Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)

A Particle Swarm Optimizer for Finding Minimum Free Energy RNA Secondary Structures

Michael Geis[1] and Martin Middendorf, *Member, IEEE*[2]

[1]Bioinformatics Group, Department of Computer Science and Interdisciplinary Center for Bioinformatics, University of Leipzig, Haertelstr. 16-18, D-04107 Leipzig, Germany.
[2]Parallel Computing and Complex Systems Group, Department of Computer Science, University of Leipzig, Augustusplatz 10/11, D-04109 Leipzig, Germany.

***Abstract*— This paper introduces the HelixPSO Particle Swarm Optimization (PSO) algorithm for finding minimum energy RNA secondary structures. It is shown experimentally that HelixPSO profits when it is combined with a genetic algorithm that finds a good starting population for HelixPSO. On all test instances this hybrid variant of HelixPSO performs significantly better than a state-of-the-art genetic algorithm. Also compared with another PSO algorithm that has been proposed very recently for the prediction of RNA secondary structures, HelixPSO is more efficient both in terms of free energy and correctly predicted base pairs.**

***Index Terms*— Particle Swarm Optimization; Genetic algorithms; RNA secondary structure;**

I. Introduction

Beyond their function in protein coding, RNA plays an important rule for many cell functions, such as controlling gene expression, catalysing chemical reactions or complementing protein enzyme based activity [1]. The secondary structure of RNA can be used infer and explain its function (see, e.g., [2], [3]). Determining the secondary structure of an RNA often constitutes an essential step towards predicting the tertiary structure.

Different algorithmic approaches exist for RNA secondary prediction. Comparative sequence analysis [4] as well as thermodynamic optimization [5] require sequence information only. Thermodynamic folding algorithms rely on an energy model that assumes additive contributions from stacked base pairs and various types of loops ([6], [7]). The corresponding energy values can be obtained, e.g., with measuring absorbance melting curves or with microcalorimetry ([8]).

The prediction of RNA secondary structures is particulary difficult when pseudoknots are involved. One reason for this is that little is known about energy models involving pseudoknots [9]. Another reason is that thermodynamic structure prediction involving pseudoknots is an NP-complete problem for the standard energy model [10].

While the benchmark for RNA folding algorithms mfold [11], uses Dynamic Programming (DP), a number of attempts have been made to apply metaheuristics to the domain. Most of the proposed algorithms are genetic algorithms (GAs) and it has been argued that GAs can simulate the actual folding process of an RNA sequence and therefore achieve higher prediction rates of base pairs than DP [12]. Among those authors proposing GAs for predicting RNA secondary structure are Shapiro and Navetta [13], van Batenburg et al. [14], Gultyaev et al. [15], and Benedetti and Morosetti [16]. Initial approaches were rather crude, but continuous refinement achieved greater prediction accuracy than DP [17]. Massively parallel implementations [18] as well as efficient algorithms for small architectures exist [19]. Further work includes Chen et al. [20], Fogel [21], and Ogata et al. [22]. RnaPredict and its parallelized version pRnaPredict are the most recent GAs predicting RNA secondary structure (see [23], [24]). Very recently, Neethling and Engelbrecht [25] proposed the first Particle Swarm Optimization algorithm called SetPSO to predict RNA secondary structures.

Particle Swarm Optimization (PSO) is an optimization technique that has been developed by Kennedy and Eberhardt [26], [27]. PSO is inspired by the behaviour of real swarms (e.g. fish schools or bird flocks) and was originally proposed for the optimization of real-valued continuous functions. Meanwhile several variants of PSO for discrete and binary problems have been developed (e.g., [28]). Further variants of PSO include multi-objective algorithms (e.g., [29]) and hybrids that incorporate elements of other approaches of Evolutionary Computation (e.g., [30]).

In this paper we propose a new PSO approach called HelixPSO for finding RNA secondary structures with minimum free energy. The free energy is calculated with the RNAeval algorithm from the ViennaRNA package (see [31], [32]). HelixPSO is compared to the state-of-the-art genetic algorithm RnaPredict that was developed by Wiese and Glen [23]. In order to have a direct comparison and to be able to use RNAeval for energy calculation we reimplemented RnaPredict. HelixPSO is also compared to the other existing PSO algorithm for secondary structure prediction SetPSO.

A short overview over RNA secondary structure is given in Section II. A description of RnaPredict in given in Section III. The PSO approach and algorithm SetPSO are presented in Section IV. HelixPSO is introduced in Section V. The experiments are described Section VI. Results are presented in Section VII. Conclusions and future work are given in the final Section VIII.

1-4244-0708-7/07/$20.00 ©2007 IEEE

II. RNA Secondary Structure

An RNA molecule is a sequence of the nucleotides adenine (A), guanine (G), cytosine (C), and uracil (U). This sequence or chain of nucleotides is called the primary structure. The secondary structure is the result of hydrogen bonds between nucleotides that are not neighbours in the chain. Typically these hydrogen bonds occur only between G and C, or A and U, or G and U (or vice versa). The so connected nucleotides are called base pairs and are denoted by GC, CG, AU, UA, or GU, or UG (the first base is the one with the smaller index in the chain). A main element of secondary RNA structures are helices, which are sets of two or more adjacent base pairs that form a ladder like structure. Thus, a helix of size $k \geq 2$ consist of k base pairs with indices (i, j),$(i+1, j-1) \ldots, (i+k, j-k)$ where $1 \leq i < i + k < j - k < j \leq n$ and n is the length of the RNA sequence. An RNA secondary structure is defined by a set of helices, such that each nucleotide occurs in at most one helix and the following properties hold: i) for each base pair (i, j) of a helix $j - i \geq 3$ holds and ii) for any two base pairs (i, j), (i', j') that occur in the helices either $i < i' < j' < j$ or $i' < i < j < j'$ holds. Property (i) implies that the connecting loop between the strands of a helix has length at least three. Property (ii) means that helices are nested. It should be mentioned that exceptions from this rules can be found in real RNA molecules. But as has been done by many authors we consider only secondary structures that have this properties.

Several algorithms for secondary structure prediction start by computing the set H of all possible helices of an RNA molecule and then trying to find a subset of H that defines an optimal (in some sense) secondary structure. This is also done by RnaPredict and both PSO algorithms that are considered in this paper.

In general, the free energy of an RNA secondary structure increases with a larger number of base pairs that are included in helices. But the energy depends also on the type of base pairs. There exists several functions to compute the free energy of an RNA secondary structure. In this paper we use the RNAeval algorithm from the ViennaRNA package (see [31], [32]).

III. RnaPredict

The genetic algorithm RnaPredict ([23]) finds low energy RNA conformations by applying selection, mutation and crossover operators to a population of chromosomes. Each chromosome is a permutation of the elements in the set H of helices. A subset of H that defines a secondary structure is derived from a chromosome as follows. Starting with an empty set the permutation is scanned from the beginning and a helix is added to the set if it is feasible. To compute the free energy of a secondary structure the Nussinov-Jacobson energy model, the individual nearest neighbor (INN) model [33], and the individual nearest neighbor-hydrogen bond (INN-HB) models [34] are used in RnaPredict. To be comparable to HelixPSO, our RnaPredict implementation uses the RNAeval algorithm of the ViennaRNA package, see [31], [32]. Its energy and stacking parameters can be found in [6] and [7]. The algorithm does not predict pseudo knots.

Algorithm 1 RnaPredict

```
Initialize population with random permutations
for i = 1 to number of iterations do
  for j = 1 to (population size/2) do
    Select chromosomes c, c' from population
    if random() < p_c then
      offspring o, o'=CX(c, c')
      if random() < p_m then
        Mutate(o)
        Mutate(o')
      Add best child and best parent to new_population
    else
      Add c and c' to new_population
  population=new_population
  global_best=FindGlobalBest(population)
return global_best
```

Mutation is done in RnaPredict by swapping two random indices in the permutation. Wiese et al. ([23], [35]) have investigated the optimization behaviour of RnaPredict with a variety of crossover operators (such as edge recombination crossover (ERC) [36], OX2 [37], and cycle crossover (CX) [38]) and different selection operators (Keep-Best-Reproduction (KBR), Standard Roulette Wheel Selection (STDS)) [39]. For our implementation of RnaPredict we used the operators that performed best, namely, the CX and KBR operators. CX crossover combines a cycle of one permutation with the remainder of another. It ensures that the result is a permutation and each value agrees in position with one of the parent permutations. KBR selection examines two parents and their two offsprings and always keeps the best child and the best parent. RnaPredict uses 1-elitism and selects particles for reproduction via fitness based roulette wheel selection. A pseudo code of RnaPredict is given in Algorithm 1, where p_c is the crossover probability and p_m is the mutation probability.

IV. PSO and SetPSO

PSO is an iterative optimization heuristic for function optimization where a swarm of m particles searches in a multidimensional space (see also [26]). Typically in PSO, each particle i has a position and a velocity. The velocity is updated in each iteration according to Formula 1 where: i) the inertia weight $w > 0$ controls the influence of the previous velocity, ii) the current position of the particle is denoted by x_i, iii) parameter $c_1 > 0$ controls the impact of the personal best position found so far y_i (called $pbest$), iv) parameter c_2 determines the impact of the best position that has been found so far by any of the particles in neighborhood $\hat{y}_i$ of particle i (called $lbest$), v) random values r_1 and r_2 are drawn with uniform probability from $[0, 1]$. After velocity update all particles move with their new velocity to their new positions (Formula 2). Then for each particle i the objective function f is evaluated at its new position. If $f(x_i(t + 1)) < f(y_i)$ (assuming the function has to be minimized) the personal best position y_i is updated accordingly, i.e. y_i is set to $x_i(t + 1)$.

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot r_1 \cdot (y_i - x_i) + c_2 \cdot r_2 \cdot (\hat{y}_i - x_i) \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

SetPSO [25] is the first PSO algorithm for finding RNA secondary structures with minimum free energy. As other PSO algorithms for discrete problem SetPSO differs from the standard PSO scheme. Similar to RnaPredict, SetPSO searches on the set of helices of a given RNA and represents a secondary structure as a set of helices (which must be feasible according to the rules given in Section II).

Movement of a particle is defined in terms of addition and subtraction of helices from the corresponding set. A set O of helices which is removed from the particles position is computed from the empty set by adding a helix that is neither in the $pbest$ solution nor in the $lbest$ solution with probability p_I. A candidate set C of helices which might be added to the particles position is computed from the empty by adding each helix of a target solution with probability p_C and each helix from the set of all helices with probability p_R. The target solution is a combination of the particles $pbest$ and $lbest$ solution. To avoid base pair conflicts, all helices in the set O are removed before those of the set C are added (if feasible). The computation of the sets O and C is called the velocity update and the actual computation of the new solution is the position update. For more details of SetPSO see [25].

V. HELIXPSO

Algorithm HelixPSO that is proposed in this section is a PSO algorithm for finding RNA secondary structures with minimum free energy. Similar as RnaPredict, algorithm HelixPSO encodes a secondary structure as a permutation of the set of all helices of an RNA sequence.

Similar as in SetPSO a particle moves with respect to a target position. For this each particle i has an associated set of candidate target positions T_i and for each $t \in T_i$ a weight $w(t) > 0$. The relative weight of a position in T_i determines the probability that it is chosen as a target. T_i is initialized with a random position, i.e., a permutation that is generated randomly, that has weight 1.0. After each iteration of HelixPSO each weight is decreased by multiplication with a parameter ρ, $0 < \rho < 1$. A position that has a weight less than a threshold τ is removed from T_i. The reason is that elements with a very small weight have only small chances to be chosen as a target but require much memory space. Then the personal position and either the global best or the cluster best position (details are given later) are added to T_i with probability $c_1 \cdot r_1$ and $c_2 \cdot r_2$, respectively where r_1 and r_2 are random numbers that are chosen uniformly from $[0, 1]$. Note, that this is similar to the impact of the personal best and global best values for the standard PSO scheme (see Formula 1). The initial weight of each position that is added to T_i is 1.0.

HelixPso is a multi-swarm algorithm where the swarm of particles is partitioned into several subswarms. Subswarms are used to encourage the swarm to search in different areas of the search space. These subswarms are called clusters. All particles in a cluster (with one exception) use their personal best position and the cluster best position to update their set of candidate target positions. Only the cluster best particle (i.e., the particle which has the best personal position within the cluster) does the update with respect to its personal best and the global best position. The idea behind this is to ensure that particles in a cluster stay close to each other, while the clusters in their entirety should converge towards the global best solution.

When a particle i has chosen its target position from the set T_i the particle moves towards the target al follows. The positions of some helices in the particles own permutation vector are swapped to make it more similar to the permutation vector of the target. As not every such transposition causes a change in the corresponding secondary structure each particle performs a series of α of transpositions, where α is a parameter of the algorithm. A transposition is done with probability $\beta > 0$ in direction of the target position as described in the following. To find the first helix of the transposition the permutation vector of the particle is scanned from the beginning. Helices which are at the same place in the particles and the targets permutation are skipped. Otherwise an index is

Algorithm 2 HelixPSO

```
Initialize the swarm by creating the particles, partition them into clusters and randomly initialize the permutation vector of each particle.
Initialize the personal best position for each particle to the current position.
Calculate the cluster best position for each cluster.
Calculate the global best position.
j = 1
while (j < maximum number of solutions) do
  for all particles p do
    Let c denote the secondary structure of particle p.
    for i = 1 to α do
      Chose a target from T_p.
      Chose an index idx in the permutation of particle p.
      if rand() < 0.1 or if no index was chosen then
        Perform a random transposition of two helices in the permutation vector and let c' be the corresponding secondary structure.
      else
        In the permutation of the particle swap the helix at idx with the helix that is at idx in the target permutation and let c' be the corresponding secondary structure.
      j++
      Compute the free energy of c'.
      if free energy of c' < free energy of c then
        c = c'
  Update pbest, cbest, and gbest.
  Apply 1-elitism, i.e., reposition the worst particle to the global best.
  For each particle p update T_p.
return gbest
```

chosen with probability $p_S > 0$. If an index j has been chosen the helix h of the target permutation at place j is determined. Then the helix h' at place j in the particles permutation is transposed with helix h. Thus, after the transposition the particle and the target have the same helix h' at place j. With probability $1 - \beta > 0$ a random transposition is done, i.e., for two randomly chosen helices their positions in the permutation of the particle are exchanged.

The series of transpositions is done greedily in HelixPSO, that is it is accepted only when the new secondary structure that is generated by the transpositions improves over the secondary structure before the transpositions were done. HelixPSO also uses 1-elitism, i.e., after each iteration the particle at the worst position is reset to the global best position. The pseudo code of HelixPSO is given in Algorithm 2.

We also tested a hybrid version of HelixPSO were the starting positions for the swarm are not chosen randomly but are handed over to HelixPSO from another algorithm. In this paper we used RnaPredict to compute the starting positions. This hybrid version of HelixPSO is denoted H-HelixPSO or γ-H-HelixPSO were γ denotes the relative number of solutions that are computed by RnaPredict.

VI. Experiments

For the experiments we used RNA sequences of different lengths that have already been used by other authors so that we can compare HelixPSO with RnaPredict and SetPSO. The four test sequences are (they are available from the comparative RNA website [40], the accession number is given in brackets): i) Aureoumbra lagunensis (U40258), Group I intron, 16S rRNA with length 468, ii) Drosophila virilis (X05914) 16S rRNA with length 784, iii) Xenopus laevis (M27605) 16S rRNA with length 945, iv) Sulfolobus acidocaldarius (D14876) 16S rRNA with length 1492. For these sequences the natural secondary structures they fold into are also available from the comparative RNA website.

For RnaPredict we used the same parameter values that were also used in ([23]): population size 700, number of iterations 400, $p_m = 0.8$, $p_c = 0.7$. Note, that the total number of solutions that are generated during one run of RnaPredict is 280000.

In [25] the following parameter values were used for SetPSO: $p_C = 0.6$, $p_R = 0.5$, p_I was decreased linearly over the iterations from 0.9 to 0.1, swarm size 50, and the runs were done over 700 iterations. Hence, during one run of SetPSO the total number of solutions that are generated is 35000.

For the comparison with RnaPredict and for most other test runs the swarm size 500 was used over 560 iterations with 7 clusters. For this test runs the hybrid version of HelixPSO was used. The relative number of solutions that were used by RnaPredict to compute a starting population varied with the size of the RNA sequence. For the shorter sequences (lengths 468 and 784) the value $\gamma = 0.02$ was used. For the longer RNA sequence of length 945 (1492) the value of γ was 0.05 (respectively 0.2). For a comparison with SetPSO the same swarm size and number of iterations as in [25] were used, namely swarm size 50 over 700 iterations with 1 cluster.

For the other parameters the following standard values were used in the test runs (unless stated otherwise): $\alpha = 25$, $\beta = 0.9$, $\rho = 0.95$, $c_1 = c_2 = 3.0$, $p_S = 0.01$.

If not stated otherwise the tests in this paper with RnaPredict and HelixPSO have be done with a search space that is the set of all maximal helices, i.e., a helix that can not be extended by adding a base pair, plus some subhelices of maximal helices. The subhelices are created from the maximal helices by iteratively removing terminal base pairs as long as the resulting structure is still a helix, i.e., at least 2 base pairs remain. A base pair is called terminal when it includes the lowest numbered base from all bases that are included in any base pair of the corresponding helix. For each test case HelixPSO and RnaPredict were run 300 times (except for RNA sequence of length 1492 were only 150 runs of each algorithm were done). For the analysis of the optimization behaviour the global best solution has been recorded after every 10000 solutions that were generated.

We also conducted tests were the search space consist only of the maximal helices. For each test case the number of test runs was 150 (except for RNA sequence of length 1492 were only 50 runs of each algorithm were done).

VII. Results

A comparison of the optimization behaviour of H-HelixPSO and RnaPredict is shown in Figure 1. The figure shows for all four test RNAs the average free energy ΔG of the best so far found solution during a run for both algorithms. The figures show that RnaPredict finds in most cases better solutions at the beginning. But from about 12000 solution generations H-HelixPSO finds always better solutions. It can also be seen that H-HelixPSO has still not converged after 280000 solution generations whereas RnaPredict seems to have already nearly converged. The free energies of the best secondary structure that have been found by both algorithms at the end of a run are shown in Table I for the search space including subhelices of maximal helices and in Table II for the search space of maximal helices. It can be seen that H-HelixPSO found secondary structures which have a free anergy that is between 1.1 and 3.1 percent (1.6 and 3.4 percent) better than the secondary structures found by RnaPredict for the search space including subhelices of maximal helices (respectively for the search space including only maximal helices).

TABLE I

Free energy (ΔG in Kcal/mol) of secondary structures found after 280000 secondary structures have been generated by RnaPredict and H-HelixPSO; relative improvement of H-HelixPSO

RNA length	RnaPredict	H-HelixPSO	% Improvement
468	-124.4	-126.5	1.7
784	-124.3	-125.7	1.1
945	-207.8	-212.7	2.4
1492	-633.8	-653.5	3.1

A comparison of the runtimes of H-HelixPSO and RnaPredict is shown in Table III. The runtimes were measured on a PC with Intel Xeon 3.00GHz dual core unit with 4GB

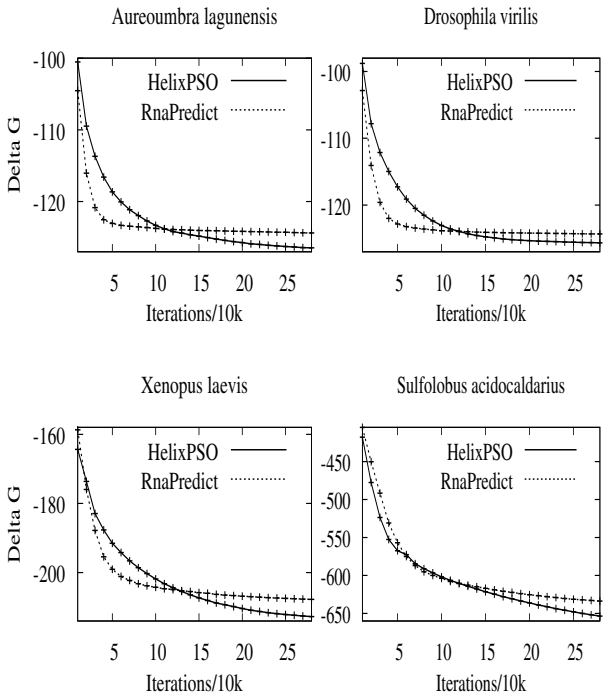

Fig. 1. Optimization behaviour of RnaPredict and H-HelixPSO over 280000 solution generations on all 4 test RNAs

TABLE II

FREE ENERGY (ΔG IN KCAL/MOL) OF SECONDARY STRUCTURES FOUND AFTER 280000 SECONDARY STRUCTURES HAVE BEEN GENERATED BY RNAPREDICT AND H-HELIXPSO RUN ON A SEARCH SPACE OF MAXIMAL HELICES; RELATIVE IMPROVEMENT OF H-HELIXPSO

RNA length	RnaPredict	H-HelixPSO	% Improvement
468	-124.6	-126.9	1.9
784	-123.1	-125.0	1.6
945	-209.3	-213.7	2.1
1492	-629.5	-650.9	3.4

RAM. It has to be mentioned that our implementations for RnaPredict and H-HelixPSO were not made with a particular emphasis to minimize the runtime. It can be seen that H-HelixPSO is between 1.5 and 1.9 times slower on the three longer RNA sequences and 2.5 times slower on the short RNA sequence. It should be noted that even when H-HelixPSO is allowed to generate only half of the number of RNA secondary structures as RnaPredict then the obtained solution quality of H-HelixPSO is still better as can be seen in Figure 1.

Table IV shows the free energy of SetPSO (this results are from [25]) and HelixPSO for two RNA sequences after 35000 solution generations. The two RNA sequences are those from our four test sequences for which results are also presented in [25]. It can be seen that HelixPSO finds secondary structures that have a free energy value ΔG that is more more then 11% less than the free energy of the secondary structures that are computed by SetPSO.

Neethling et al. [25] presented the number of base pairs in

TABLE III

RUNTIME OF RNAPREDICT AND H-HELIXPSO IN MINUTES

RNA length	RNSPredict	H-HelixPSO	Ratio
468	2.6	6.6	2.6
784	4.2	6.4	1.5
945	7	13	1.9
1492	133	234	1.8

TABLE IV

FREE ENERGY (ΔG IN KCAL/MOL) OF SOLUTIONS FOUND AFTER 35000 SOLUTIONS HAVE BEEN GENERATED BY SETPSO AND HELIXPSO; RELATIVE IMPROVEMENT OF HELIXPSO

RNA length	SetPSO	HelixPSO	% Improvement
945	-105.8	-120.4	11.8
784	-173.3	-201.8	11.6

TABLE V

NUMBER OF BASE PAIRS (BP) OF THE REFERENCE SECONDARY STRUCTURE FROM THE RNA WEBSIDE [40], SETPSO AND HELIXPSP; NUMBER OF BASE PAIRS (#CORRECT BP) THAT ARE THE SAME IN THE REFERENCE SECONDARY STRUCTURE AND THE SECONDARY STRUCTURE COMPUTED BY SETPSO AND HELIXPSO

RNA length	Reference	SetPSO		HelixPSO	
	bp	bp	#correct bp	bp	#correct bp
784	233	225.5	29.7	210.3	42.9
945	251	241.8	57.8	196.9	67.0

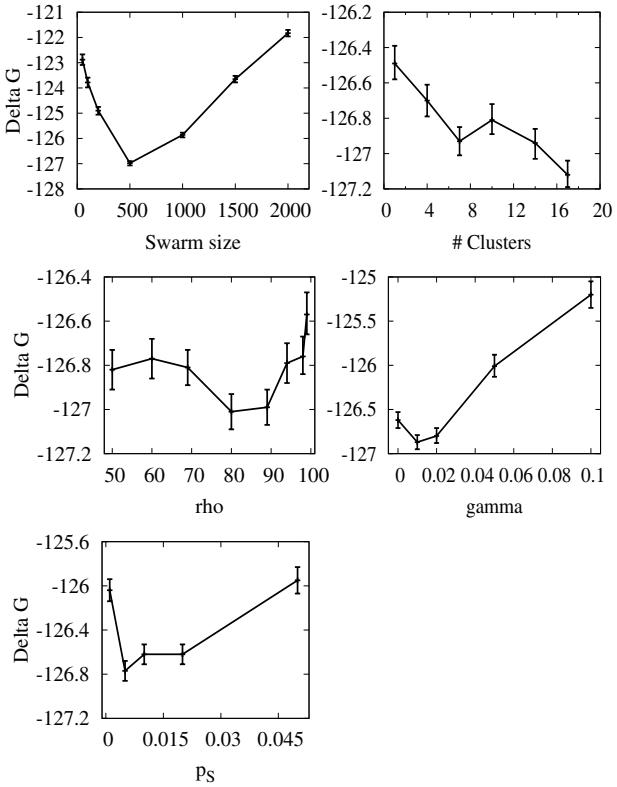

Fig. 2. Free energy of best solution found by H-HelixPSO for the RNA sequence of length 468 after generation of 280000 solutions when different parameter values are varied and all other parameter values are as in the standard value set; bars indicate the standard error

the secondary structure that were computed by SetPSO and compared this number to the number of base pairs in the true reference secondary structure that can be found on the RNA website [40]. They also computed the number of "correct" base pairs, i.e., the number of base pairs that are the same in the SetPSO solution and the reference structure. Table V presents this results together with the corresponding results for Helix-PSO. It can be seen that HelixPSO finds significantly more correct base pairs with respect to the reference secondary structure (44.4% more for the RNA sequence of length 784 and 15.9% more the RNA sequence of length 945). These results should be interpreted with care because HelixPSO and SetPSO use the minimum free energy as their optimization criterion but not the number of correct base pairs with the reference structure.

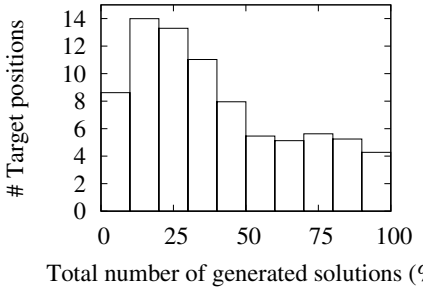

Fig. 3. Average length of the target set of a particle during a run of HelixPSO for the RNA sequence of length 486; the run is divided into 10 slices each corresponding to 28000 generated solutions

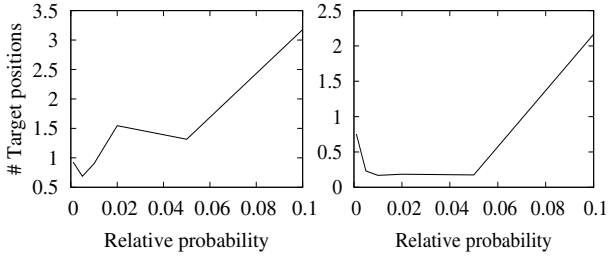

Fig. 4. Number of target positions with different weights(measured with respect to the probability to be chosen) in the target set of a particle during a run of HelixPSO for the RNA sequence of length 486; left (right) average over the first (respectively last) 28000 solution generations

In order to study the influence of different parameters on the optimization behaviour of H-Helix-PSO we made several test where the value of one parameter was varied. All other parameters values were the same as in the standard parameter set for H-HelixPSO. Figure 2 shows the results. As can be seen the swarm size has a great influence on the optimization behaviour. This is no surprise because for a fixed number of generated solutions the swarm size determines directly the number of iterations. The principle to use several subswarms in H-HelixPSO has a clear positive effect on its optimization behaviour. The free energy of the best found secondary structure decreased from -126.5 for one cluster to less then -127 for 17 clusters. The influence of parameter ρ which

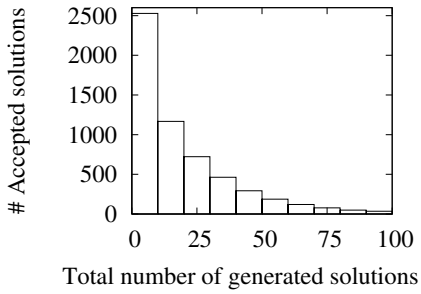

Fig. 5. Number of accepted solutions after a move during a run of HelixPSO for the RNA sequence of length 486; the run is divided into 10 slices each corresponding to 28000 generated solutions

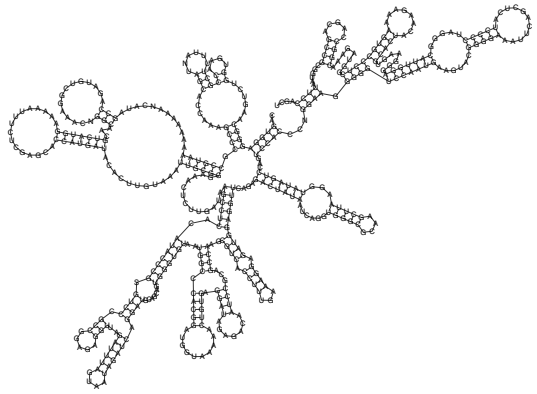

Fig. 6. Global best RNA secondary structure computed by H-HelixPSO for the Aureoumbra lagunensis RNA of length 486 over 300 runs

determines the decrease rate of the weights in the target set on the optimization behaviour is not very strong as long as $\rho < 0.98$. For higher values of ρ the solution quality decreases. One possible reason is that older target psoitions have a very long influence. The advantage to use RnaPredict to compute the starting positions for the particles of HelixPSO is not very strong for the short RNA sequence of length 468. But not too much effort should be spent by RnaPredict. When more than 50% ($\gamma = 0.5$) of the generated solutions are computed by RnaPredict the free energy of the found secondary structure is more than -126. The probability p_S to skip an index of the permutation vector when scanning the permutation vector to determine an index for a transposition operation should not too be small (not more then 30%) even for the short RNA sequence of length 468.

To gain further insight into the behaviour of Helix-PSO the size and the weight distribution of the particles target set was measured during a run of the algorithm of the RNA sequence of length 486. Figure 3 shows the number of positions in the target set. Each bar in the figure is an average over 300 runs

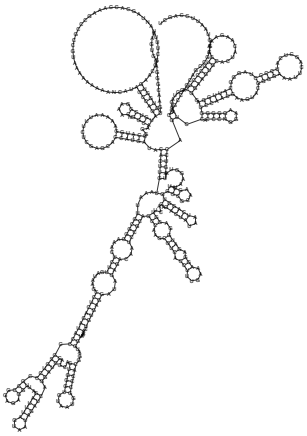

Fig. 7. Natural fold of the Aureoumbra lagunensis RNA of length 486

and over 28000 solution generations (i.e., the total run over 280000 solution generations was divided into 10 sections). It can be seen that the target set grows form an average size of 8 to 14 during the first 1/5 of the total run time (measured in number of generated solutions). Then, when the algorithm starts to converge the size of the target set decreases so that is contains only 4 solutions at end of the run. The distribution of the probabilities to be chosen for the positions in the target set (which is determined by the distribution of the weights) is shown in Figure 4. The positions were grouped with respect to their probability to be chosen into groups with a probability in the following intervals $[0, 0.001]$, $[0.001, 0.005]$, $[0.005, 0.01]$, $[0.01, 0.02]$, $[0.02, 0.05]$, $[0.05, .1]$, and $[0.1, 1]$. It can be seen that the target set contains several positions with a small probability to be chosen at the beginning of a run so that a particle moves potentially into many different directions. Later during a run when the algorithm converges most positions in the vector have a height probability to be chosen.

HelixPSO uses a greedy strategy for the movement of a particle so that the transpositions of helices in the permutation vector of a particle are accepted only if the free energy of the particle decreases. Figure 5 shows the number of positions that are accepted during different parts of a run of Helix-PSO. It can be seen that the number of accepted solutions decreases during the run from about 2500 (per 28000 generated solutions) to about 500 after 1/3 of the run time. At the end of the run only very few solutions are accepted after a move because it has become very difficult to find a position that is better than the current position of a particle. It is interesting that HelixPSO can still improve its best so far found secondary structure even when only few new positions are accepted during the end of the run. But this might also indicate a potential for further improvement of HelixPSO because some particles might stuck into local minima.

Figure 6 shows the best secondary structure of the RNA of length 486 that was generated by HelixPSO over 300 runs. Clearly, when compared to the natural fold of the RNA that is shown in Figure 7 there many differences. Since the aim of H-HelixPSO is to find low energy secondary structures it can not be expected that the natural fold is matched exactly. Two other secondary structures that were the best found in two of the 300 runs of H-HelixPSO are in Figure 8. The figure illustrates that there is still variation in the secondary structures that are found by H-HelixPSO. But it has to be noted that not all runs found different secondary structures some structures were found in several runs.

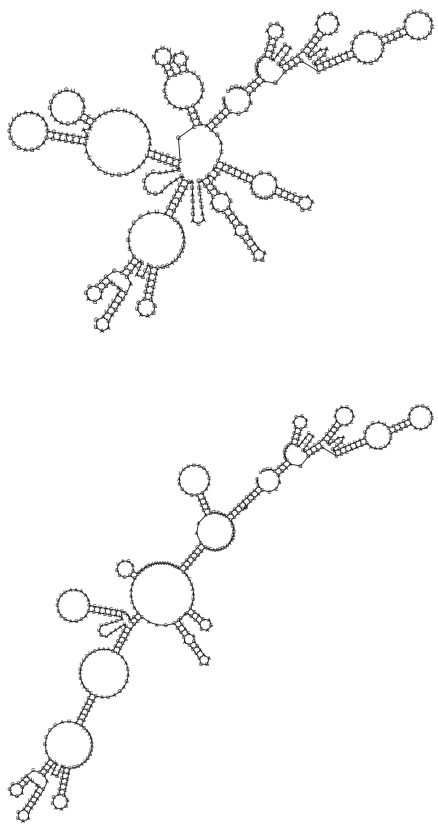

Fig. 8. Two best found RNA secondary structures of H-HelixPSO for the Aureoumbra lagunensis RNA of length 486 for two randomly chosen runs (out of the 300 runs)

VIII. Conclusions

In this paper we have proposed a new Particle Swarm Optimization (PSO) algorithm called HelixPSO for finding minimum energy RNA secondary structures. Helix-PSO uses a multiple swarm approach. Each particle has a target set of reference positions which are used to define the direction of movement of a particle. It was shown experimentally that HelixPSO profits when the starting positions of the particles are computed by a genetic algorithm (the RnaPredict genetic algorithm was used for this). The corresponding hybrid version of HelixPSO is called H-HelixPSO. It was shown experimentally on four standard RNA sequences with different lengths (length 486 to length 1492) that H-HelixPSO performs significantly better than the state-of-the-art genetic algorithm

RnaPredict. Compared to the other existing PSO algorithm that has been proposed very recently for the prediction of RNA secondary structures called SetPSO, the algorithm HelixPSO found better secondary structures both in terms of free energy and correctly predicted base pairs.

For future work it will be interesting to investigate whether Helix-PSO can become a viable candidate also for base pair prediction. For this other means than only thermodynamic optimization need to be employed.

ACKNOWLEDGMENT

The authors would like to acknowledge the support of the EMBIO project, which is funded within the European "New and emerging science and technology NEST - PATHFINDER" programme.

REFERENCES

[1] J. A. Doudna, "Structural genomics of rna," *Nature Struct. Biol.*, vol. 7, pp. 954–956, 2000.
[2] M. de Smit and J. van Duin.
[3] P. M. D.R. Mills, C. Priano and B. Binderow, "Q rna bacteriophage: mapping cis-acting elements within an rna genome," *J.Virol.*, vol. 64, pp. 3872–3881, pages =.
[4] C. R. Woese and N. R. Pace, *The RNA World*, R. F. Gesteland and J. F. Atkins, Eds. Cold Spring Harbor, NY: Cold Spring Harbor Laboratory Press, 1993.
[5] M. Zuker and D. Sankoff, "Rna secondary structures and their prediction," *Bull. Math. Biol.*, vol. 46, pp. 591–621, 1984.
[6] D. H. Mathews, J. Sabina, M. Zuker, and H. Turner, "Expanded sequence dependence of thermodynamic parameters provides robust prediction of rna secondary structure," *JMB*, vol. 288, pp. 911–940, 1999.
[7] A. Walter, D. Turner, J. Kim, M. Lyttle, P. Mller, D. Mathews, and M. Zuker, "Coaxial stacking of helixes enhances binding of oligoribonucleotides and improves predictions of rna folding," *Proc. Natl. Acad. Sci.*, vol. 91, pp. 9218–9222, 1994.
[8] J. S. Jr. and D. H. Turner, "Measuring the thermodynamics of rna secondary structure formation," *Biopolymers*, vol. 44, pp. 309–319, 1997.
[9] H. Isambert and E. D. Siggia, "Modeling RNA folding paths with pseudoknots: Application to hepatitis delta virus ribozyme," *Proc Natl Acad Sci USA*, vol. 97, no. 12, pp. 6515–6520, jun 6 2000.
[10] R. Giegerich and J. Reeder, "From rna folding to thermodynamic matching including pseudoknots," Universitt Bielefeld, Tech. Rep. Technical Report 2003-03, mar 2003.
[11] M. Zuker, "Mfold web server for nucleic acid folding and hybridization prediction," *Nucleic Acids Res*, vol. 31, pp. 3406–3415, 2003.
[12] A. P. Gultyaev *et al.*, "Dynamic competition between alternative structures in viroid rnas simulated by an rna folding algorithm," *J. Mol. Biol.*, vol. 276, pp. 43–55, 1998.
[13] B. A. Shapiro and J. Navetta, "A massively-parallel genetic algorithm for rna secondary structure prediction," *J. Supercomput.*, vol. 8, pp. 195–207, 1994.
[14] F. H. D. van Batenburg, A. P. Gultyaev, and C. W. A. Pleij, "An apl-programmed genetic algorithm for the prediction of rna secondary structure," *J. theor. Biol.*, vol. 174, pp. 269–280, 1995.
[15] A. P. Gultyaev, F. H. D. van Batenburg, and C. W. A. Pleij, "The computer-simulation of rna folding pathways using a genetic algorithm," *J. Mol. Biol.*, vol. 250, pp. 37–51, 1995.
[16] G. Benedetti and S. Morosetti, "A genetic algorithm to search for. optimal and suboptimal rna secondary structures," *Biophys. Chem.*, vol. 55, pp. 253–259, 1995.
[17] B. A. Shapiro and J. C. Wu, "An annealing mutation operator in the genetic algorithms for rna folding," *Comput. Appl. Biosci.*, vol. 12, pp. 171–180, 1996.
[18] B. A. Shapiro *et al.*, "The massively parallel genetic algorithm for rna folding: mimd implementation and population variation," *Bioinformatics*, vol. 17, pp. 137–148, 2001.
[19] I. I. Titov *et al.*, "A fast genetic algorithm for rna secondary structure analysis," *Russ. Chem. Bull.*, vol. 51, pp. 1135–1144, 2002.
[20] J. H. Chen, S. Y. Le, and J. V. Maizel, "Prediction of common secondary structures of rnas: a genetic algorithm approach," *Nucleic Acids Res.*, vol. 28, no. 4, pp. 991–999, 2000.
[21] G. Fogel *et al.*, "Discovery of rna structural elements using evolutionary computation," *Nucleic Acids Res.*, vol. 30, pp. 5310–5317, 2002.
[22] H. Ogata, Y. Akiyama, and M. Kanehisa, "A genetic algorithm based molecular modeling technique for rna stem-loop structures," *Nucleic Acids Res.*, vol. 23, no. 3, pp. 419–426, 1995.
[23] K. C. Wiese and E. Glen, "A permutation-based genetic algorithm for the rna folding problem: a critical look at selection strategies, crossover operators, and representation issues," *BioSyst. Comput. Intel. Bioinformatics*, vol. 72, pp. 29–41, 2003.
[24] K. C. Wiese and A. Hendriks, "Comparison of p-rnapredict and mfold–algorithms for rna secondary structure prediction," *Bioinformatics*, vol. 22, no. 8, pp. 934–942, 2006.
[25] M. Neethling and A. P. Engelbrecht, "Determining rna secondary structure using set-based particle swarm optimization," in *IEEE Congress on Evolutionary Computation(CEC2006)*, 2006, pp. 1670–1677.
[26] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. of IEEE International Conference on Neural Networks (ICNN)*, Perth, Australia, 1995, pp. 1942–1948.
[27] R. C. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, 1995, pp. 39–43.
[28] J. Kennedy and W. M. Spears, "Matching algorithms to problems: an experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator," in *Proc. Intl. Conf. on Evolutionary Computation*, Piscataway, NJ:IEEE Service Center, 1998, p. 7883.
[29] J. Moore and R. Chapman, "Application of particle swarm to multiobjective optimization," Department of Computer Science and Software Engineering, Auburn University, Tech. Rep., 1999.
[30] J. Robinson, S. Sinton, and Y. Rahmat-Samii, "Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna," in *IEEE International Symposium on Antennas and Propagation*, vol. 1, San Antonio, Texas, jun 2002, pp. 314–317.
[31] I. L. Hofacker, W. Fontana, P. F. Stadler, S. Bonhoeffer, M. Tacker, and P. Schuster, "Fast folding and comparison of rna secondary structures," *Monatshefte f. Chemie*, vol. 125, pp. 167–188, 1994.
[32] I. L. Hofacker and P. F. Stadler, "Memory efficient folding algorithms for circular rna secondary structures," *Bioinformatics*, vol. 22, no. 10, pp. 1172–1176, 2006.
[33] M. J. Serra and D. H. Turner, "Predicting thermodynamic properties of rna," *Meth. Enzymol*, vol. 259, pp. 242–261, 1995.
[34] T. Xia *et al.*, "Thermodynamic parameters for an expanded nearest-neighbor model for formation of rna duplexes with watson-crick base pairs," *Biochemistry*, vol. 37, pp. 14 719–14 735, 1998.
[35] K. C. Wiese, A. Deschenes, and E. Glen, "Permutation based rna secondary structure prediction via a genetic algorithm," in *Proceedings of the 2003 Congress on Evolutionary Computation (CEC2003)*. IEEE Press, 2003, pp. 335–342.
[36] D. Whitley, T. Starkweather, and D. Shaner, "The traveling salesman and sequence scheduling: quality solutions using genetic edge recombination," *Handbook of Genetic Algorithms*, pp. 350–372, 1991.
[37] T. Starkweather, S. McDaniel, K. E. Mathias, L. D. Whitley, C. Whitley, R. Belew, and L. Booker, "A comparison of genetic sequencing operators," in *Proceedings of the Fourth International Conference on Genetic Algorithms*. San Mateo, CA: Morgan Kaufman, 1991, pp. 69–76.
[38] I. M. Oliver, D. J. Smith, and J. R. C. Holland, "A study of permutation crossover operators on the traveling salesman problem," in *Proceedings of the Second International Conference on Genetic Algorithms (ICGA-87)*. Lawrence Erlbaum Associates, Inc., 1987, pp. 224–230.
[39] K. Wiese and S. D. Goodwin, "Keep-best reproduction: a local family competition selection strategy and the environment it flourishes in," *Constraints*, vol. 6, pp. 399–422, 2001.
[40] J. J. Cannone *et al.*, "The comparative rna web (crw) site: an online database of comparative sequence and structure information for ribosomal, intron, and other rnas," *BMC Bioinformatics*, vol. 3, 2002.