Netw Model Anal Health Inform Bioinforma (2012) 1:69–85
DOI 10.1007/s13721-012-0013-7

ORIGINAL ARTICLE

# A hybrid harmony search algorithm for ab initio protein tertiary structure prediction

**Mohammed Said Abual-Rub · Mohammed Azmi Al-Betar · Rosni Abdullah · Ahamad Tajudin Khader**

Received: 10 March 2012 / Revised: 20 May 2012 / Accepted: 23 May 2012 / Published online: 4 July 2012


**Abstract** Predicting the tertiary structure of proteins from their linear sequence is a big challenge in biology. The existing computational methods are not powerful enough to search for the precise structure in a huge conformational space. This inadequate capability of the computational methods, however, is a major obstacle when trying to tackle this problem. The observations of some previous studies have revealed much interest in hybridizing a local search-based metahuristic algorithm within the population-based metahuristic algorithm. This study introduces a hybrid harmony search algorithm (HHSA) as a means to solve ab initio protein tertiary structure prediction problem. In HHSA, the iterated local search (ILS) is incorporated with the harmony search algorithm (HSA) to empower it so as to find the local optimal solution within the search space of the new harmony. Furthermore, the global-best concept of particle swarm optimization (PSO) is incorporated in memory consideration as a selection scheme to accelerate the convergence speed. The HHSA predicts the tertiary structure of a protein giving its sequence alone (i.e., from scratch). Our algorithm converges faster than the classical harmony search algorithm. We evaluate our algorithm using two protein sequences. The results show that our algorithm can find more precise solutions than other previous studies.

**Keywords** ab initio protein structure prediction · Protein folding · Harmony search · Metaheuristic algorithms · Optimisation · Local search

**Abbreviations**

| | |
|---|---|
| AHSA | Adaptive harmony search algorithm |
| HHSA | Hybrid harmony search algorithm |
| HMCR | Harmony memory consideration rate |
| HSA | Harmony search algorithm |
| PAR | Pitch adjustment rate |
| PSP | Protein structure prediction |
| SMMP | Simple molecular mechanics for proteins |
| NMR | Nuclear magnetic resonance |

## 1 Introduction

Bioinformatics refers to the field concerned with the analysis of biological information including link prediction and classification (Almansoori et al. 2012), detecting disease (Tang et al. 2012), and others using computers and statistical techniques. Predicting the three-dimensional structure of a protein from its linear sequence is currently a great challenge in computational biology . The problem can be described as the prediction of the three-dimensional structure of a protein from its amino acid sequence or the prediction of a protein's tertiary structure from its primary structure. There are two categories of methods for protein

M. S. Abual-Rub
college of Shari′a and Islamic Studies , Imam Muhammad Ibn Saud Islamic University, AlAhsaa 31982, Kingdom of Saudi Arabia
e-mail: mohammad@cs.usm.my

M. A. Al-Betar (✉)
Department ofComputer Science, Jadara University, PO Box 733, Irbid, Jordan
e-mail: mohbetar@cs.usm.my

M. A. Al-Betar · R. Abdullah · A. T. Khader
School of Computer Sciences, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia
e-mail: rosni@cs.usm.my

A. T. Khader
e-mail: tajudin@cs.usm.my

structure prediction: experimental and computational. The two main experimental methods available for protein structure prediction are X-ray crystallography and nuclear magnetic resonance (NMR). Unfortunately, these methods are not efficient enough being both expensive and time-consuming (Abual-Rub and Abdullah 2008). There are currently three main categories of computational methods for protein structure prediction. These categories depend mainly on the percentage of similarity of the input protein sequence with other existing sequences in the database. The first is homology modeling—also known as comparative modeling. It is used when there is a similarity between the target sequence and the sequences that already exist in protein database (Chothia and Lesk 1986). The second is fold recognition—also known as protein threading, which is an inverse of the protein folding problem. It is based on the fact that the number of the new folded protein structure is not growing fast compared to the number of new protein sequences, which leads to the observation that any new predicted structure will be almost folded to an existing structure in the database. The third computational prediction category is ab initio modeling. It seeks to predict the tertiary structure of a protein from its amino acid sequence alone—without any knowledge of similar folds. Ab initio—also known as de novo modeling, free modeling, or physics-based modeling (Lee et al. 2009)—is based on the thermodynamic hypothesis which states that the tertiary structure of the protein is the conformation with the lowest free energy (Anfinsen 1973). Ab initio modeling, however, is challenging for the following reasons. First, there is a huge number of proteins that have no homology with any of the known structure proteins. Second, some proteins which show high homology with other proteins have different structures. Third, comparative modeling does not offer any perception of why a protein adopts a specific structure (Helles 2008). A successful ab initio method for protein structure prediction depends on a powerful conformational search method to find the minimum energy for a given energy function. molecular dynamics (MD), Monte Carlo (MC) and genetics algorithm (GA) are common methods to explore protein conformational search space.

A recent meta-heuristic population-based optimization algorithm, which mimics the improvisation process in the musical context (Geem et al. 2001), is a harmony search algorithm (HSA). It has special advantages in comparison with traditional optimization techniques: it requires fewer mathematical requirements without initial value settings for decision variables; it considers all the existing vectors to generate a new vector, whereas the methods like genetic algorithm (GA) only considers the two parent vectors, and HSA does not need to encode and decode the decision variables into binary strings (Mahdavi and Abolhassani 2009). These advantages enable it to be successfully used for a wide variety of optimization problems such as RNA secondary structure prediction (Mohsen et al. 2010), timetabling (Al-Betar and Khader 2012; Al-Betar et al. 2010a, b, c), Structural Engineering (Saka et al. 2011), and many others as overviewed by Ingram and Zhang (2009); Alia and Mandava (2011). Furthermore, the structure and performance of the HSA are under development to be adopted for the ongoing challenges. It is hybridized with other successful optimization techniques such as GA (Nadi et al. 2010), particle swarm optimization (PSO) (Omran and Mahdavi, 2008) and Hill climbing (Al-Betar et al. 2012b). Furthermore, its parameter is deterministically adaptive during the search (Mahdavi et al. 2007; Geem and Sim, 2010; Pan et al, 2010; Alatas, 2010). Quite recently, there have been some mathematical analysis studies to investigate the exploratory power of HSA (Das et al. 2011; Al-Betar et al 2012a).

The main objective of this paper is twofold: (1) to adapt HSA for ab initio protein tertiary structure prediction (PPSP) which can be set as an initial study to apply this algorithm for this problem (henceforth called adaptive harmony search algorithm, AHSA), (2) to hybridize iterated local search (ILS) within the process of the AHSA to improve its local exploitation and incorporate global-best concept of PSO in the memory consideration to improve the convergence speed (henceforth called hybrid harmony search algorithm, HHSA). Using a well-studied benchmark established for PPSP, the results show that the AHSA is, by comparison, able to competitively provide a good quality solution. Interestingly, HHSA is able to yield more precise results than those of the comparative methods.

## 2 Materials and methods

### 2.1 Problem description

The PSPP considered in this paper is the ab initio protein tertiary structure prediction, which predicts the tertiary structure of a protein from its amino acids sequence alone. It is based on the thermodynamic hypothesis (Anfinsen 1973) which states that the tertiary structure of the protein is the conformation with the lowest free energy. Thus, the PSPP can be formulated as an optimization problem whose basic objective is to find the conformation that has the lowest energy of the protein. The most important task in solving the protein structure prediction problem using an optimization algorithm is to choose an applicable representation of the conformation and a suitable energy function. The problem modeling is introduced based on these two factors in the following two sections.

## 2.2 Problem modeling

### 2.2.1 Problem representation

There are many common representations of polypeptide chains such as (Cutello et al. 2006): Cα coordinates, all-heavy-atom coordinates, all-atom three-dimensional coordinates, backbone atom coordinates + side-chain centroid, and backbone and side-chain torsion angles.

The most detailed representation is the one that includes all atoms of the protein. It is worth mentioning that representing these all atoms with their interactions is computationally expensive though not essential during the search process (Chivian et al. 2003). Hence, to reduce the computational time and space, many researchers, such as Levitt (1976), Abagyan and Maiorov (1988), Hinds and Levitt (1992), Baker (2000), and Dudek and Objects (2007) used a simplified representation. This research, likewise, uses a simplified representation of the polypeptide chain; namely, backbone and side-chain torsion angles representation based on the fact that each residue type requires a fixed number of torsion angles to fix the three-dimensional coordinates of all atoms.

The protein is represented in this research as a vector of amino acids, $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_M)$, where $\boldsymbol{x}_1 = \Phi_1(1), (\Psi_1(1), \omega_1(1), x_1(1), x_1(2), \ldots, x_1(N)), \boldsymbol{x}_2 = (\Phi_2(1), \Psi_2(1), \omega_2(1), x_2(1), x_2(2), \ldots, x_2(N)), \ldots, \boldsymbol{x}_M = (\Phi_M(1), \Psi_M(1), \omega_M(1), x_M(1), x_M(2), \ldots, x_M(N))$. However, AHSA deals with this vector as a vector of torsion angles $\mathbf{x}_i = (x(1), x(2), \ldots, x(N))$, where $N$ is the number of torsion angles in the protein, and each angle $(x(i))$ in this vector can be assigned with a value within the range $[-\pi, \pi]$. Therefore, the solution length is equal to the number of torsion angles in the protein. The amino acid consists of two parts: (1) main chain angles $(\Phi, \Psi, \omega)$ and (2) side chain angles $x_i = x_i(1), x_i(2), \ldots, x_M(N))$. Each amino acid essentially includes the main chain, while the number of side chain angles depends on the amino acid type. In other words, this research deals with the side chain vector of amino acids each assigned with a particular torsion angle $\mathbf{X} = (x(1), x(2), \ldots, x(N))$, where $N$ is the number of torsion angles in the protein, and each angle $(x_i(j))$ in this vector can be assigned with a value within the range $[-\pi, \pi]$.

### 2.2.2 Energy function

There are many well-known physics-based force fields including: CHARMM by Brooks et al. (1983), AMBER by Weiner et al. (1984), and OPLS by Jorgensen and Tirado-Rives (1988). Unfortunately, these force field packages are complex and difficult to modify by user to accommodate his/her own algorithm (Eisenmenger et al. 2006). Moreover, many researches need to determine an energy function that can be easily modified and adapted to specific needs of the user; such energy function also needs to be efficient for the evaluation studies. Therefore, the energy function used in this research is simple molecular mechanics for proteins (SMMP), which is a modern package for simulation of proteins. This force field package has been established by Eisenmenger et al. (2001) and has been revised by Eisenmenger et al. (2006). This research uses the revised version.

Arguably, many reasons can be given for such use: First, the program is fast and may be successfully exploited even on a single PC. Second, its code is free and has an open source. Third, the code is simple and can be modified and adapted by users to meet their specific needs. Fourth, the program does not contain any machine-dependent routines. Finally, it has been tested in many simulations of small peptides and has been proved to be a convenient and effective tool for numerical investigations of proteins (Eisenmenger et al. 2006). In SMMP, the protein molecule is described by the set of internal coordinates, in which the dihedral angles $(\Phi, \Psi, \omega)$ that describe rotations around the chemical bonds in the backbone of the amino acids, and the dihedral angles $\boldsymbol{X}_i$ in the side chains, are flexible. A set of energy minimization routines are used based on ECEPP force field with two different parameter sets to calculate the internal energy: ECEPP/2 potential and ECEPP/3 potential.

SMMP used the following energy function (Eisenmenger et al. 2001):

$$\min \quad f(\mathbf{x}) = E_{\mathrm{LJ}} + E_{\mathrm{el}} + E_{\mathrm{hb}} + E_{\mathrm{tors}} \tag{1}$$

where

$$E_{\mathrm{LJ}} = \sum_{j>i} \left( \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}} \right) \tag{2}$$

$$E_{\mathrm{el}} = 332 \sum_{j>i} \frac{q_i \times q_j}{\varepsilon \times r_{ij}} \tag{3}$$

$$E_{\mathrm{hb}} = \sum_{j>i} \left( \frac{C_{ij}}{r_{ij}^{12}} - \frac{D_{ij}}{r_{ij}^{10}} \right) \tag{4}$$

$$E_{\mathrm{tors}} = \sum_{n} U_n (1 \pm \cos(k_n \times \varphi_n)) \tag{5}$$

where $r_{ij}$ refers to the distance in {Å between atoms $i$ and $j$ while $\{A_{ij}, \{B_{ij}, \{C_{ij}, \{\mathrm{D}_{ij}$ are the empirical potential parameters. The two variables $q_i$ and $q_j$ refer to the partial charges in the atoms $i$ and $j$, $\varepsilon$ is the dielectric constant of environment; it is recommended to be $\varepsilon = 2$. The factor (332) in (3) used to describe the energy in kcal/mol. $U_n$ is the energetic torsion barrier of rotation about the bound $n$, and $k_n$ is multiplicity of the torsion angle $\varphi_n$. It is

important to note here that all torsion angles (main chain + side chain) contribute to this formula. In the end, energy function is a function of $N$ torsion angles for a given protein formula.

### 2.3 Harmony search algorithm

Harmony search algorithm (HSA) is a soft computing metaheuristic algorithm inspired by the improvisation process of musicians. In a musical improvisation process, a group of musicians play the pitch of their musical instruments seeking a perfect harmony as estimated by esthetics standard. Similarly, in the optimization context, this process is formulated as follows: a set of decision variables assigned by values seeking for a global optimal solutions as evaluated by an objective function. More details about HSA can be found in (Al-Betar and Khader 2012; Al-Betar et al. 2012b.)

HS procedure has five main steps which will be described as follows:

Step 1. Initialize the problem and HSA parameters:

The optimization problem can be modeled as:$\min f(\boldsymbol{x})$ s.t. $x(i) \in \mathbf{X}(i)$. where $f(\boldsymbol{x})$ is the objective function, $\boldsymbol{x} = \{x(i)|i = 1, \ldots, N\}$ is the set of decision variables, and $N$ is the number of decision variables. $\mathbf{X} = \{\boldsymbol{X}_i|i = 1, \ldots, N\}$ contains all the possible values of each decision variable, i.e., $\mathrm{LB}(i) \leq \boldsymbol{X}(i) \leq \mathrm{UB}(i)$, where $\mathrm{LB}(i)$ and $\mathrm{UB}(i)$ are lower and upper bound values for $x(i)$.

The parameters of the HSA required to solve the optimization problem are also specified in this step:

- Harmony memory size (HMS) which determines the number of initial solutions.
- Harmony memory consideration rate (HMCR) which is used to determine whether the value of a decision variable is to be selected from the accumulative search or randomly from its possible range.
- Pitch adjustment rate (PAR) which decides whether the decision variables are to be adjusted to a neighboring value or not
- Number of improvisations (NI) which is equivalent to the number of iterations in other iterative improvement methods.

Step 2. Initialize the harmony memory.

The harmony memory (HM) is a memory location which stores all the solution vectors determined by HMS. These solution vectors are randomly generated as $x_j(i) = \mathrm{LB}(i) + U(0,1)(\mathrm{UB}(i) - \mathrm{LB}(i)), \forall i \in (1, 2, \ldots, N)$ and $\forall j \in (1, 2, \ldots, \mathrm{HMS})$, where $U(0,1)$ generate a uniform random number between 0 and 1. These vectors will be sorted in ascending order according to their objective function values [see (6]).

$$\mathrm{HM} = \begin{bmatrix} x_1(1) & x_1(2) & \cdots & x_1(N) \\ x_2(1) & x_2(2) & \cdots & x_2(N) \\ \vdots & \vdots & \ddots & \vdots \\ x_{\mathrm{HMS}}(1) & x_{\mathrm{HMS}}(2) & \cdots & x_{\mathrm{HMS}}(N) \end{bmatrix} \quad (6)$$

Step 3. Improvise a new harmony:

In this step, the HSA generates (or *improvises*) a new harmony vector, $\boldsymbol{x}' = (x'_1, x'_2, \cdots, x'_N)$, based on three mechanisms: (1) memory consideration, (2) random consideration, and (3) pitch adjustment.

1. Memory consideration: in memory consideration, the value of the first decision variable $x'(1)$ is randomly assigned from the historical values stored in HM vectors such that $x'(1) \in \{x_1(1), x_2(1), \ldots, x_{\mathrm{HMS}}(1)\}$. Values of the other decision variables, $(x'(2), x'(3), \ldots, x'(N))$, are sequentially assigned in the same manner with probability of HMCR (HMCR $\in (0, 1)$). The operation of this operator is similar to the recombination operator in other population-based methods and is a good source of exploitation (Yang 2009).
2. Random consideration: random consideration is functionally similar to the mutation operator in Genetic Algorithm (Yang 2009); it is a source of global exploration in HSA. In random consideration, the decision variables that are not assigned with values according to memory consideration are assigned with random values using random consideration with a probability of (1-HMCR) according to their possible range, as illustrated in (7).

$$x'(i) \leftarrow \begin{cases} \in \{x_1(i), \ldots, x_{\mathrm{HMS}}(i)\} & U(0,1) \leq \mathrm{HMCR} \\ \in \boldsymbol{x}(i) & \text{otherwise} \end{cases} \quad (7)$$

Note that HMCR and PAR are the main parameters used to control the improvisation process. The HMCR parameter is the probability of assigning one value of a decision variable, $x'_i$, based on historical values stored in the HM. For example, if HMCR = 0.80, this means that the probability of assigning the value of each decision variable from historical values stored in the HM vectors is 80 %, while the probability of assigning the value of each decision variable randomly from its possible value range is 20 %.

3. Pitch adjustment: the value of every decision variable $x'_i$ of a new harmony vector, $\boldsymbol{x}' = (x'_1, x'_2, x'_3, \ldots, x'_N)$, that has been assigned with a value using memory consideration, is examined to determine whether or not it should be pitch adjusted with the probability of PAR ($0 \leq \{\mathrm{PAR} \leq 1$) as follows:

$$\text{Adjust } x'(i)? \leftarrow \begin{cases} \text{Yes} & U(0,1) \leq \mathrm{PAR} \\ \text{No} & \text{otherwise} \end{cases} \quad (8)$$

If the pitch adjustment decision for $x'(i)$ is Yes, the value of $x'(i)$ is adjusted to its neighboring value as follows:

$$x'(i) = x'(i) \pm U(0,1) \times \text{BW} \quad (9)$$

where BW is a parameter (distance bandwidth) for continuous optimization problems (e.g., PPSP) which normally takes a value in advance and remains constant during the search. The BW = 0.01 is recommended (Omran and Mahdavi 2008).

Step 4. Update the harmony memory:

If the new harmony vector, $\boldsymbol{x}' = (x'(1), x'(2), \ldots, x'(N))$, is better than the worst harmony vector in harmony memory, the new harmony vector replaces the worst harmony vector.

Step 5. Check the stop criterion:

HS algorithm will repeat steps 3 and 4 until maximum number of improvisations determined by NI is met.

Algorithm 1 describes the pseudo-code of HS algorithm.

**Algorithm 1** General pseudo code for Harmony search Algorithm

```
Set HMCR, PAR, NI, HMS, BW.
x_j(i) = LB(i)+(UB(i)−LB(i)×U(0,1), ∀i = 1,2,...,N and ∀j = 1,2,...,HMS {generate HM solutions}
Calculate(f(x_j)), ∀j = (1,2,...,HMS)
Sort(HM)
itr = 0
while (itr ≤ NI) do
  x' = φ
  for i = 1,···,N do
    if (U(0,1) ≤ HMCR) then
      x'(i) ∈ {x_1(i), x_2(i),..., x_HMS(i)} {memory consideration}
      if (U(0,1) ≤ PAR) then
        x'(i) = x'(i) ± U(0,1) × BW { pitch adjustment }
      end if
    else
      x'(i) = LB(i) + (UB(i) − LB(i) × U(0,1) { random consideration }
    end if
  end for
  if (f(x') < f(x^worst)) then
    Include x' to the HM.
    Exclude x^worst from HM.
  end if
  itr = itr + 1
end while
```

## 3 Method

This section provides a description for adapting harmony search algorithm (AHSA) and how the HMCR and PAR parameters are iteratively updated. Thereafter, The way of improving the AHSA is presented by proposing HHSA that incorporates Iterative local search and global best concept of PSO in AHSA.

### 3.1 AHSA for ab initio PSPP

The protein structure is initialized for the harmony search optimization as follows: For a particular protein sequence which is picked from the protein data bank, some parameters are extracted from the data base. These parameters comprise the number of amino acids ($M$) and the number of torsion angles ($N$). For example, for the two proteins experimented in this research; the 'Met-enkephalin' has 5 amino acids and 24 torsion angles, while '1CRN' has 46 amino acids and 238 torsion angles. The solution is then represented as a vector of amino acids, $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_M)$, where $\boldsymbol{x}_1 = (\Phi_1(1), \Psi_1(1), \omega_1(1), x_1(1), x_1(2), \ldots, x_1(N)), \boldsymbol{x}_2 = (\Phi_2(1), \Psi_2(1), \omega_2(1), x_2(1), x_2(2), \ldots, x_2(N)), \ldots, \boldsymbol{x}_M = (\Phi_M(1), \Psi_M(1), \omega_M(1), x_M(1), x_M(2), \ldots, x_M(N))$.However, AHSA deals with this vector as a vector of torsion angles $\mathbf{x}_i = (x(1), x(2), \ldots, x(N))$, where $N$ is the number of torsion angles in the protein, and each angle ($x(i)$) in this vector can be assigned with a value within the range $[-\pi, \pi]$. Therefore, the solution length is equal to the number of torsion angles in the protein.

The Harmony Memory (HM) is initialized with random vectors as determined by HMS. A different random seed is used to generate torsion angles randomly within the range $[-\pi, \pi]$. The objective function $f(\boldsymbol{x})$ in (1) is utilized to calculate the energy value for each vector of torsion angles in HM. The vectors (solutions) in HM are sorted in ascending order based on their energy values, such as $f(\boldsymbol{x}_1) \leq f(\boldsymbol{x}_2) \leq \ldots \leq f(\boldsymbol{x}_{\text{HMS}})$. In improvising a new harmony step, the AHSA generates a new harmony vector, $\boldsymbol{x}' = (x'(1), x'(2), \ldots, x'(N))$, based on three operators discussed in Sect. 2.3: (1) memory consideration, (2) random consideration, and (3) pitch adjustment.

If the new torsion angles vector, $\boldsymbol{x}' = (x'(1), x'(2), \ldots, x'(N))$, has better energy than the worst harmony vector in HM, the new vector replaces the worst one in HM. This process is repeated until the maximum number of improvisations is met. At the end of the improvisations, the AHSA passes the torsion angles of the best vector to a procedure to represent the structure of this vector that will be stored in a protein structure data base.

It has to be noted that the HSA iterates toward the optimal solution using two main parameters, HMCR and PAR. The optimization process should consider the balance between *exploration* and *exploitation* concepts; memory consideration is the source of exploitation in HSA while the random consideration is the source of exploration. Increasing HMCR leads the search to tend toward exploitation (Yang 2009). On the other hand, the pitch adjustment operator performs a set of random local changes for the torsion angles that are assigned by values based on memory consideration. Therefore, the higher the PAR value is, the more the search tends toward the exploration.

In optimization, previous theories indicated that the search should concentrate on the exploration in the early stage of search, while at the final stage, it should concentrate on the exploitation (Blum and Roli 2003). This is the very same idea upon which simulated annealing

metaheuristic algorithm is built. In the searching process, the simulated annealing algorithm accepts not only better but also worse neighboring solutions with a certain probability based on two factors: the energy value of the current solution and the temperature (Wang et al. 2001). During the search, simulated annealing reduces the chance of accepting the worse solution until reaching the final stage of search by reducing the value of temperature parameter linearly. In the final stage of search, simulated annealing concentrates on the search space of current solution by means of accepting only the downhill (or better) moves. This idea of simulated annealing is utilized in the proposed AHSA.

In HSA shown in Sect. 2.3, the HMCR and PAR are assigned their values in advance while remaining constant during the search process. However, HMCR and PAR help the algorithm find globally and locally improved solutions, respectively (Lee and Geem 2005). After applying the AHSA to protein structure prediction problem, and after testing different values of HMCR and PAR, it has been observed that when the value of HMCR is high (i.e., 0.99), the AHSA obtains good results and fast convergence rate, because the probability to select the new value from the harmony memory (which has already improved values) will be high. However, this will cause the HSA, most often, to get stuck in local optima because the power of exploration is low. On the other hand, a smaller HMCR value can avoid a premature convergence, but the search will be slow. Moreover, for PAR parameter, it is clear that when the PAR value is high, the value selected from harmony memory will be adjusted with greater chance; while when the PAR value is less, the probability to change this value will be less. Therefore, when the PAR value is small, the results will be better but will also cause the HSA to get stuck in local optima. Moreover, if the value of HMCR is small (i.e., 0.50) and the value of PAR is high (i.e., 0.50), the HSA will not obtain good results.

These observations lead this research to propose a new mechanism for controlling the values of HMCR and PAR. The new mechanism is to update the values of HMCR and PAR dynamically during the search process, rather than fix them in the initial step. After a series of experiments, this study has identified the following assumptions:

- Assigning a small value to HMCR and a high value to PAR in the first stage of the search increases the power of exploration of the search space and increases diversity of the solutions in HM.
- Increasing the value of HMCR and decreasing the value of PAR gradually during the search process increases the exploitation. Note that exploration is useful in the first stage of search, while exploitation is more useful in the final stage of search.

The AHSA proposes using two values of HMCR; $\text{HMCR}_{\text{min}}$ (minimum value of HMCR that the AHSA starts with) and $\text{HMCR}_{\text{max}}$ (maximum value of HMCR that the AHSA ends with), and two values of PAR; $\text{PAR}_{\text{min}}$ (minimum value of PAR that the AHSA ends with) and $\text{PAR}_{\text{max}}$ (maximum value of PAR that the AHSA starts with). However, all these four parameters are initialized in the first step of the AHSA.

Then, the AHSA updates the values of HMCR and PAR dynamically during the search process as follows.

$$\text{HMCR}_{i+1} = \text{HMCR}_i + (1 - \text{HMCR}_i) \times \left(1 - e^{\left(-\frac{|(f(x_{\text{best}})|}{t_i}\right)}\right) \tag{10}$$

$$\text{PAR}_{i+1} = \text{PAR}_i \times e^{\left(-\frac{|(f(x_{\text{best}})|}{t_i}\right)} \tag{11}$$

where $i$ is the iteration number, $\boldsymbol{x}_{\text{best}}$ is the best solution in HM that has the lowest energy value (lowest is best), $t_i$ is a variable equivalent to the temperature in simulated annealing; this variable is reduced linearly by a control variable, $\alpha$, such that:

$$t_{i+1} = \alpha t_i$$

where $\alpha$ is a control parameter small but close to 1.

The two previous equations change the values of both HMCR and PAR slowly because they use the best energy value (which is decreased by iterations) in the exponential function. This means that the value of exponential function depends on the best energy of the previous iteration. Therefore, if the value of the best energy is high, the value of the exponential function will be low which means faster change in PAR parameter. When the search proceeds, the energy value will decrease which leads to slower decrease in PAR. However, for HMCR, the change will be slow because the equation uses 1-HMCR to multiply the exponential function. It is important to highlight that the energy starts with very high values for long protein and low values for the short protein. Therefore, to guarantee a slow change in both HMCR, and PAR values, we need to adjust the value of $t_i$ in the denominator of the fraction; the value of $t_i$ is set to a high value for long protein and a smaller value for short protein. This also applies to the value of $\alpha$, where selecting this value to be a high fraction (close to 1) leads to a slow decrease in $t_i$, thus, a slow change in both PAR and HMCR.

The above two equations, (10) and (11), are derived based on the experiments and after observing the change of PAR and HMCR values during the optimization process. This way of updating the values of HMCR and PAR throughout the search process is inspired by the Monte Carlo acceptance rule of simulated annealing approach (Kirkpatrick et al. 1983).

Applying the previous way of controlling the values of PAR and HMCR dynamically enables the AHSA to

concentrate more on the exploration in the early stages of search by assigning a large value for PAR and a small value of HMCR. Then, during the search, the value of HMCR will be increased exponentially while the value of PAR will be decreased exponentially leading the search to concentrate on exploitation.

In (10), the value of $\text{HMCR}_{\text{min}}$ is increased exponentially with iteration number until it reaches the value $\text{HMCR}_{\text{max}}$ and in (11), the value $\text{PAR}_{\text{max}}$ is decreased exponentially every iteration until it reaches the value $\text{PAR}_{\text{min}}$. Algorithm 2 shows the pseudo code of AHSA steps.

**Algorithm 2** The Adaptive Harmony Search Algorithm

**STEP1 Initialize AHSA and PSPP**

1: Set the AHSA parameters (HMCR, PAR, NI, and HMS).
2: Extract the number of amino acids ($M$) and the number of torsion angles ($N$) for the protein sequence

**STEP2 Initialize the harmony memory**

1: Construct vectors of torsion angles randomly and store them in HM, $\mathbf{HM} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{\text{HMS}}\}$
2: Recognize the vector with worst energy in **HM**, $\boldsymbol{x}_{\text{worst}} \in \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{\text{HMS}}\}$

**STEP3 Improvise a new harmony**

1: $\boldsymbol{x}' = \phi$ // a new torsion angles vector
2: **for** $i = 1, \cdots, N$ **do**
3: **if** $(U(0,1) \leq \text{HMCR})$ **then**
4: $x'(i) \in \{x_1(i), x_2(i), \ldots, x_{\text{HMS}}(i)\}$ {memory consideration}
5: **if** $(U(0,1) \leq \text{PAR})$ **then**
6: $x'(i) = x'(i) \pm U(0,1) \times \text{BW}$ { pitch adjustment }
7: **end if**
8: **else**
9: $x'(i) \in \boldsymbol{X}_i$ { random consideration }
10: **end if**
11: **end for**

**STEP4 Update the harmony memory**

1: **if** $(f(\boldsymbol{x}') < f(\boldsymbol{x}_{\text{worst}}))$ **then**
2: Include $\boldsymbol{x}'$ to the **HM**.
3: Exclude $\boldsymbol{x}_{\text{worst}}$ from **HM**.
4: **end if**

**STEP5 Update HMCR and PAR values**

1: $\text{HMCR}_{i+1} = \text{HMCR}_i + (1 - \text{HMCR}_i) \times \left(1 - e^{\left(-\frac{|(f(\boldsymbol{x}_{\text{best}})|}{t_i}\right)}\right)$
2: $\text{PAR}_{i+1} = \text{PAR}_i \times e^{\left(-\frac{|(f(\boldsymbol{x}_{\text{best}})|}{t_i}\right)}$

**STEP6 Check the stop criterion**

1: **while** (Number of improvisations is less than NI) **do**
2: Repeat **STEP3** to **STEP5**
3: **end while**

**STEP7 Output the torsion angles of the best vector and represent the protein structure**

## 3.2 HHSA for ab initio PSPP

In the AHSA, the memory consideration is responsible for the global improvement. The new harmony can be improved by focusing on the good solutions stored in Harmony Memory by means of the natural selection principle of the 'survival of the fittest'. Additionally, the source of local improvement in HSA is the pitch adjustment operator that adjusts the value of the variables (i.e., torsion angles) to their neighboring values locally, with the hope to affect the energy function positively. Recall, the pitch adjustment performs a random adjustment that is not guided by energy function. Due to the complex nature of PSPP, the random adjustment cannot be guaranteed to improve the new harmony solution to reach the local optimal solution. This shortcoming in the AHSA has led this research to propose hybridizing a local search method within the AHSA, which is a population-based method. The observations of some previous studies have revealed much interest in hybridizing a local search-based algorithm within the population-based algorithm (Blum and Roli 2003).

Therefore, a HHSA is proposed in this research with two modifications to the AHSA; first, the ILS is incorporated with the AHSA to help find the local optimal solution within the search space of the new harmony. Second, the global-best concept of PSO is incorporated with memory consideration as a selection concept to select the values from the best vector in the harmony memory instead of random selection. These two concepts, ILS and global best of PSO, have enhanced the local exploitation capability and the speed of convergence of AHSA, respectively.

The process of hybridizing the ILS with the AHSA is explained in Sect. 3.2.1 while the global-best memory consideration is explained in Sect. 3.2.2.

### 3.2.1 *Hybridizing with iterated local search*

Algorithm 3 illustrates how the ILS is incorporated within the AHSA; the ILS is called after the improvisation step of the adaptive HSA and works as a new operator to fine-tune the new harmony in each improvisation to the local optimal solution in the region to which the HSA converges. The initial solution for ILS is the new harmony solution $\boldsymbol{x}'$ generated by the adaptive HSA operators (i.e., memory consideration, random consideration and pitch adjustment).

**Algorithm 3** The HHSA calling ILS Function

1: Initialize(HM)
2: **while** (No termination criterion is met) **do**
3: $\boldsymbol{x}' = \phi$ // a new torsion angles vector
4: Improvise a new harmony solution ($\boldsymbol{x}'$) {Algorithm 2}
5: Update values of HMCR and PAR {based on Eq. (10) and Eq. (11)}
6: Iterated Local Search ($\boldsymbol{x}'$ )
7: Update(HM)
8: **end while**

The pseudo-code of the ILS function is described in Algorithm 4. During the improvement loop in Line 2, the function $\textit{Explore}(\mathcal{N}(\boldsymbol{x}'))$ navigates the neighboring solutions $\mathcal{N}(\boldsymbol{x}')$ of $\boldsymbol{x}'$, and moves to the first neighboring

solution $\boldsymbol{x}'' \in \mathcal{N}(\boldsymbol{x}')$ which has an equal or lower energy value, i.e, $f(\boldsymbol{x}'') \leq f(\boldsymbol{x}')$. This process is repeated until no further improvement is obtained.

The function $Explore(\mathcal{N}(\boldsymbol{x}'))$ explores the search space of $\boldsymbol{x}'$ using the following criteria:

$$x'(i) = x'(i) \pm \alpha, \quad \forall i \in (1, 2, \ldots, N)$$

where $\alpha$ = the value of bw $\times$ $U\sim(0,\ 1)$, bw is an arbitrary distance bandwidth for the continuous variable; and $U\sim(0,1)$ generates a uniform distribution number between 0 and 1. Note that bw takes a static value within the range $[-\pi,\ \pi]$. This is how the ILS provides a guided method for local exploitation.

**Algorithm 4** ILS Function

```
Input(x') {Initial solution}
while ( Local optimal solution is not reached) do
    x'' = Explore(N(x'))
    if (f(x'') ≤ f(x')) then
        x' = x''
    end if
end while
```

### 3.2.2 *Hybridizing with global-best memory consideration*

The HHSA has incorporated a new parameter called particle swarm rate (PSR) to control the global best memory consideration (GBMC) proposed in HHSA. The GBMC has enhanced the way of selecting the torsion angles values from the solutions stored in HM, and subsequently, enhancing the performance of the HHSA. During the optimization process of HHSA, the new value is selected from either the memory with a probability of HMCR, or randomly with a probability of 1-HMCR. The new parameter PSR is used within the probability of selecting the new value from the harmony memory (i.e., within HMCR). So, instead of picking the value randomly from any vector in the harmony memory, as usual in the classical harmony search, it will be rather selected either from any vector in the harmony memory with a probability of PSR or from the best solution in harmony memory with a probability of (1-PSR); Algorithm 5 describes this procedure within the memory consideration operation of the improvisation step. This new operation has increased the capability of HHSA in terms of the convergence rate.

**Algorithm 5** Global-best incorporated with Memory Consideration Process

```
if (U(0,1) ≤ HMCR) then
    if (U(0,1) ≤ PSR) then
        x'(i) ∈ {x1(i), x2(i), ..., xHMS(i)} {memory consideration}
    else
        x'(i) ∈ xBEST(k), ∀k ∈ (1, 2, ..., N) { select the value from the best vector }
    end if
end if
```

## 4 Results and discussion

### 4.1 Benchmark

The AHSA and HHSA have been evaluated using two protein sequences publicly available at 'www.pdb.org'; the first protein sequence is called 'Met-enkephalin' (or 1 PLW as in PDB). It has been first identified from the enkephalin mixture of brains,and is involved in a variety of physiological processes. This sequence is one of the most used model peptides; it has a short residue sequence of five amino acids: Tyr1-Gly2-Gly3-Phe4-Met5 consists of 75 atoms described by 24 independent backbone and side chain dihedral angles. It is, therefore, used to evaluate many state-of-the-art methods to which the three algorithms proposed in this thesis will be compared. Although it is a small peptide, Met-enkephalin needs a complex conformational space with a total number of more than $10^{11}$ local minima (Li and Scheraga 1988). Met-enkephalin has been extensively studied computationally and has been regarded as a benchmark model,and has also been used frequently for testing simulation methods in recent years because of the complexity in its configuration space and the short sequence allowing significant computational studies (Zhan et al. 2006).

The second sequence, namely '1CRN', is a plant seed protein which has 46 amino acids consisting of 238 angles.

### 4.2 Experimental design

A series of experiments have been conducted to measure the influence of different parameters on the performance of the proposed HHSA. Twelve cases of different parameter settings have been selected, as indicated in Table 1. Each case has been repeated for 30 runs for both benchmark protein sequences. The first six cases have smaller HMS values than the last 6. This is to show the impact of HMS on the behavior of HHSA.

The $\text{HMCR}_{\text{min}}$ is selected in different cases with large (i.e., 0.90) and small (i.e., 0.50) values. Recall, the HMCR is the rate of selecting the solution from the harmony memory. The more the value of HMCR is, the more the exploitation increases and the exploration decreases. When $\text{HMCR}_{\text{min}}$ has a small value, the algorithm considered the exploration with minimal exploitation at the initial stage of search. In contrast, large $\text{HMCR}_{\text{min}}$ leads to better exploitation in the initial stage of search.

PAR is the rate of using pitch adjustment operator. The values of $\text{PAR}_{\text{min}}$ and $\text{PAR}_{\text{max}}$ are fixed throughout the whole convergence cases. The $\text{PAR}_{\text{max}}$ initially takes a high value and is decreased exponentially to reach a high exploitation power at the final stage of search. $\text{PAR}_{\text{min}}$ is fixed to 0.05 and $\text{PAR}_{\text{max}}$ is fixed to 0.25 for all the 12 convergence cases.

**Table 1** Cases used to evaluate the HHSA convergence ability

| Cases | HMS | $HMCR_{min}$ | PSR |
|---|---|---|---|
| Case 1 | 10 | 0.50 | 0.10 |
| Case 2 | 10 | 0.50 | 0.50 |
| Case 3 | 10 | 0.50 | 0.90 |
| Case 4 | 10 | 0.90 | 0.10 |
| Case 5 | 10 | 0.90 | 0.50 |
| Case 6 | 10 | 0.90 | 0.90 |
| Case 7 | 50 | 0.50 | 0.10 |
| Case 8 | 50 | 0.50 | 0.50 |
| Case 9 | 50 | 0.50 | 0.90 |
| Case 10 | 50 | 0.90 | 0.10 |
| Case 11 | 50 | 0.90 | 0.50 |
| Case 12 | 50 | 0.90 | 0.90 |

The PSR is the rate of selecting the value of the angles either from the best vector having the lowest energy value in harmony memory, or randomly from any vector in harmony memory. Three different values of PSR have been experimented with: low (i.e., 0.10), medium (i.e., 0.50), and high (i.e., 0.90). The PSR of value = 0.10 means that the memory consideration will select the value of an angle randomly from any solution in harmony memory with a probability of 10 %, and from the best vector in harmony memory with a probability of 90 %.

As for the number of improvisations (NI), it is fixed in all the 12 cases to NI = 100,000 and $HMCR_{max} = 0.99$.

### 4.3 Experimental results

Table 2 shows the best, average, worst, and standard deviation of the energy values obtained of both 'Met-enkephalin' and '1CRN' for the 12 cases after 30 runs for each case. The best results are highlighted in bold while the best averages are highlighted in italic.

The boxplots in Figs. 1 and 2 visualize the distribution of the energy values obtained in the 30 runs of the 12 convergence cases for both 'Met-enkephalin' and '1CRN', respectively. The boxplot summarizes the following statistical measures: median, upper and lower quartiles, and minimum and maximum energy values. In the boxplot, the larger the range of the first and third quartile are, the worse the searching accuracy of the convergence case is.

### 4.4 Discussion

The experimental results in Table 2 and the Boxplots in Figs. 1 and 2 describe the behavior of the HHSA during the search process for all the 12 convergence cases. This section provides a discussion of the behavior of the HHSA based on Table 2 and Figs. 1, 2.

Cases 7–12 are similar to cases 1–6 with different HMS values; while the HMS in the first six cases is set to a small value (i.e 10), it is set to 50 in the last six cases; this is meant to show the impact of HMS on the behavior of the algorithm. In general, the cases with HMS = 10 have obtained better results than those with HMS = 50 while fixing the other parameters. The experiments also show that for the short protein, 'Met-enkephalin', the cases with PSR = 0.10 have recorded better results than those with PSR = 0.50 or 0.90. Whereas, for the longer protein, '1CRN', the cases with PSR = 0.90 have recorded better results than those with PSR = 0.10 or 0.50. Recall, PSR = 0.10 means that the torsion angle is selected from the best solution in harmony memory with a probability of 90 %.

Both Table 2 and the Boxplot in Fig. 1 indicate that cases 4 and 5 have obtained the best results for the 'Met-enkephalin' protein. Although cases 1, 7, and 10 have

**Table 2** Results of HHSA for 30 runs of the 12 cases

| Cases | Met-enkephalin | | | | 1CRN | | | |
|---|---|---|---|---|---|---|---|---|
| | Best | Avg. | Worst | SD | Best | Avg. | Worst | SD |
| Case 1 | **−12.43** | −11.00 | −10.12 | 0.40 | −139.08 | −109.41 | −47.30 | 27.71 |
| Case 2 | −11.27 | −10.94 | −10.12 | 0.32 | −176.29 | −140.71 | −118.37 | 14.58 |
| Case 3 | −11.27 | −10.89 | −10.58 | 0.25 | −161.39 | −146.31 | −130.48 | 7.28 |
| Case 4 | **−12.43** | −11.28 | −9.86 | 0.78 | −164.33 | −110.82 | 131.29 | 52.23 |
| Case 5 | **−12.43** | *−11.41* | −10.60 | 0.50 | −201.56 | −139.15 | −102.30 | 16.68 |
| Case 6 | −12.41 | −10.87 | −10.11 | 0.44 | **−204.81** | *−156.83* | −141.42 | 16.85 |
| Case 7 | **−12.43** | −11.02 | −10.60 | 0.43 | −149.46 | −124.95 | −91.16 | 13.49 |
| Case 8 | −12.38 | −11.12 | −10.57 | 0.49 | −159.41 | −140.31 | −105.15 | 10.82 |
| Case 9 | −11.46 | −10.73 | −10.45 | 0.32 | −147.76 | −136.17 | −125.35 | 5.83 |
| Case 10 | **−12.43** | −11.21 | −10.12 | 0.62 | −138.28 | −116.09 | −81.60 | 15.64 |
| Case 11 | −12.42 | −10.99 | −10.11 | 0.48 | −153.67 | −138.03 | −113.91 | 9.50 |
| Case 12 | −12.25 | −10.79 | −10.52 | 0.41 | −149.63 | −138.74 | −128.18 | 6.02 |

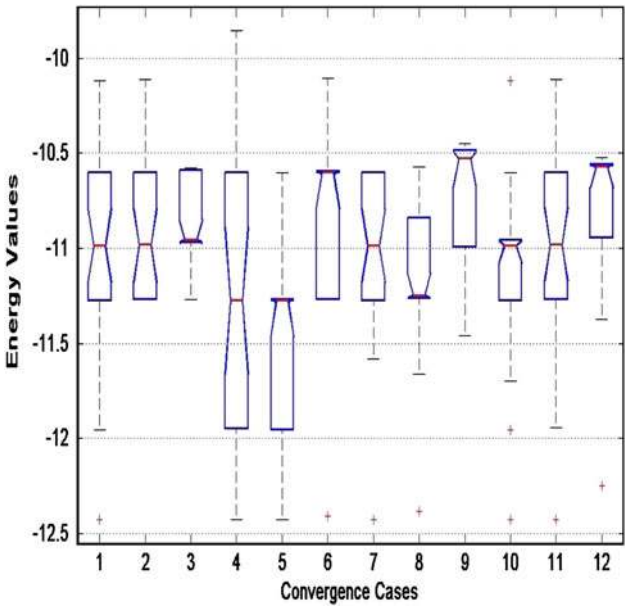



**Fig. 1** Boxplot showing the distribution of the results for 30 experiments done for each convergence case for Met-enkephalin

obtained the lowest energy recorded for the Met-enkephalin, the boxplot indicates that cases 4 and 5 are better because most of the energy values are within the range −11.3 and −12, and there are not many extreme values far from the average value; Besides, both cases 4 and 5 have obtained better averages with an advantage of case 5 over case 4. For '1CRN', both Table 2 and the Boxplot in Fig. 2 indicate that cases 5 and 6 have obtained the best energy values with an advantage of case 6 over case 5; the distribution of case 6 is better than case 5 because most of the energy values are within the range −140 and −170; besides, the average of case 6 is better than the average of case 5.

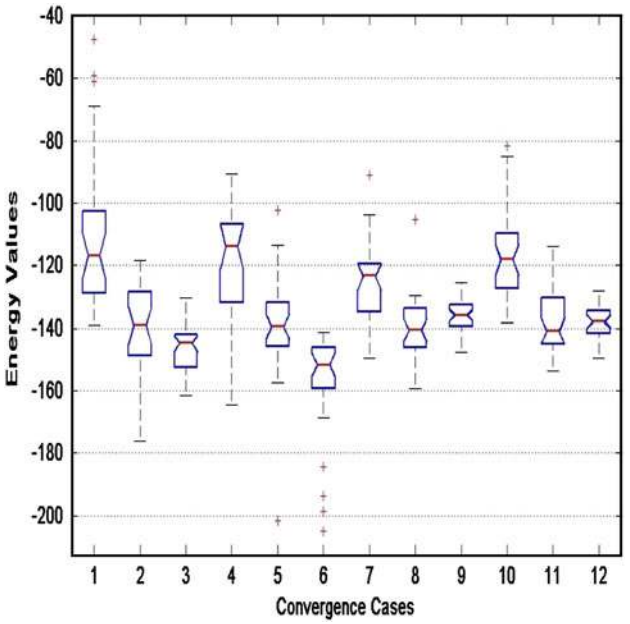



**Fig. 2** Boxplot showing the distribution of the results for 30 experiments done for each convergence case for 1CRN

Figures 3 and 4 show the convergence behavior of the HHSA of the 12 convergence cases for 'Met-enkephalin' during the first 10,000 improvisations. Note that the energy values of each case are obtained using a random run out of the 30 runs. *X* axis shows the number of iterations, *Y* axis shows the energy values of each case, and the trends represent the experimented cases. It is apparent that the convergence rate is very fast in the initial iterations as the trend is reduced sharply for almost all cases. However, the convergence rate is gradually reduced throughout the 10,000 improvisations until the equilibrium state is reached. Note that the cases 6, 10, and 11 have obtained the most desired results, and they have almost converged to a similar point in the plot.

Figures 5 and 6 show the convergence behavior of the HHSA of the 12 convergence cases for the longer protein, '1CRN', during the first 10,000 improvisations. The convergence rate is very fast in the initial iterations because the initial energy values are obtained randomly. However, the convergence rate is gradually reduced in the early improvisations until the equilibrium state is reached. For this long protein, cases 5 and 6 have converged faster to the lowest energy values and that is because they have high HMCR values (i.e., 0.90). Comparing cases 5 and 6 with cases 11 and 12 shows the impact of HMS on the behavior of the HHSA; although cases 5 and 6 differ from cases 11 and 12 in HMS only (while they have same HMCR and PSR values), cases 5 and 6 have converged faster to the lowest energy than cases 11 and 12, respectively. This proves that selecting a smaller value of HMS has obtained better results than a larger HMS.

## 5 Comparative results

### 5.1 Comparison between AHSA and HHSA

This section provides a comparison between the results of AHSA and HHSA. Cases 3, 4, 7, and 8 of AHSA and cases 4, 5, 10, and 11 of HHSA are used in this comparison. These cases are chosen because they have almost the same parameter design. In Table 3, the best, average, worst, and standard deviation of the energy values are recorded for the 30 runs of every case. The best energy obtained is highlighted in bold.

It is clear from Table 3 that only case 3 of AHSA is able to obtain the best energy value for the small protein, while in HHSA, cases 4, 5, and 10 are able to obtain the best

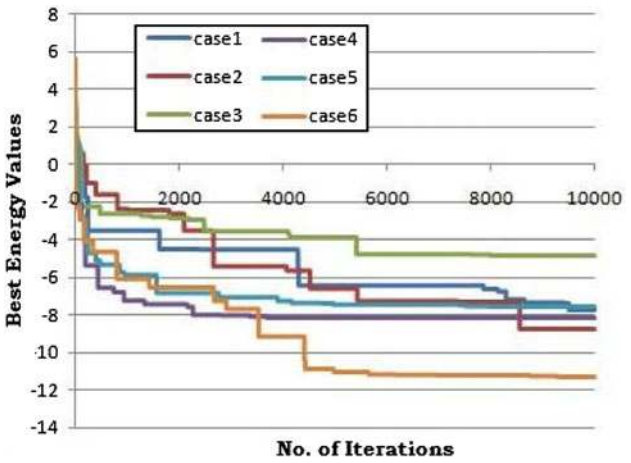



**Fig. 3** The best energy values against the number of iterations for the first six convergence cases of Met-enkephalin

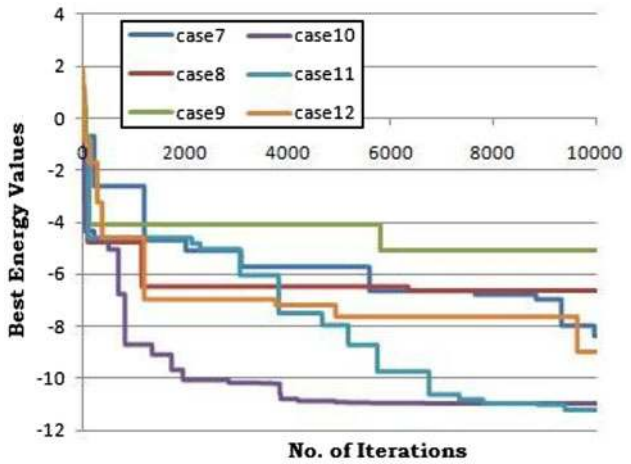



**Fig. 4** The best energy values against the number of iterations for the last six convergence cases of Met-enkephalin

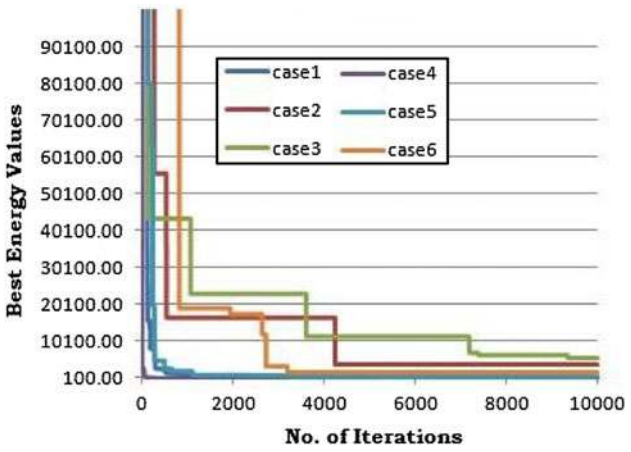



**Fig. 5** The best energy values against the number of iterations for the first six convergence cases of 1CRN

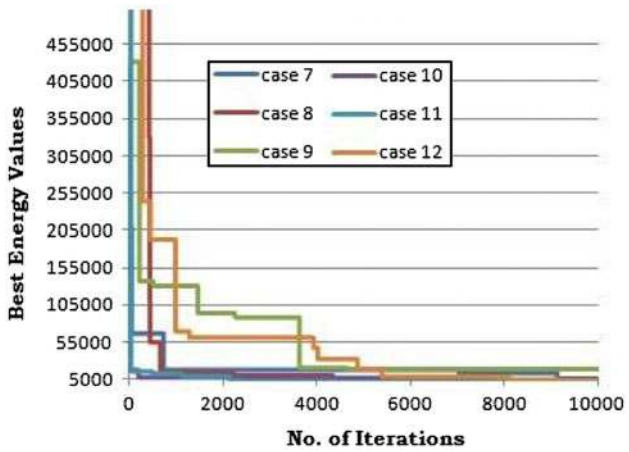



**Fig. 6** The best energy values against the number of iterations for the last six convergence cases of 1CRN

energy value, and case 11 is also very close to best energy value. However, for the longer protein, '1CRN', all the four cases of HHSA have obtained better energy values with comparison to those of AHSA.

Moreover, a *t* test has been performed to find out whether the results of the adaptive HSA and HHSA are significantly different or not. Tables 4 and 5 summarize the *p* values of the *t* test for 'Met-enkephalin' and '1CRN', respectively. It is clear that for both the 'Met-enkephalin' and '1CRN' proteins, the results of the HHSA and the AHSA approaches are statistically different in favor of HHSA.

## 5.2 Comparison between HHSA and other studies

Tables 6 and 7 show the results of HHSA in comparison with some previous studies. The proposed HHSA is able to record the best results obtained until now which are $E = -12.43$ by Zhan et al. (2006) based on ECEPP/3 force field and $E = -12.91$ by Zhan et al. (2006), Meirovitch et al. (1994), and Li and Scheraga (1987) based on ECEPP/2 force field. Moreover, two new global optima energy values of the Met-enkephalin protein has been recorded by HHSA based on ECEPP/3 and ECEPP/2 force fields with $\omega = 180°$; the current energy optimum based on ECEPP/3 force field with $\omega = 180°$ is $E = -10.90$ kcal/mol by Zhan et al. (2006) while HHSA records a new energy value, $E = -11.26$ kcal/mol. Furthermore, based on ECEPP/2 force field with $\omega = 180°$, the current lowest energy is $E = -10.72$ kcal/mol, while the HHSA obtains $E = -11.57$ kcal/mol.

Tables 6 and 7 show the results of HHSA in comparison with some previous studies. The proposed HHSA is able to record the best results obtained until now which are $E = -12.43$ by Zhan et al. (2006) based on ECEPP/3 force field and $E = -12.91$ by Zhan et al (2006), Meirovitch et al

**Table 3** Comparison results between AHSA and HHSA for the two proteins

| Protein | AHSA | | | | HHSA | | | |
|---|---|---|---|---|---|---|---|---|
| | Case 3 | Case 4 | Case 7 | Case 8 | Case 4 | Case 5 | Case 10 | Case 11 |
| Met-enkephalin | | | | | | | | |
| Best | **−12.43** | −11.52 | −11.17 | −4.13 | **−12.43** | **−12.43** | **−12.43** | −12.42 |
| Average | −9.1 | −4.5 | −7.7 | −2.3 | −11.3 | −11.4 | −11.2 | −11.0 |
| Worst | −7.2 | −2.6 | −6.2 | −1.3 | −9.9 | −10.6 | −10.1 | −10.1 |
| SD | 1.4 | 0.1 | 1.1 | 0.7 | 0.8 | 0.5 | 0.6 | 0.5 |
| 1CRN | | | | | | | | |
| Best | −155.6 | −184.9 | −134.1 | −134.5 | −164.3 | **−201.6** | −138.3 | −153.7 |
| Average | −115.8 | −130.3 | −54.5 | 281.4 | −110.8 | −139.2 | −116.1 | −138.0 |
| Worst | 89.9 | −52.6 | 518.3 | 8829.9 | 131.3 | −102.3 | −81.6 | −113.9 |
| SD | 56.9 | 31.9 | 129.1 | 1645.1 | 52.2 | 16.7 | 15.6 | 9.5 |

**Table 4** Independent samples test for 'Met-enkephalin'

| | Levene's test for equality of variances | | *t* test for equality of means | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | *F* | Sig. | *t* | *df* | Sig. (2-tailed) | Mean difference | Std. error difference | 95 % confidence interval of the difference | |
| | | | | | | | | Lower | Upper |
| Equal variances assumed | 2.183 | 0.145 | −9.123 | 58 | 8.32E−13 | −2.484 | 0.2723 | −3.029 | −1.939 |
| Equal variances not assumed | | | −9.123 | 47.561 | 8.32E−13 | −2.484 | 0.2723 | −3.032 | −1.937 |

**Table 5** Independent samples test for '1CRN'

| | Levene's test for equality of variances | | *t* test for equality of means | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | *F* | Sig. | *t* | *df* | Sig. (2-tailed) | Mean difference | Std. error difference | 95 % confidence interval of the difference | |
| | | | | | | | | Lower | Upper |
| Equal variances assumed | 7.804 | 0.007 | −3.55 | 58 | 7.72E-04 | −10.196 | 2.87218 | −15.945 | −4.4464 |
| Equal variances not assumed | | | −3.55 | 46.208 | 7.72E-04 | −10.196 | 2.87218 | −15.976 | −4.4149 |

(1994), and Li and Scheraga (1987) based on ECEPP/2 force field. Moreover, two new global optima energy values of the Met-enkephalin protein has been recorded by HHSA based on ECEPP/3 and ECEPP/2 force fields with $\omega = 180°$; the current energy optimum based on ECEPP/3 force field with $\omega = 180°$ is $E = -10.90$ kcal/mol by Zhan et al. (2006) while HHSA records a new energy value, $E = -11.26$ kcal/mol. Furthermore, based on ECEPP/2 force field with $\omega = 180°$, the current lowest energy is $E = -10.72$ kcal/mol, while the HHSA obtains $E = -11.57$ kcal/mol.

The barcharts in Figs. 7 and 8 describe the best energies obtained by HHSA based on ECEPP/3 with $\omega$ relaxed, and ECEPP/3 with $\omega = 180°$, respectively. With comparison to other studies, the barcharts indicate that HHSA obtains the same optimal energy recorded by Zhan et al. (2006), while it outperforms others. Moreover, based on ECEPP/3 with $\omega = 180°$, HHSA obtains a new energy value and outperforms the previous lowest energies obtained by Zhan et al. (2006) and Eisenmenger and Hansmann (1997).

The barcharts in Figs. 9 and 10 describe the best energies obtained by HHSA based on ECEPP/2 with $\omega$ relaxed,

**Table 6** The lowest energies of Met-enkephalin (in kcal/mol) obtained by HHSA compared with previous studies based on ECEPP/3 force fields

| Source | Force field | Energy |
|---|---|---|
| Androulakis et al. (1997) | ECEPP/3 | −11.70 |
| Zhan et al. (2006) | ECEPP/3 | −12.43 |
| HHSA | ECEPP/3 | −12.43 |
| Eisenmenger and Hansmann (1997) | ECEPP/3 $\omega = 180°$ | −10.85 |
| Lixin Zhan (2006) | ECEPP/3 $\omega = 180°$ | −10.90 |
| HHSA | ECEPP/3 $\omega = 180°$ | −11.26 |

**Table 7** The lowest energies of Met-enkephalin (in kcal/mol) obtained by HHSA compared with previous studies based on ECEPP/2 force field

| Source | Force field | Energy |
|---|---|---|
| Meirovitch et al (1994) | ECEPP/2 | −12.91 |
| Lixin Zhan (2006) | ECEPP/2 | −12.91 |
| HHSA | ECEPP/2 | −12.91 |
| Lixin Zhan (2006) | ECEPP/2 $\omega = 180°$ | −10.72 |
| Eisenmenger and Hansmann (1997) | ECEPP/2 $\omega = 180°$ | −10.72 |
| HHSA | ECEPP/2 $\omega = 180°$ | −11.57 |

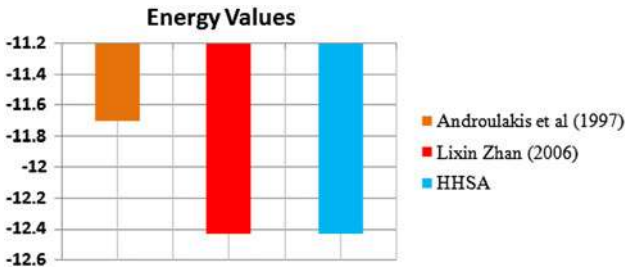



**Fig. 7** Barchart showing the results obtained for Met-enkephalin using ECEPP/3 force field

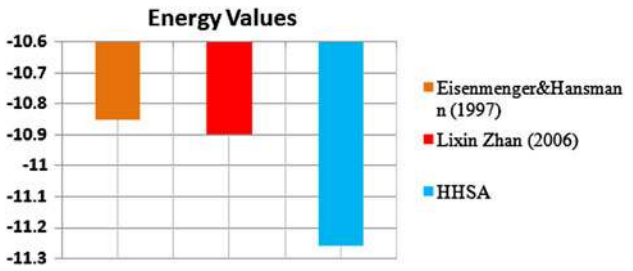



**Fig. 8** Barchart showing the results obtained for Met-enkephalin using ECEPP/3 with $\omega = 180°$

and ECEPP/2 with $\omega = 180°$, respectively. With comparison to other studies, the barcharts indicate that HHSA obtains the same optimal energy recorded by Zhan et al. (2006) and Meirovitch et al. (1994), while it outperforms others. Furthermore, based on ECEPP/2 with $\omega = 180°$, HHSA obtains a new energy value and

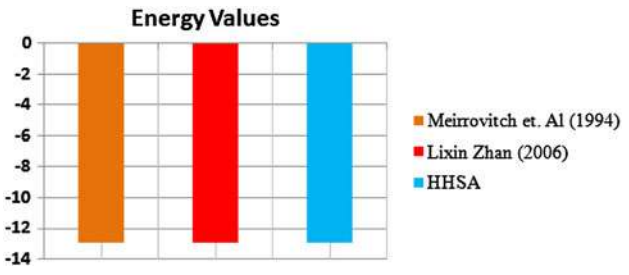



**Fig. 9** Barchart showing the results obtained for Met-enkephalin using ECEPP/2 force field

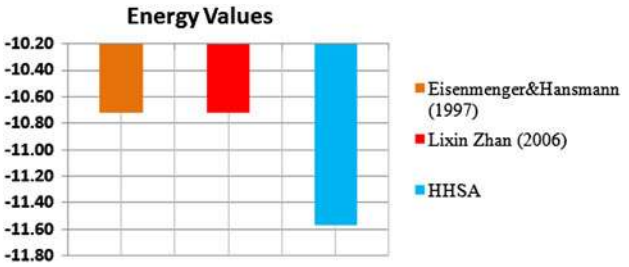



**Fig. 10** Barchart showing the results obtained for Met-enkephalin using ECEPP/2 with $\omega = 180°$

outperforms the previous lowest energies obtained by Zhan et al. (2006) and Eisenmenger and Hansmann (1997).

The coordinates of the lowest energies of the 'Met-enkephalin' protein for the results obtained in this study by HHSA and the results obtained in some of the previous studies are indicated in Table 8. The labels E/2 and E/3 mean that the configurations are obtained based on ECEPP/2 and ECEPP/3 force fields, respectively. If the label has a subscript $\pi$, it means that the angle $\omega$ is fixed at 180°. The lowest is the energy, the more stable is the protein. Table 8 shows the internal coordinates of lowest energy for Met-enkephalin in this study and some previous studies, those coordinates show the values of the torsion angles of the amino acids after the energy is stable on the lowest value.

Moreover, HHSA results are compared to the recent study of Nicosia and Stracquadanio (2009) who have studied some proteins, including 'Met-enkephalin', '1CRN', '1E0L', and '1IGD' based on ECEPP/3 force field with explicit solvent term. HHSA has obtained better energy values than the three proteins, 'Met-enkephalin', '1CRN', '1E0L', but could not obtain the same energy value for the longer protein 1IGD as can be seen in Table 9 and Fig. 11.

However, for the protein '1CRN', HHSA has obtained a value of RSDM = 6 Å which is considered a successful prediction based on CASP6, where the most successful ab initio methods have presented values of RSMD ranging from 4 to 6 Å for the proteins of length less than 100 residues (Dorn et al. 2008).

It is worthy to mention that all the methods in our comparison are ab initio methods as we cannot compare

**Table 8** Internal coordinates of lowest energy for Met-enkephalin in this study and some previous studies

| | Torsion | E/2 | $E/2_{\pi}$ | E/3 | $E/3_{\pi}$ | E/2(a) | E/2 $(a)_{\pi}$ | E/3 (a) | E/3 $(a)_{\pi}$ | E/3 (b) | E/3 $(b)_{\pi}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Tyr1 | x1 | −172.60 | −172.81 | −173.20 | −171.95 | −172.60 | −179.80 | −173.20 | 59.90 | −173.20 | −174.20 |
| | x2 | 78.70 | 76.19 | 79.30 | 93.43 | −101.30 | 68.60 | −100.70 | 94.10 | −100.50 | −85.20 |
| | x6 | −165.90 | −154.90 | −166.30 | −176.93 | 14.10 | −34.70 | 13.70 | −21.30 | 13.60 | 2.80 |
| | $\phi$ | −85.80 | −81.51 | −83.10 | −162.80 | −85.80 | −86.30 | −83.10 | 168.10 | −83.50 | −162.70 |
| | $\psi$ | 156.20 | 155.87 | 155.80 | −40.53 | 156.20 | 153.70 | 155.80 | 0.90 | 155.80 | −41.70 |
| | $\omega$ | −176.90 | 180.00 | −177.10 | 180.00 | −176.90 | 180.00 | −177.10 | 180.00 | 177.20 | 180.00 |
| Gly2 | $\phi$ | −154.50 | −156.36 | −154.20 | 64.70 | −154.50 | −161.50 | −154.20 | 126.80 | −154.30 | 65.80 |
| | $\psi$ | 83.60 | 81.08 | 85.80 | −89.86 | 83.70 | 71.10 | 85.50 | −21.20 | 86.00 | −87.00 |
| | $\omega$ | 168.60 | 180.00 | 168.50 | 180.00 | 168.60 | 180.00 | 168.50 | 180.00 | 168.50 | 180.00 |
| Gly3 | $\phi$ | 83.70 | 87.31 | 83.00 | −152.59 | 83.70 | 64.10 | 83.00 | 83.70 | 83.00 | −157.30 |
| | $\psi$ | −73.80 | −69.25 | −75.00 | 34.17 | −73.90 | −93.50 | −75.00 | −61.60 | −75.10 | 34.90 |
| | $\omega$ | −170.10 | 180.00 | −170.00 | 180.00 | −170.10 | 180.00 | −170.00 | 180.00 | −169.90 | 180.00 |
| Phe4 | x1 | 58.80 | 57.53 | 58.90 | 52.19 | 58.80 | 179.80 | 58.90 | 58.60 | 58.80 | 52.40 |
| | x2 | −85.40 | −86.41 | 94.50 | −97.20 | −85.40 | −100.00 | −85.50 | 92.90 | −85.50 | −96.00 |
| | $\phi$ | −137.00 | −141.91 | −136.80 | −155.66 | −137.00 | −81.70 | −136.80 | −128.20 | −136.90 | −158.80 |
| | $\psi$ | 19.30 | 17.48 | 19.10 | 159.89 | 19.30 | −29.20 | 19.10 | 18.80 | 19.10 | 159.50 |
| | $\omega$ | −174.10 | 180.00 | −174.10 | 180.00 | −174.10 | 180.00 | −174.10 | 180.00 | −174.10 | 180.00 |
| Met5 | x1 | 52.80 | 57.09 | 52.90 | −66.73 | 52.80 | −65.10 | 52.90 | 55.70 | 52.90 | −66.10 |
| | x2 | 175.30 | 174.07 | 175.30 | −180.00 | 175.30 | −179.20 | 175.30 | −178.60 | 175.30 | −179.60 |
| | x3 | 180.00 | −178.95 | 180.00 | 180.00 | −179.80 | −179.30 | −179.90 | 177.00 | −179.90 | −179.90 |
| | x4 | −178.60 | 63.01 | −58.60 | 59.94 | 61.40 | −179.90 | −178.60 | −179.30 | −178.60 | 60.10 |
| | $\phi$ | −163.60 | −164.10 | −163.40 | −79.27 | −163.60 | −80.70 | −163.40 | −162.10 | −163.40 | −82.40 |
| | $\psi$ | 160.40 | 171.13 | 160.80 | 130.95 | 160.40 | 143.50 | 160.80 | 7.50 | 160.80 | 134.10 |
| | $\omega$ | −179.70 | 180.00 | 180.00 | 180.00 | −179.70 | 180.00 | −179.80 | 180.00 | −179.80 | 180.00 |
| Energy (kcal/mol) | | −12.91 | −11.57 | −12.43 | −11.26 | −12.91 | −10.72 | −12.43 | −10.90 | −11.71 | −10.85 |

**Table 9** The lowest energies (in kcal/mol) obtained for benchmark sequences based on ECEPP/3 force fields with explicit solvent

| Protein | Nicosia | HHSA |
|---|---|---|
| Met-enkephalin | −24.84 | −31.42 |
| 1E0L | −233.02 | −235.47 |
| 1CRN | −225.22 | −244.03 |
| 1IGD | −584.26 | −556.41 |

with other methods for two reasons: first, in the comparison, the energy function should be same as different energy functions give different energy values, and the energy function used in our research is for ab initio methods so we cannot compare with other methods. Second, the ab initio methods predict the tertiary structure of protein from scratch, without any prior knowledge about sequence, while other methods use prior knowledge of protein so no way to compare.

### 5.3 Structures predicted by HHSA compared to the native structures

This section shows the graphical representation of the structures obtained by the HHSA for the four proteins,

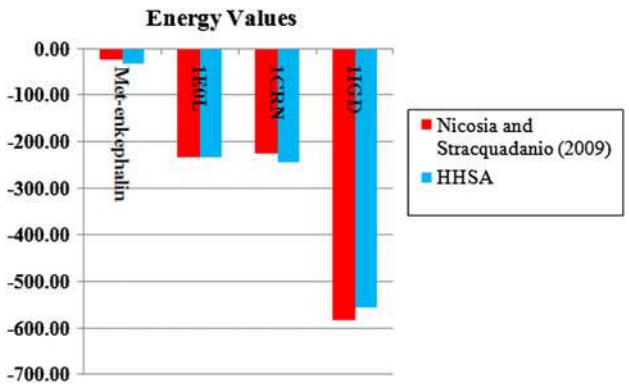
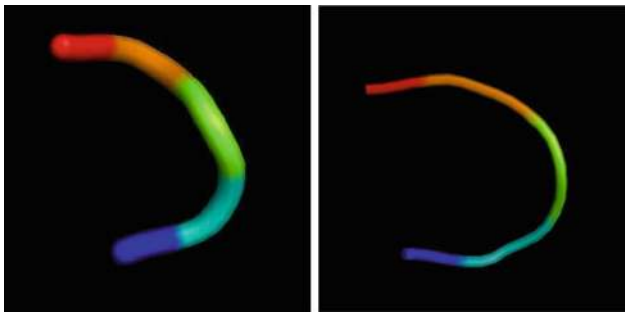


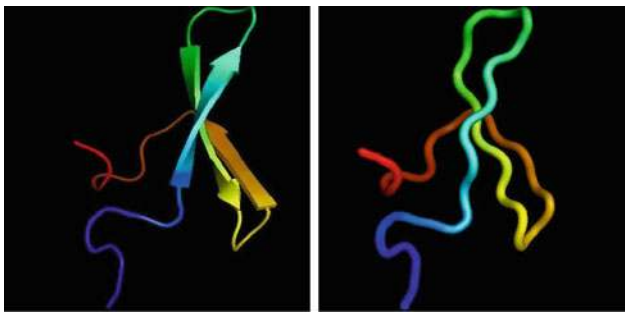

**Fig. 11** The lowest energies obtained based on ECEPP/3 force field with explicit solvent

'Met-enkephalin', '1CRN', '1E0L', and '1IGD', compared to the graphs of the native structure of the same proteins in PDB. Figures 12, 13, 14, and 15 show the graphical representation of the native and predicted structures of the proteins Met-enkephalin, 1E0L, 1CRN, and 1IGD, respectively. The figures show the similarity between the predicted protein and native one using the same rotation angle, they show the effectiveness of HHSA for proteins up



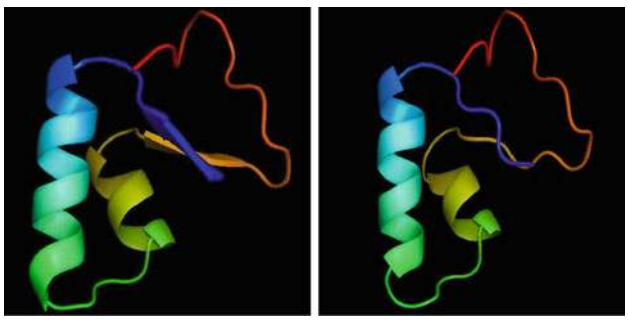
(a) (b)

**Fig. 12** Comparison between native and predicted structures of Met-enkephalin. **a** Native structure, **b** predicted structure



(a) (b)

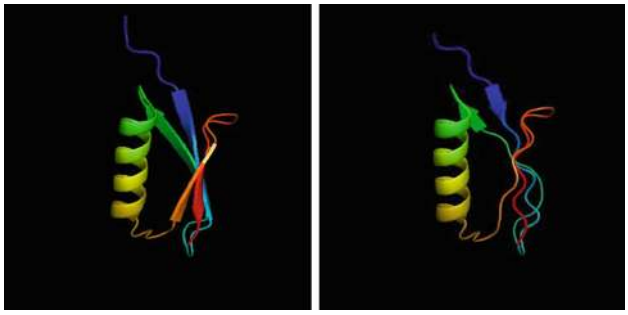
**Fig. 13** Comparison between native and predicted structures of 1E0L, **a** Native structure, **b** predicted structure



(a) (b)

**Fig. 14** Comparison between native and predicted structures of 1CRN. **a** Native structure, **b** predicted structure

to 50 residues. The figures were generated using PyMol DeLano (2002), as used by Zhan et al. (2006). Moreover, a root mean square deviation (RMSD) is calculated for the three proteins, 1E0L, 1CRN, and 1IGD, to validate the similarity between the native and predicted structures. The HHSA has obtained a value of RMSD = 4.5 Å for the protein 1E0L, RMSD = 6 Å for the protein 1CRN, and



(a) (b)

**Fig. 15** Comparison between native and predicted structures of 1IGD. **a** Native structure, **b** predicted structure

RMSD = 6.9 Å for the protein 1IGD. A value of RMSD = 6 Å is considered a successful prediction based on CASP6, where most successful ab initio methods have presented values of RMSD ranging from 4 to 6 Å for the proteins of length less than 100 residues (Dorn et al. 2008).

## 6 Conclusions and future work

This paper has presented a HHSA for ab initio protein tertiary structure prediction (PSPP). Initially, the harmony search is adapted to PSPP, called AHSA. AHSA is the basic HSA (Geem et al. 2001) with adaptive HMCR and PAR. The adaptation takes into consideration the exploration and exploitation concepts of the search space. The HHSA is the AHSA hybridized with ILS to improve the local exploitation and global best concept of PSO to improve the convergence rate.

A parameter sensitivity analysis for HHSA has been conducted using 12 convergence cases each of which having a particular parameter setting. The results show that, in general, using larger values of HMCR and PSR increases the performance of HHSA. A comparative study between AHSA and HHSA has been carried out; the comparative evaluation shows that the HHSA achieves better results than AHSA. Interestingly, two new best results were obtained by HHSA in comparison with the comparative methods using the same benchmark.

Hybridizing ILS as a local optimizer and global best as convergence accelerator with AHSA as a global optimizer produced a superior method for PSPP. More testing benchmarks can be investigated in future to further investigate the successful performance of HHSA.

**Acknowledgments** The work in this paper was partly supported by the Universiti Sains Malaysia (USM) Fellowship awarded to the first author. The second author is grateful to be awarded a Postdoctoral Fellowship from the School of Computer Sciences (USM).

## References


Abagyan R, Maiorov V (1988) A simple qualitative representation of polypeptide chain folds: comparison of protein tertiary structures. J Biomol Struct Dyn 5(6):1267–1279

Abual-Rub M, Abdullah R (2008) A survey of protein fold recognition algorithms. J Comput Sci 4(9):768–776

Al-Betar M, Khader AT (2012) A harmony search algorithm for university course timetabling. Ann Oper Res 194(1):3–31

Al-Betar MA, Khader AT, Liao IY (2010a) A harmony search algorithm with multi-pitch adjusting rate for university course timetabling. In: Geem ZW (ed) Recent advances in harmony search algorithm, SCI, vol 270. Springer, Berlin, pp 147–162

Al-Betar MA, Khader AT, Nadi F (2010b) Selection mechanisms in memory consideration for examination timetabling with harmony search. In: GECCO '10: proceedings of genetic and evolutionary computation conference. ACM, Portland

Al-Betar MA, Khader AT, Thomas JJ (2010c) A combination of metaheuristic components based on harmony search for the uncapacitated examination timetabling. In: 8th International conference on the practice and theory of automated timetabling (PATAT 2010), Belfast, Northern Ireland

Al-Betar MA, Doush IA, Khader AT, Awadallah MA (2012a) Novel selection schemes for harmony search. Appl Math Comput 218(10):6095 – 6117

Al-Betar MA, Khader AT, Zaman M (2012b) University course timetabling using a hybrid harmony search metaheuristic algorithm. IEEE Trans Syst Man Cybern Part C Appl Rev. doi:10.1109/TSMCC.2011.2174356:1-18

Alatas B (2010) Chaotic harmony search algorithms. Appl Math Comput 216(9):2687–2699

Alia O, Mandava R (2011) The variants of the harmony search algorithm: an overview. Artif Intell Rev 36:49–68

Almansoori W, Gao S, Jarada T, Elsheikh A, Murshed A, Jida J, Alhajj R, Rokne J (2012) Link prediction and classification in social networks and its application in healthcare and systems biology. In: Network modeling and analysis in health informatics and bioinformatics, pp 1–10. http://dx.doi.org/10.1007/s13721-012-0005-7

Androulakis I, Maranas C, Floudas C (1997) Prediction of oligopeptide conformations via deterministic global optimization. J Glob Optim 11(1):1–34

Anfinsen CB (1973) Principles that govern the folding of protein chains. Science 181(4096):223–230

Baker D (2000) A surprising simplicity to protein folding. Nature 405(6782):39–42

Blum C, Roli A (2003) Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput Surv 35(3):268–308

Brooks BR, Bruccoleri RE, Olafson BD, States DJ, Swaminathan S, Karplus M (1983) CHARMM: a program for macromolecular energy, minimization, and dynamics calculations. J Comput Chem 4(2):187–217

Chivian D, Robertson T, Bonneau R, Baker D (2003) Ab initio methods. Methods Biochem Anal 44:547–558

Chothia C, Lesk A (1986) The relation between the divergence of sequence and structure in proteins. EMBO J 5(4):823–826

Cutello V, Narzisi G, Nicosia G (2006) A multi-objective evolutionary approach to the protein structure prediction problem. J Roy Soc Interface 3(6):139–151

Das S, Mukhopadhyay A, Roy A, Abraham A, Panigrahi BK (2011) Exploratory power of the harmony search algorithm: Analysis and improvements for global numerical optimization. IEEE Trans Syst Man Cybern Part B Cybern 41(1):89–106

DeLano WL (2002) The PyMOL molecular graphics system. http://www.pymol.org

Dorn M, Breda A, Norberto de Souza O (2008) A hybrid method for the protein structure prediction problem. In: Bazzan A, Craven M, Martins N (eds) Advances in bioinformatics and computational biology. Lecture notes in computer science, vol 5167. Springer, Berlin, Heidelberg, pp 47–56

Dudek M, Objects B (2007) Igor, a simple integrable model of a polypeptide chain III. Model parameterization, TechReport. http://biomoleculeobjects.com/paper6/paper3.pdf

Eisenmenger F, Hansmann U (1997) Variation of the energy landscape of a small peptide under a change from the ecepp/2 force field to ECEPP/3. J Phys Chem B 101(16):3304–3310

Eisenmenger F, Hansmann U, Hayryan S, Hu C (2001) [SMMP] a modern package for simulation of proteins. Comput Phys Commun 138(2):192–212

Eisenmenger F, Hansmann U, Hayryan S, Hu C (2006) An enhanced version of SMMP—open-source software package for simulation of proteins. Comput Phys Commun 174(5):422–429

Geem ZW, Sim KB (2010) Parameter-setting-free harmony search algorithm. Appl Math Comput 217(8):3881–3889

Geem ZW, Kim JH, Loganathan GV (2001) A new heuristic optimization algorithm: harmony search. Simulation 76(2):60–68

Helles G (2008) A comparative study of the reported performance of ab initio protein structure prediction algorithms. J Roy Soc Interface 5(21):387–396. doi:10.1098/rsif.2007.1278

Hinds D, Levitt M (1992) A lattice model for protein structure prediction at low resolution. Proc Natl Acad Sci USA 89(7):2536–2540

Ingram G, Zhang T (2009) Overview of applications and developments in the harmony search algorithm. In: Geem ZW (ed) Music-inspired harmony search algorithm. Springer, Berlin, pp 15–37

Jorgensen WL, Tirado-Rives J (1988) The OPLS [optimized potentials for liquid simulations] potential functions for proteins, energy minimizations for crystals of cyclic peptides and crambin. J Am Chem Soc 110(6):1657–1666

Kirkpatrick S, Gelatt J C D, Vecchi MP (1983) Optimization by simulated annealing. Science 220(4598):671–680

Lee J, Wu S, Zhang Y (2009) ab initio Protein Structure Prediction, Springer Netherlands, chap From Protein Structure to Function with Bioinformatics, pp 3–25

Lee KS, Geem ZW (2005) A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. Comput Methods App Mech Eng 194(36–38):3902–3933. doi:10.1016/j.cma.2004.09.007

Levitt M (1976) A simplified representation of protein conformations for rapid simulation of protein folding. J Mol Biol 104(1):59–107

Li Z, Scheraga H (1988) Structure and free energy of complex thermodynamic systems. J Mol Struct THEOCHEM 179(1):333–352

Li Z, Scheraga HA (1987) Monte carlo-minimization approach to the multiple-minima problem in protein folding. Proc Natl Acad Sci USA 84(19):6611–6615

Mahdavi M, Abolhassani H (2009) Harmony k-means algorithm for document clustering. Data Min Knowl Discov 18:370–391

Mahdavi M, Fesanghary M, Damangir E (2007) An improved harmony search algorithm for solving optimization problems. Appl Math Comput 188(2):1567–1579

Meirovitch H, Meirovitch E, Michel A, Vasquez M (1994) A simple and effective procedure for conformational search of macromolecules: application to met-and leu-enkephalin. J Phys Chem 98(25):6241–6243

Mohsen A, Khader A, Ramachandram D (2010) An optimization algorithm based on harmony search for rna secondary structure prediction. In: Geem Z (ed) Recent advances in harmony search algorithm, studies in computational intelligence, vol 270. Springer, Berlin, pp 163–174

Nadi F, Khader AT, Al-Betar MA (2010) Adaptive genetic algorithm using harmony search. In: Proceedings of the 12th annual conference on Genetic and evolutionary computation, GECCO '10. ACM, New York, pp 819–820

Nicosia G, Stracquadanio G (2009) A design-for-yield algorithm to assess and improve the structural and energetic robustness of proteins and drugs. In: Experimental algorithms, pp 245–256

Omran MGH, Mahdavi M (2008) Global-best harmony search. Appl Math Comput 198(2):643–656

Pan QK, Suganthan P, Tasgetiren MF, Liang J (2010) A self-adaptive global best harmony search algorithm for continuous optimization problems. Appl Math Comput 216(3):830–848

Saka M, Aydogdu I, Hasancebi O, Geem Z (2011) Harmony search algorithms in structural engineering. In: Yang XS, Koziel S (eds) Computational optimization and applications in engineering and industry, studies in computational intelligence, vol 359. Springer, Berlin, pp 145–182

Tang W, Yarowsky P, Tasch U (2012) Detecting als and parkinsons disease in rats through locomotion analysis. In: Network modeling and analysis in health informatics and bioinformatics, pp 1–6. http://dx.doi.org/10.1007/s13721-012-0004-8

Wang TY, Wu KB, Liu YW (2001) A simulated annealing algorithm for facility layout problems under variable demand in cellular manufacturing systems. Comput Ind 46(2):181–188

Weiner SJ, Kollman PA, Case DA, Singh UC, Ghio C, Alagona G, Profeta S, Weiner P (1984) A new force field for molecular mechanical simulation of nucleic acids and proteins. J Am Chem Soc 106(3):765–784

Yang XS (2009) Harmony search as a metaheuristic algorithm. In: Geem ZW (ed) Music-inspired harmony search algorithm. Springer, Berlin, pp 1–14

Zhan L, Chen J, Liu W (2006) Conformational study of met-enkephalin based on the ECEPP force fields. Biophys J 91(7):2399–2404