Determining RNA Secondary Structure using Set-based Particle Swarm Optimization

Marais Neethling, and A.P. Engelbrecht, Member, IEEE

Abstract—Determining RNA secondary structure computationally, rather than manually, has the advantage of being cheaper and quicker. This paper introduces a new Set-based Particle Swarm Optimization algorithm to optimize the structure of an RNA molecule, using an advanced thermodynamic model. Results show that it is possible to use this SetPSO algorithm to optimise RNA secondary structure and produce candidate RNA conformations.

I. INTRODUCTION

The important role that RNA molecules play in living organisms is becoming more apparent with the latest research findings. RNA is responsible for a variety of different functions inside cells such as messaging (mRNA), transferral of genetic code (tRNA), translation of RNA to proteins (rRNA) and other catalytic functions [1]–[3].

RNA molecules consist of nucleotides connected together on a sugar-phosphate backbone to form a polymer. The nucleotides are called adenine, cytosine, guanine and uracil (A, C, G and U for short).

Two pairs of bases are complimentary to each other. ‘A’ is complimentary to ‘U’ and ‘C’ is complimentary to ‘G’. These complimentary bonds tend to form hydrogen bonds between themselves when they are unbound. Thus ‘A’ bases tend to form contacts with ‘U’ bases and ‘C’ bases tend to form contacts with ‘G’ bases. There is also a third pairing of bases which occurs regularly namely ‘G’ and ‘U’. This is called a wobble-pair. Although other base pairings are possible and do exist, they are not as common and are not considered in this paper. These 3 pairings are called canonical base pairs or Watson-Crick base pairs (after the discoverers of the structure of DNA).

Because RNA is single stranded, as opposed to the complimentary double strands of DNA, the molecule folds back on itself and complimentary base pairs form strong hydrogen bonds between themselves [1].

This folding gives rise to unique structures for all of the different RNA molecules. The structure of the folded molecule determines the bindings it can make and is crucial to the functioning of the molecule.

Knowledge of the structure of the RNA molecule enables the prediction of its function and allows for the design of drugs that could interact with this molecule.

Determining the structure of RNA molecules is a non-trivial task. Current physical methods include x-ray crystallography [4] and NMR spectroscopy (Nuclear Magnetic Resonance spectroscopy) [5], [6], both expensive and time consuming methods.

A different method altogether is a computational approach. A model of the thermodynamic rules that govern the folding and binding of the molecules is used to find the most likely folding (also known as a conformation) of the molecule.

This paper introduces a new optimization algorithm based on the Particle Swarm Optimization algorithm developed by Eberhart and Kennedy [7] to optimize the structure of a RNA molecule conformation.

The rest of the paper is structured as follows. Previous attempts to predict RNA Secondary Structure is discussed in section II. A brief overview of the operators in a generic PSO algorithm is given in section III. In section IV, a new Set-based PSO algorithm is introduced. The experimental approach is discussed in section V, with results given in section VI. Lastly, there is a short conclusion and mention of future work to be done in section VII.

II. RELATED WORK

A number of population-based approaches have been developed to optimize RNA secondary structures. Shapiro and Navetta [8], Batenburg et al [9] and Benedetti [10] were the first to successfully apply genetic algorithms to RNA secondary structure prediction. These algorithms mostly used crude thermodynamic models for RNA folding. One of Batenburg’s implementations simulated the folding pathway of the RNA molecule [11]. Shapiro and Navetta implemented a genetic algorithm (GA) to solve this difficult problem on massively parallel supercomputers [8]. Shapiro and Wu [12] implemented a new annealing based mutation operator specifically for RNA folding.

More recently, Wiese and Glen [13] developed a new GA with complex mutation operators and a more sophisticated thermodynamic model, called RnaPredict. RnaPredict performs on par with mfold, a dynamic programming algorithm developed by Zuker [14], considered to be the benchmark in RNA folding algorithms.

III. GENERIC PSO

This section provides an overview of a generic PSO algorithm. PSO is based on a simple model of the flocking behaviour of birds [7]. Particles are flown through N-dimensional space in order to find an optimum solution to an objective function, \( f \). Let \( x_i(t) \) represent the position of particle \( i \) at timestep \( t \). The particle’s position is updated by adding velocity \( v_i(t) \), to the current position, i.e.
\[ x_i(t + 1) = x_i(t) + v_i(t + 1) \] (1)

Where

\[ v_{ij}(t + 1) = w v_{ij}(t) + c_1 r_{ij}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[\hat{y}_j(t) - x_{ij}(t)] \] (2)

with \( v_{ij}(t) \) the velocity of particle \( i \) in dimension \( j \in [1, n_x] \) at time step \( t \), \( x_{ij}(t) \) is the position of particle \( i \) in dimension \( j \) at time step \( t \). \( y_{ij}(t) \) is the personal best position found by particle \( i \) in dimension \( j \), \( \hat{y}_j \) is the best position found by the neighbourhood of particle \( i \) in dimension \( j \), \( c_1 \) and \( c_2 \) are positive acceleration constants and \( r_{ij}(t), r_{2j}(t) \sim U(0, 1) \). The \( w \) parameter is called the *inertia weight* [15] and controls the momentum of the particle by weighing the contribution of the previous velocity.

Each particle keeps track of its own best position, \( y_i \), found so far and is called the *personal best* position or \( pbest \). Therefore, the \( y_{ij}(t) - x_{ij}(t) \) part of equation 2 is called the *cognitive component*. This component tends to direct the search of particle \( i \) in the direction of the particle’s own best experience. Similarly, each neighbourhood keeps track of the best position, \( \hat{y} \), found so far in the neighbourhood and is called the *neighbourhood best* position or \( lbest \). The \( \hat{y}_j(t) - x_{ij}(t) \) part of equation 2 is known as the *social component*. This component tends to direct the search towards the neighbourhood’s best solution.

**IV. SetPSO**

This section proposes a different version of the PSO algorithm [7], namely SetPSO which operates on mathematical sets in order to solve set-based combinatorial problems. SetPSO therefore applies to discrete search spaces. To achieve this, new operators are defined to work on sets instead of the usual continuous vector space. Other approaches that change the representation and meaning of the position and velocity vectors have been developed. Binary PSO, created by Eberhart and Kennedy [16] changed the representation of the position vector in Binary PSO into a bit-string, while the velocity is interpreted as the probability of a bit being flipped in the position string. Clerc [17] and Schoofs and Naudts [18] also modified particle swarm optimizers to solve discrete-valued problems.

The solution space for the SetPSO is discussed in section IV-A. The representation of the particle’s position is discussed in section IV-B. The redefinition of the addition and subtraction operators is discussed in sections IV-C and IV-D respectively.

**A. Solution Space**

The solutions (particle positions) generated with the SetPSO algorithm, are sets. Therefore, elements within a particle are unique. The setPSO implementation assumes a finite number of elements in the universal set. Implementations where the universal set is infinite are also possible, although the solution set \( S \) is always finite.

In order to change the vector space the setPSO algorithm operates in, only two operators need to be redefined in the algorithm, namely the

- addition operator, and the
- subtraction operator

There is no equivalent to the multiplication operator in SetPSO and thus no redefinition of it is needed. The solution to a problem is a subset \( S \subseteq U \), which is a combination of elements from the *universal set* \( U \).

**B. Particle Position**

The current position of a particle is a potential solution set. Thus, the position is a subset of elements from the *universal set* \( U \).

The addition operator and subtraction operator have their naive set theory definitions. Subsequently it is shown how each of these operators are applied to the new setPSO.

**C. Addition Operator**

The addition operator ‘adds’ together two sets. The operator is also known as the union operator in set theory. Given two sets, \( A \) and \( B \), their union is written as

\[ A \cup B \]

or

\[ A + B \]

**D. Subtraction Operator**

The subtraction operator also know as relative complement or *set theoretic difference* between two sets \( A \) and \( B \), written

\[ A - B \]

or

\[ A \setminus B \]

denotes the set of all elements which are members of \( A \) but not members of \( B \).

**E. Pseudo code algorithm**

A short pseudo code algorithm follows with more detailed discussions on the operations implemented for the setPSO. In this algorithm, the notation \( S.X_i \) is used to denote the position \( X \) of particle \( i \) in the swarm \( S \).
Create and initialize a swarm, $S$, of size $n_s$, see section IV-G

repeat
  for each particle $i = 1, ..., S.n_s$ do
    // set personal best position
    if $f(S.X_i) < f(S.Y_i)$ then
      $S.Y_i = S.X_i$
    end
    // set neighbourhood best position
    if $f(S.Y_i) < f(S.Y)$ then
      $S.Y = S.Y_i$
    end
  end
for each particle $i = 1, ..., S.n_s$ do
  update the velocity, see section IV-H
  update the position, see section IV-I
end
until stopping condition

F. RNA Representation

When an RNA molecule folds back on itself, bindings form between canonical base pairs. ‘A’ pairs with ‘U’, ‘C’ pairs with ‘G’, and ‘G’ pairs with ‘U’ (to form wobble pairs). When adjacent base pairs form bindings, it is called ‘stacking’. A stack is also known as a stem (see figure 2 for an example). Bases are numbered sequentially, starting from the 5’ end of the sequence and ending at the 3’ end of the sequence. See figure 1 for an example of numbered bases.

There are some constraints that RNA secondary structures must abide by. For any two base pairings $[i, j]$ and $[k, l]$ with $i < j$ and $k < l$ where $i, j, k$ and $l$ are simply the numbers of the bases in a sequence, then

1) The contacts must form canonical pairs.
2) Each base must pair with only one other: $i = k$ iff $j = l$
3) No pseudo-knots are allowed:
   If $i < k < j$, then $i < k < l < j$ must hold, otherwise a pseudo-knot forms ($i < k < j < l$ is an example of a pseudo-knot).

An RNA conformation can be represented as a collection of stems. Finding the correct combination of stems is the challenging part.

All possible stems for a given RNA sequence are enumerated. This forms the universal set $U$. Not all of these stems are compatible with one another when put in the same conformation $S.X$ because it might share bases with another stem or form a pseudo-knot. Stems should not be added to a conformation if the resulting conformation violates any of the above mentioned constraints.

G. Initialize particles

Each particle is initialized to a random subset of the (finite or infinite) universal set $U$ that does not violate the constraints given in section IV-F; that is,

$$S.X_i \subseteq U, \ |S.X_i| \leq |U|$$

H. Velocity Update

The velocity of a particle is actually represented by two sets of elements: The first set contains the elements that
should be removed from the current position set. In the context of RNA conformations, this set is called the open set or \( O \). The second set contains the elements that should be added to the current position set. In the context of RNA conformations, this set is called the close set or \( C \). The open set is a random subset of the current position. The elements in \( O \) will be removed from the current position in the position update step.

In this setPSO implementation, the open set is computed as follows:

- Step 1
  Add the \( pbest \) and \( lbest \) positions sets together, i.e.
  \[
  B = S.Y \cup S.Y
  \] (3)

- Step 2
  \[
  \text{for each element } e \text{ in } X \\r \sim \ U(0, 1) \\
  \text{if } r < P_I \text{ and } e \text{ not in } B \\
  \text{add } e \text{ to } O 
  \]
  end for

- Step 3
  \[
  \text{for each element } e \text{ in } B \\r \sim \ U(0, 1) \\
  \text{if } r < P_C \text{ and } e \text{ not in } X \\
  \text{add } e \text{ to } C 
  \]
  end for

- Step 4
  \[
  \text{for each element } e \text{ in } U \\r \sim \ U(0, 1) \\
  \text{if } r < P_R \\
  \text{add } e \text{ to } C 
  \]
  end for

The parameters \( P_I \), \( P_C \) and \( P_R \) are discussed in section IV-J.

The close set is a random subset of the target position plus a random subset from the universal set. The elements included in the close set, which is taken from the target set, will be added to the current position of the particle in the position update step. This tends to move the current particle towards the target position. The target position is a combination of the \( lbest \) position and the particle’s own best position \( pbest \). In order to introduce diversity, random elements from the universal set should be added to the close set. If this is not done, the particles would explore only combinations of the elements with which they were initialised.

I. Position update

The objective of the position update is to create a new solution. In this process the current position set \( S.X_i \) for particle \( i \), is modified to produce a new position \( S.X_i' \). The position update occurs in two steps.

The first step removes all the elements in the open set \( O \) from the current position \( S.X_i \) to give an intermediate value \( S.X_i'' \).

- Step 1
  \[
  S.X_i' = S.X_i - O 
  \]
  The second step adds all the elements in the close set \( C \) to the current position set to produce

- Step 2
  \[
  S.X_i'' = S.X_i' \cup C 
  \]

Recall that the open set \( O \) contains elements that are in the current position \( S.X_i \). That means these elements are removed from the current position. Similarly, the close set \( C \) contains a combination of elements from the personal best, neighborhood best and universal sets.

An optimised addition operator can be implemented where the elements in the close set \( C \) are added in a greedy manner to \( S.X_i'' \). The elements that increases the fitness of \( S.X_i'' \) the most, are added to \( S.X_i'' \) first.

J. Parameters

The performance of the SetPSO is influenced by three parameters, namely the closing probability, \( P_C \), the random add probability, \( P_R \), and the entropy weight, \( P_I \).

The closing probability parameter, \( P_C \), is analogous to the social and cognitive component parameters in the original PSO. \( P_C \) controls the influence of both the neighboring particles’ as well as the particle’s own memory, on the position of the particle. \( P_C \) determines the probability of an element, contained in the particle’s \( pbest \) or in the \( lbest \) neighbor, to be added to the close set of the particle’s velocity.

The random add probability, \( P_R \), controls the probability that new random stems are added to the close set of the particle’s velocity. Thus, \( P_R \) determines the amount of diversity introduced to the solutions. The higher the value of \( P_R \), the better the chance that a randomly selected stem from the universal set, \( U \), will be added to the close set \( C \).

The entropy weight parameter, \( P_I \), is analogous to the inertia weight in the original PSO. This parameter controls the probability of stems being added to the open set of the velocity. Hence it controls the size of the open set and the ‘disruption’ that the removal of stems from the conformation will cause. The greater the probability, the greater the disruption to the solutions, which in turn introduces more diversity into the solutions.

V. Experimental Approach

This section summarizes the specific RNA sequences tested with the SetPSO, the objective function used and the parameter values used.

A. RNA sequences

Simulations where run on 4, different length, RNA sequences, namely Xenopus laevis 16S rRNA (945 nt), Drosophila virilis 16S rRNA (784 nt), Haloarcula marismortui 5S rRNA (122 nt) and Saccharomyces cerevisiae 5S rRNA (118 nt). These sequences where obtained from the Comparative RNA Website [19]. The sequences where chosen
to represent different complexities. Longer sequences gives rise to much more complex search spaces. Each simulation was run for 700 iterations. For each experiment, results where averaged over 30 samples.

B. Objective function

It is well known that the native conformations of RNA molecules are close to their global minimum energy state. Therefore, the objective function is a thermodynamic energy function which calculates the free energy, \( \Delta G \), of a structure. The aim is to minimize the free energy of the structure which results in an increase in stability of the structure. The free energy function used to evaluate the free energy of an RNA structure, is part of the Vienna RNA package [20].

C. Parameter Selection

For the experiments conducted for this paper, a PSO with a Von Neumann neighborhood topology was used. This topology has shown in numerous experiments to give superior results compared to other neighborhood topologies in PSO optimizations [21], [22].

For the entropy parameter \( P_I \), a linear decreasing value was used, starting with a higher value 0.9 and decreasing 0.1 during the experiment. The linear decreasing entropy allowed for greater diversity (more disruption) in the beginning stages of the search and better refinement (less disruption) in the later stages of the search.

For the other two parameters, \( P_C = 0.6 \) and \( P_R = 0.5 \) were used, slightly biasing the search in favor of previous best solutions (personal and neighborhood) as opposed to new solutions. More work needs to be done to empirically determine the influence of these parameters on the performance of the algorithm.

The swarm size was set to 50 particles.

VI. EXPERIMENTAL RESULTS

Table I summarises the results for the 4 experiments. Section VI-A discusses the results and section VI-B compares the results of SetPSO to results obtained with mfold.

A. Results

Table I shows that the mean free energy, \( \Delta G \), obtained for \( S.cerevisiae \) is \(-53.4 \text{ kcal/mol}\) with a standard deviation of 0 after 700 iterations. Forty base-pairs were predicted in total, of which 28 pairs were correctly predicted. The number of base-pairs in the know structure for \( S.cerevisiae \) is 37, of which 2 pairs are non-canonical and can therefore not be predicted by the current SetPSO implementation.

The mean free energy for \( H.marismortui \) after 700 iterations was \(-48.4 \text{ kcal/mol}\). 33 base-pairs were predicted, of which 16 was correctly predicted. The \( H.marismortui \) known structure contains 38 base-pairs, of which 4 are non-canonical base pairs.

For the \( X.laevis \) sequence, a mean free energy of \(-173.3 \text{ kcal/mol}\) was predicted with a standard deviation of 4.4. Of the 225.53 ± 6.44 mean predicted pairs, 57.83 ± 11.98 were correctly predicted. \( X.laevis \) contains 251 base-pairs in the known structure, of which 22 are non-canonical.

Lastly, the mean free energy predicted for the \( D.virilis \) sequence was \(-105.75 \text{ kcal/mol}\) with a standard deviation of 6.76. Only 29.73 ± 9.47 of the base-pairs from a total of 241.77 ± 6.76 predicted base-pairs, were correct. The known structure for \( D.virilis \) contains 233 base-pairs, of which 11 are non-canonical.

Figure 3 shows the known conformation for \( S.cerevisiae \) 5S rRNA while figure 4 displays the predicted structure for \( S.cerevisiae \) 5S rRNA.

For the two shorter RNA sequences, \( S.cerevisiae \) and \( H.marismortui \), the swarm quickly converged (after only 14 and 18 iterations respectively) on the same minimum, every single run (see figure 5).

For the longer sequence \( X.laevis \), the fitness, \( \Delta G \), of the best particle decreased rapidly in the beginning and continued to decrease (albeit slower) over the 700 iterations (as shown in figure 6).

It is interesting to note that although the minimum free energy (refer to figure 6) of the molecule continued to decrease, the number of predicted base pairs (refer to figure 7) and the number of correctly predicted base pairs (refer to figure 8), reached a maximum. As the free energy continued to decrease, the number of predicted pairs and the accuracy got worse again.

Figure 5 shows this phenomenon of worsening accuracy for simulations on \( S.cerevisiae \) as well. In fact, the worsening of the accuracy was noted in all the experiments on different RNA sequences. Numerous findings have shown that the native conformation of an RNA molecule is not always at the lowest energy state [23]–[25].

Fig. 3. The known secondary structure for \( S.cerevisiae \) 5S rRNA.
<table>
<thead>
<tr>
<th>Sequence</th>
<th>$\Delta G$ kcal/mol</th>
<th>Pairs Predicted</th>
<th>Pairs Correct</th>
<th>Pairs in known struct</th>
<th>Percentage Correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>S.cerevisiae</td>
<td>-53.4 ± 0</td>
<td>40 ± 0</td>
<td>28 ± 0</td>
<td>37</td>
<td>75.7%</td>
</tr>
<tr>
<td>H.marismortui</td>
<td>-48.42 ± 0</td>
<td>33 ± 0</td>
<td>16 ± 0</td>
<td>38</td>
<td>42.1%</td>
</tr>
<tr>
<td>X.laevis</td>
<td>-173.31 ± 4.4</td>
<td>225.53 ± 6.44</td>
<td>57.83 ± 11.98</td>
<td>251</td>
<td>23.0%</td>
</tr>
<tr>
<td>D.virilis</td>
<td>-105.75 ±3.57</td>
<td>241.77 ± 6.76</td>
<td>29.73 ± 9.47</td>
<td>233</td>
<td>12.8%</td>
</tr>
</tbody>
</table>

Fig. 4. *S.cerevisiae* as predicted by the SET PSO algorithm. ($\Delta G = -53.4$ kcal/mol)

Fig. 5. This graph shows how the mean accuracy of the prediction changes with the mean free energy ($\Delta G$) change for *S.cerevisiae* 5S rRNA. The bars represent the minimum and maximum values.

Fig. 6. The line depicts the mean free energy of *X.laevis* over 700 iterations with the minimum and maximum values for the fitness depicted by the shaded area.

Fig. 7. Mean number of predicted pairs (line) for *X.laevis* with the minimum and maximum number of predicted pairs shown by the shaded area.
From table I it is clear that the optimization algorithm found the global minimum (for $\Delta G$) for the short sequences very easily. Unfortunately, the global minimum was not the best possible solution as demonstrated by figure 5. A more accurate solution does exist, and the particles’ early trajectories (positions) did include that solution.

B. Comparison to mfold

The mfold algorithm by Zuker et al. is currently the benchmark RNA folding algorithm [14], [26]. It uses dynamic programming and an advanced thermodynamic model with the latest energy rules and parameters available. The results for the mfold algorithm are given in table II.

The mfold algorithm performs very well when folding the shorter sequences and finds a more accurate structure than SetPSO in all the cases. Note that the structures in table II are only the structures with the lowest energy totals that mfold returned. Usually, the most accurate structure is found in one of the 15 best optimal structures that mfold returns. Unfortunately, there is no indication of which of the 15 structures might be the most accurate.

Although SetPSO also temporarily predicts a conformation with 33 correct pairs for S. cerevisiae during simulation like mfold (see table II), it quickly discards it in favour of a less accurate folding with lower energy.

VII. CONCLUSION AND FUTURE WORK

This paper presented a new Set-based PSO, namely SetPSO, to find solutions to set-based optimization problems. The SetPSO was successfully applied to the prediction of the secondary structure of RNA molecules. It was found that, although the optimization algorithm presented here could find optimal or near optimal solutions, the objective function is not yet robust enough to give good solutions for RNA secondary structure optimization. It is not sufficient to base a prediction of the secondary structure only on free energy minimisation.

Future work will include a detailed analysis of influence of parameters on the performance of the SetPSO algorithm, on the prediction of RNA secondary structure. A better RNA thermodynamic model, which includes calculations for pseudo-knots will be implemented and analyzed.

The SetPSO algorithm has never been applied to other combinatorial and set based problems. The performance of the SetPSO, as an optimization algorithm, should be studied in order to determine its effectiveness in solving such problems.

ACKNOWLEDGMENT

The RNA structures were drawn using the jViz.RNA program [27].

REFERENCES

<table>
<thead>
<tr>
<th>Sequence</th>
<th>$\Delta G$ kcal/mol</th>
<th>Pairs Predicted</th>
<th>Pairs Correct</th>
<th>Pairs in known struct</th>
<th>Percentage Correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>S.cerevisiae</td>
<td>-53.5</td>
<td>41</td>
<td>33</td>
<td>37</td>
<td>89.2%</td>
</tr>
<tr>
<td>H.marismortui</td>
<td>-56.44</td>
<td>34</td>
<td>29</td>
<td>38</td>
<td>76.3%</td>
</tr>
<tr>
<td>X.laevis</td>
<td>-250.6</td>
<td>249</td>
<td>92</td>
<td>251</td>
<td>36.7%</td>
</tr>
<tr>
<td>D.virilis</td>
<td>-146.3</td>
<td>236</td>
<td>37</td>
<td>233</td>
<td>15.9%</td>
</tr>
</tbody>
</table>


