW(g) are the pitch adjustment rate and the distance bandwidth in generation g respectively; NI is the maximum number of iterations, and g is the current iteration; PAR_{min} and PAR_{max} are the minimum and the maximum pitch adjustment rate respectively; BW_{min} and BW_{max} are the minimum and the maximum bandwidth respectively.

IHS algorithm has critical drawbacks (Wang and Huang, 2010). For instance, there is difficultly in setting suitable values of BW_{min} and BW_{max} and, on the other hand, PAR should be decreased with search time to limit perturbation. Surprisingly, Omran and Mahdavi (2008) in their subsequent research claimed that they achieved better results, in spite of giving PAR a small constant value.

Later, Omran and Mahdavi (2008) developed another version of HS called the Global-best Harmony Search (GHS). GHS is different from the simple HS in the improvisation step by modifying the pitch adjustment rule. The idea was inspired from swarm intelligence to enhance the performance of HS. To improvise new harmony, the pitch adjustment of the GHS was modified such that a new harmony is affected by the best harmony in the harmony memory. GHS simplifies the pitch adjustment step and BW is not used anymore. Formally, the rule to

adjust the pitch is given in Equation 2.8 as follows:

$$X_{new}(j) = X_B(k)\ , j = 1,2,\ldots,\ n\ and\ k = Rand(1,n), \tag{2.8}$$

where X_{new} is the new harmony, X_B is the best harmony in harmony memory and k is a random integer between 1 and n. According to Omran and Mahdavi (2008), this modification allows the GHS algorithm to work more efficiently on both continuous and discrete problems.

The GHS was also criticized by Wang and Huang (2010). They listed a number of disadvantages of GHS. It suffered from premature convergence. Moreover, there are some obvious mistakes in the GHS and so the reliability of the numerical results is decreased.

Later, dos Santos Coelho and Mariani (2009) proposed a modified version of HS. They inspired the concept from Mahdavi (Mahdavi et al., 2007) for using variable PAR with small changes to the Equation 2.2. The modification is the inclusion of the grade of the solution vectors into Equation 2.6. The grade is updated according to the following expression:

$$Grade = \frac{F_{max}(g) - mean(F)}{F_{max}(g) - F_{min}(g)}, \tag{2.9}$$

where $F_{max}(g)$ and $F_{min}(g)$ are the maximum and minimum objective function values in generation g, respectively; mean (F) is the mean of the objective function value of the harmony memory. The new PAR rule shown in Equation 2.10 as follows:

$$PAR(g) = PAR_{min} + \frac{PAR_{max} - PAR_{min}}{NI} \times g \times grade \tag{2.10}$$

Then, Pan et al. (2010c) proposed a variant of HS called A Self-Adaptive Global Best

Harmony Search Algorithm for continuous optimization problems (SGHS). Unlike GHS, in SGHS the value of the decision variable $X_B(j)$ in X_B is assigned to $X_{new}(j)$, while in GHS, $X_{new}(j)$ is determined by selecting one of the decision variables of X_B randomly. According to Equation 2.11, PAR is updated as follows:

$$X_{new}(j) = X_B(j)\ ,\ j = 1,2,\ldots,n \tag{2.11}$$

where X_{new} is the new harmony, X_B is the best harmony in harmony memory and j is an integer between 1 and n which refers to the current location in the corresponding harmony. The results showed that the SGHS algorithm outperforms the existing HS, IHS and GHS algorithms.

2.1.4 Multiple Harmony Memories Models

According to Yang (2009), since HS algorithm is a population-based metaheuristic, a group of multiple harmonies can be used in parallel. Proper parallelism could result in a better performance with higher efficiency. Conducting a balance between intensification and diversification could also be achieved with the use of parallelism and elitism.

Pan et al. (2010b,a) proposed two variants. The first one is called referred to them as a local-best harmony search algorithm with dynamic subpopulations for solving continuous optimization problems; and the other is called a local-best harmony search algorithm with dynamic sub-harmony memories for lot-streaming flow shop scheduling problem. These two are the only methods that take advantage of multiple harmony memories to improve the HS performance. Numerical experiments showed that these techniques overcome the existing HS, IHS, GHS, and MHS algorithms. The methods have some limitations due to the incomplete model of multiple harmony memories and the effects of the parameters of the multiple harmony memories model was neglected.

2.2 RNA Secondary Structure

The linear sequence of RNA molecule consists of a single stranded sequence of four nucleotides. This linear sequence is the primary structure of RNA molecule. The RNA strand has the ability to fold back upon itself. During the folding process, the hydrogen bonds which lie between different nucleotides form base pairs. These hydrogen bonds, which occur mostly between G and C or A and U, are called the Watson-Crick base pairs and the bond between G and U is called the wobble base pair. These base pairs- GC, AU, and GU, and their mirrors, CG, UA, and UG- are called the canonical base pairs.

Definition 2.1. *Given a single stranded RNA sequence of length* $L, x = (x_1, x_2, ..., x_L)$, *with* $x_i \in \{A, C, G, U\}$ *for all i, the RNA secondary structure for x is defined as a set P of ordered base pairs, written as (i, j), with* $1 \leq i \leq j \leq L$, *which satisfy the following constraints (Wiese and Hendriks, 2006; Mathews, 2006a; Zuker, 1994):*

- *i) for (i,j), it must be a canonical base pair;*
- *ii) each base pair cannot share more than one base (nucleotide);*
- *iii) pairing bases must be at least three bases apart* $i - j > 3$*; and*
- *iv) two base pairs must not cross, i.e.:* $\{i, j\} \bigcap \{i', j'\} = \Phi$ *or for all* (i, j), (i', j') *either* $i < i' < j' < j$ *or* $i' < i < j < j'$ *holds.*

The RNA secondary structure has a number of elements including stacked base pairs which form helices, hairpin loops, internal loops, bulges, multi-branched loops and external bases.

- **Hairpin loop** is a group of nucleotides which are enclosed by a helix but not canonically paired. Formally, in a given secondary structure, the tuple (i, j) defines a hairpin loop if

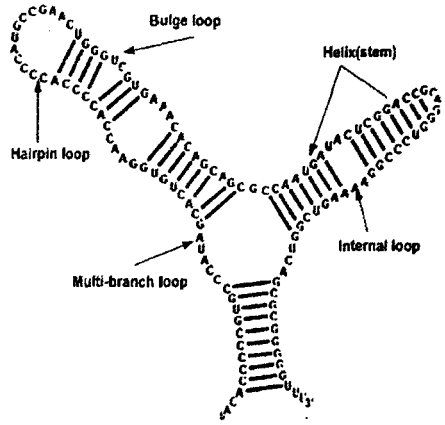

Figure 2.3: RNA secondary structure components: stems (helices), interior loops, hairpin loop, multi loops and bulges loops. This figure was created using jViz.RNA (Wiese et al., 2005) for the *Deinococcus radiodurans* organism.

i and j are paired, and $[i+k, j-k]$ is an empty region $\forall k > 3, (i, j)$ is called the closing base pair of the hairpin loop. The hairpin marked in Figure 2.3 contains two hairpin loops with thirteen and four unpaired bases from left to right respectively.

- **Stacked loop** or helix contains a contiguous stacking of base pairs. Formally, in a given secondary structure, a tuple (i, j) defines a stacked pair if (i, j) are paired and consecutively $(i+k, j-k)$ are base pairs, $\forall k >= 3$. The structure in Figure 2.3 contains six helices. Generally, stacked pairs exist when two or more base pairs exist in such a way that the ends of the pairs are adjacent, forming a helical structure as shown in Equation 2.12:

$$(i, j), \ldots, (i+n, j-n), i \leq n < m, where \quad m = \frac{j-i-3}{2}, n \in [1 \ldots m] \tag{2.12}$$

- **Internal loop**, sometimes called interior loop, is a loop inside the helices which separates two helices by having unpaired or no canonically paired nucleotides. An internal loop is symmetric if the number of nucleotides in each side of the helix is tied, asymmetric otherwise. Formally, the tuple (i, j) and the tuple $(\acute{i}, \acute{j})$ define an internal loop if (i, j) are paired, $(\acute{i}, \acute{j})$ are paired with $i + 1 < \acute{i} < \acute{j} < j - 1$ and k_1, k_2 are unpaired regions,

$\forall k,\ i < k_1 < \acute{i}$ *and* $\acute{j} < k_2 < j$. The structure in Figure 2.3 contains two symmetric internal loops with six and four unpaired bases from left to right respectively.

- **Bulge loop** interrupts helices by having unpaired nucleotides, but occurs at only one side. It is considered as a special case of internal loop, where it has no free base on one side, but has at least one free base on the other. The bulge loop marked in Figure 2.3 contains two unpaired bases.

- **Multi-branched loop** is a loop region which arises from the confluence of three or more helices. In other words, it is enclosed by three or more base pairs. Formally, $(i_1, j_1, i_2, j_2, ..., i_m, j_m)$, with $m \geq 3, i_1 < i_2 < j_2 < ... < i_m < j_m < j_1$ define a multi-loop with m branches if $(i_1, j_1), (i_2, j_2), \ldots, (i_m, j_m)$ are base pairs and $k_1, k_2, ..., k_m$ are free bases, $\forall k, i_1 > k_1 < i_2, j_2 < k_2 < i_3, ..., j_m < k_m < j1$. The multi-branched loop marked in Figure 2.3 contains one multi-branched loop with three branches containing five, one and three unpaired bases from left to right respectively.

- **external loop** is consist of all those bases that are not enclosed by a base pair in the structure. In Figure 2.3 there are two unpaired bases in both sides.

The stability of the RNA secondary structure is quantified as the amount of free energy being released or used by the formation of base pairs. The stability increases according to the number of GC versus AU and GU base pairs, and the number of base pairs in a hairpin loop region. The number of unpaired bases such as interior loops or bulges decreases the stability of the structure (Tsang, 2007).

2.3 RNA Secondary Structure Determination

Two primary physical methods are available for determining RNA structure: X-ray crystallography and Nuclear Magnetic Resonance (Cheong et al., 2004; Mattson et al., 1997; Neidle,