ELSEVIER

Biophysical Chemistry 55 (1995) 253–259

Biophysical
Chemistry

A genetic algorithm to search for optimal and suboptimal RNA secondary structures

Giorgio Benedetti, Stefano Morosetti *

Dipartimento di Chimica, Università di Roma 'La Sapienza', P. le A. Moro 5, 00185 Rome, Italy

Received 28 September 1994; accepted 11 November 1994

Abstract

Genetic algorithms are a search method used in solving problems by selection, recombination and mutation of tentative solutions, until the better ones are achieved. They are very efficient when the 'building block' hypothesis is effective for the solutions, which means that a better solution can be obtained by assembling short 'motifs' or 'schemata' that can be retrieved in some other worse solutions. The additive nature of the secondary structure free energy rules suggests the validity of this hypothesis, and therefore the likely power of a genetic algorithm approach to search for RNA secondary structures. We describe in detail an original genetic algorithm specific for this problem. The sharing function used to obtain differentiated solutions is also described. It results in a greater effectiveness of the algorithm in retrieving a large number of suboptimal RNA foldings besides the optimal one. RNA sequences of different length are used to test the method. The PSTV viroid sequence has been studied.

Keywords: Genetic algorithm; RNA folding; RNA secondary structure; Computer analysis

1. Introduction

1.1. The problem

RNA molecules have diverse biological functions, that presumably require well-defined three-dimensional conformations [1]. The complexity of the life cycle (e.g. in the viroids) [2] or the implication in the regulatory mechanisms [3], suggest in some cases the intervening of different tertiary structures. Predicting secondary structure first and then proceeding on to tertiary structure could be a fruitful approach.

The problem of determining RNA secondary structures is a multimodal problem, that is a great number of local solutions exists. Computing suboptimal foldings is necessary because of the uncertainties inherent in the thermodynamic data, and the loss of the three-dimensional interactions. Such research can be difficult because the number of the suboptimal foldings can be very large, and they are often local alternatives in the base pairing rather than structures very different in branching.

1.2. The approach

The main available computer approaches are recursive and combinatorial ones [4]. The last ones are usually used together with heuristic criteria to avoid

* Corresponding author.

SSDI 0301-4622(94)00130-8

the 'combinatorial catastrophe', that is the exponential growth of the possible foldings.

We tried to find optimal and suboptimal foldings of the PSTV viroid sequence that shows the abovesaid difficulties. The viroids are single stranded RNA molecules [5]. They have been shown to exist as rod-like structures [6]. The existence in the viroids of conserved hairpins [7] which are not involved in rod-like structure formation, suggests the idea of other structures important in their life cycle. Therefore it could be considered a good test for the algorithm to search for secondary structures.

We tried to fold the PSTV sequence with our previous combinatorial method [8], but the foldings obtained were local alternatives of the optimal rod structure.

Zucker also met with the same difficulties using his algorithm designed to find the best structures containing single given base pairs [9]. The algorithm is a widening of its original recursive one [10], but it is limited to a subset of foldings: those characterized by only one prescribed base pair.

Searching a new approach, we estimated the genetic algorithms [11,12] as a promising one. Genetic algorithms are a very efficient searching method used for solving problems by selection, recombination and mutation of tentative solutions, until the better ones are achieved. This strategy mimics the fundamental rules of natural genetic evolution. Genetic algorithms have been used in a wide variety of problems that range from the design of aircraft to the modelling of biological systems [13,14]. Its generality derives from the efficiency in solving problems when the 'building block' hypothesis is effective for the solutions, which means that a better solution can be obtained by assembling short 'motifs' or 'schemata' that can be retrieved in some other worse solutions. Its efficiency resides in the implicit parallelism, that is in the possibility of manipulating a large number of possible solutions in parallel.

There are two characteristics of genetic algorithms that seem appropriate to our problem. (1) The 'building block' hypothesis seems to be very suitable to the additive rule used in calculating the free energy of the secondary structures [15]; (2) a sharing function based on the 'distance' between the individuals is a well-established procedure in multimodal problems [11].

Projecting a genetic algorithm designed to solve a particular problem, implies the following definitions: the coding of the solutions (the individuals), and the fitness function that supplies the numerical evaluation of the goodness (the adaptation) of the individuals about the problem (the environment). Moreover, it is often necessary to specify operators (rules that modify the individuals) specific to the problem.

The next section deals with the main definitions and operators that form the genetic algorithm specific for the problem of searching RNA secondary structures.

2. Method

2.1. Individuals representation

A secondary structure without knots is an individual, and it is stored as a collection of the helixes with which it is built. If n is the number of all possible helixes for the sequence under consideration, it can be thought as a string of n bits (see Fig. 1a), where each position is associated with a helix, and 1 indicates its presence while 0 its loss. This is a representation with a low cardinality alphabet, which is more effective in genetic algorithms [11].

The structure is really stored as a collection of the

List of helixes

1)	7	1	72
2)	5	2	42
3)	6	3	49
4)	2	9	50
5)	4	10	25
6)	3	11	22
7)	5	12	35
8)	3	17	29
9)	2	23	33
10)	5	27	43
11)	3	38	47
12)	5	49	65
13)	4	50	73
14)	3	52	64
15)	3	54	67

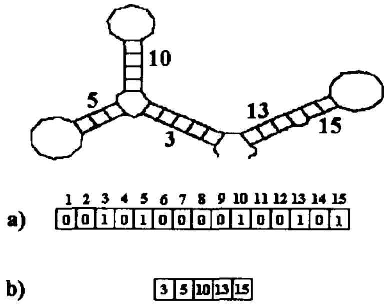

Fig. 1. Secondary structure and its record representations. Numbers 1 to 15 indicate the helixes in an ordered list. The numbers in the list indicate respectively: order number of the helix; length of the helix, in term of number of base pairs forming it; first nucleotide of the 5′ strand of the helix, in the sequence numbered from 5′ to 3′ end; final nucleotide of the 3′ strand of the helix. In record (a) the bit in position i indicates the presence (1) or the absence (0) of the helix i. Record (b) is the compressed representation really used in the algorithm.

helixes that build it, to reduce the storage requirements (see Fig. 1b).

2.2. Population size

Larger populations lead to better final results because of the larger pool of diverse schemata available [11].

In our case the population size is also dependent on the number of possible helixes, and these are dependent on the length and base composition of the sequence. The criterion is to obtain the richest variety of helixes in the initial population.

In the practical case of the PSTV viroid (359 bases), we obtain good results with a small number of iterations (200), using a large population (1000 individuals).

2.3. Starting population

The individuals of the starting population are generated by a random choice of the helixes. The random choice helix is added to the 'growing' individual only if it does not knot with the helixes as yet present. The number of random trials is a fixed one (e.g. 100), but the individuals are collections of a variable number of helixes, because of the incompatibility among them. Only structures with a negative free energy are accepted [15].

2.4. Fitness function

The fitness function is an estimate of the 'goodness' of the individual, and it determines its probability of reproduction.

We used the free energy calculated by following Freier's rules [15], and we divided it by a sharing function defined in Section 2.5. The fitness function so obtained is scaled by following a classical approach in genetic algorithms, to have a predetermined multiplicative reproduction coefficient C_m for the best member of the population (see also Section 2.6). Values of $C_m = 1.2$ to 2.0 have been used successfully [11]. We often use the middle value $C_m = 1.6$.

The scaled fitness function will be minimized during the iterative algorithm.

2.5. Sharing function

Our aim is to obtain structures which are as different as possible, and not local variants of the same individual. We use the 'distance' between two structures, defined as one minus the fraction of base pairs in common.

If n_b is the greatest of the numbers n_i and n_i of base pairs in i and j structures respectively, and n_{ij} is the number of base pairs in common between them, the distance d_{ij} is:

$$d_{ij} = 1 - n_{ij}/n_b \tag{1}$$

d_{ij} is 0 for identical individuals, and 1 for completely different ones.

The sharing function s_i of the i structure is the sum of all its distances with all the individuals of the population (including itself):

$$s_i = \sum_j (1 - d_{ij}) = \sum_j n_{ij}/n_b \tag{2}$$

The values of the function s_i can be from 1 (one individual completely different from the others) to n_p (all the individuals are identical, with $n_p =$ population size).

2.6. Reproduction

The random choice of the individuals for the next generation is weighted by the scaled fitness function. As a consequence, the structures with lower free energy and more dissimilar will be selected more frequently, with a maximum frequency determined by C_m.

A fraction p_c of the selected structures creates new individuals following the crossover operator (see Fig. 2a), whereas 1-p_c fraction survives without changes. We often use a value $p_c = 0.6$ (see Fig. 2b).

2.7. Crossover

The generation of two new individuals is performed by a crossover mechanism applied to two old individuals: there is a random choice of a point of crossover, and there is an exchange of the bits following this point between the two individuals. This mechanism is the 'heart' of the genetic algorithms, and it runs very efficiently in optimizing the problem when the solutions can be thought as obtained by assembling blocks that contribute in a rather independent way to the fitness function. This is well known as the 'building block' hypothesis [11]. In optimizing free energy of RNA secondary structures, this hypothesis seems well appropriate since the additive rule of the free energies. To enhance such characteristic we perform the crossover in the following manner: we randomly select a helix in an individual (e.g. helix *n*.13 in Fig. 2a), and subsequently all the helixes that are included in it (helix *n*.15 in Fig. 2a). Afterwards we determine in the second individual the helixes that must be displaced to make place to the helixes coming from the first individual (helixes 1 and 12 in Fig. 2a), and then the two groups of helixes are exchanged. In this way each individual is split in two substructures, its energy is the sum of the energies of the two parts, and the crossover involves the exchange of substructures (and their related energies) between the two individuals (see also Fig. 3 for a better detail). Then the optimal and suboptimal structures are obtained by assembling optimal substructures.

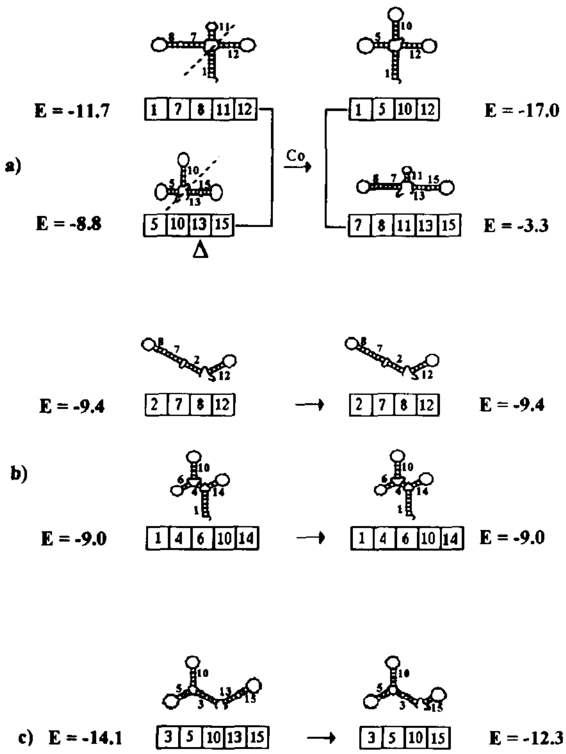

Fig. 2. Schematic representation of the main operators of the genetic algorithm for RNA secondary structures search. (a) Reproduction with crossover: selection of two individuals (on the left), and application of the crossover mechanism to generate two offspring structures (on the right). (b) Reproduction without crossover: selection of two individuals and their inclusion in the new generation, without changes in the secondary structures. (c) Mutation: random change of an individual, that can be a deletion (as shown in the figure for helix *n*.13), an insertion or a substitution of a randomly chosen helix. Δ indicates the helix randomly chosen in the crossover step. The hatched lines represent the crossover sites. See text for further details.

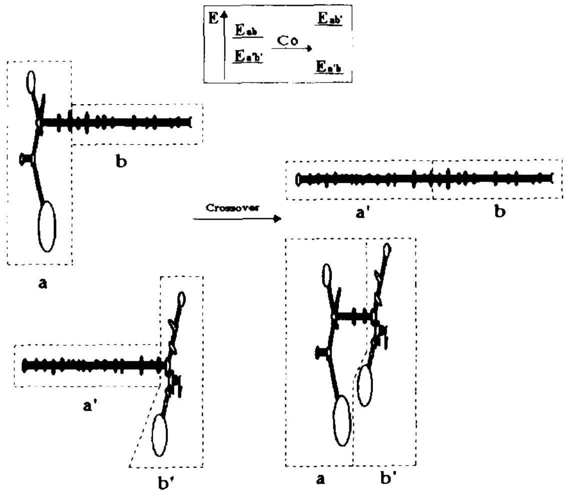

Fig. 3. Detail of the crossover mechanism between two individuals. In the upper box the action of the crossover (Co) on the energies of the structures, is graphically shown.

2.8. Mutation

This operator is a random change with a low frequency of the bits of the individuals. In our case, if n_p is the population, we choose at random a low fraction of individuals (e.g. 5%), and we operate on them one mutation. This can be an addition of a new helix, a removal of an old one (see Fig. 2c) or a substitution of a present helix with a new one randomly chosen. All of these can be seen as the exchange between the values 1 and 0 of the bit that indicates the presence (1) or the absence (0) of the helix in the structure.

The operator protects the genetic algorithms against an occasional and irrecoverable loss. In our case the loss means the absence in all individuals of a particular helix: it cannot be recovered without the presence of the mutation.

Mutation is particularly important in this application of genetic algorithms, because the short helixes, which moderately contribute to the stabilization of the structures, can be lost in the initial searching for the better structures, but they are important for the refinement of stable ones, in the final stages of the search.

2.9. *Injection of new structures*

In the beginning of the search there are structures with a small number of helixes. In this situation, when the crossover mechanism is applied, it can occasionally happen that all the helixes of one structure are selected to be shifted to a region free of helixes of the other structure. Then one of the new structures is the 'sum' of the two parents, but the other has completely lost the helixes. This will be replaced by a new structure generated with the criteria illustrated above (see Section 2.3).

2.10. *Mutation of positive free energy structures*

The crossover can generate individuals with positive free energy. Here an attempt is made to introduce a substitution that brings the free energy to a negative value. A helix of the individual is chosen at random, and the program looks for a helix that can take its place, making negative the energy of the individual. If all the possible helixes are tried without success, the structure is substituted with a new one, following the above approach (see Section 2.9).

2.11. *Mutation of duplicate structures*

Identical structures can be generated during the search. We introduce the mutation of the duplicate copies, with the same mechanism of the mutation of positive free energy structures (see Section 2.10), but here our attempt is to obtain a better free energy. If it is not possible, the structure remains unchanged. This operator contributes to the variety of present helixes, and it is very effective in shortening the search. In fact, its suppression results in an increment of a factor from 2 to 3 of the number of required cycles of iteration.

2.12. *Iteration*

The reproduction, crossover, mutation and injection steps are iterated until stable best and mean values of the fitness function are reached.

2.13. *Drawings*

We used the computer program SQUIGGLES of the GCG software package [16,17] to draw some parts of our figures.

3. Results and discussion

A new genetic algorithm approach to search for RNA secondary structures has been described.

We used sequences of different length as a test, for which hypotheses of secondary structure exist or are experimentally known. They are the tRNA^{Phe} [18] and Human U2 snRNA [19–22] sequences. The structures obtained are shown in Figs. 4 and 5.

In all sequences reported above, we obtain the optimal free energy secondary structure and a rather wide group of suboptimal ones.

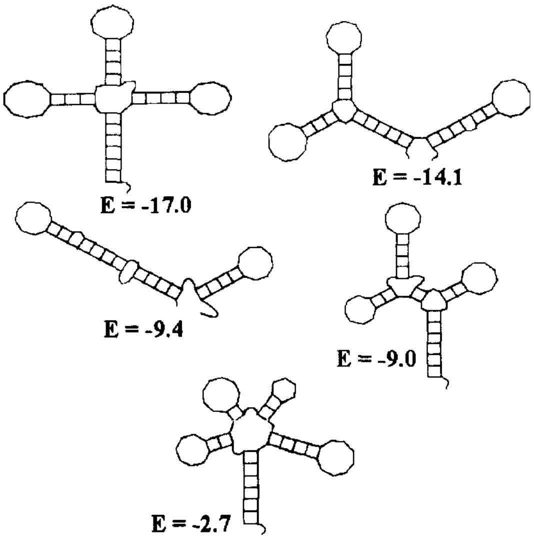

Fig. 4. tRNA^{Phe} sequence. Optimal and suboptimal free energy secondary structures. The more different foldings among the suboptimal ones are shown. Free energies are reported in kcal/mole.

It is noteworthy that foldings that look very different in branching, are present among the suboptimal structures, and that they differ extensively in constitutive helical regions. The best examples are the human U2 snRNA structures which have very similar energies despite having very different foldings. In selecting them we chose a free energy 'window' from the best value, which corresponds to the error attributed to the energy model [23].

Our approach used for the PSTV viroid, gives the optimal and a large number of suboptimal foldings. Among them can be retrieved highly branched foldings very different from the rod structures. Some suboptimal structures are shown in Fig. 6. Their free energies are in the range of 15% from the best one. Therefore this method could be used to propose the foldings involved in the life cycle of the viroids that are a 'hard test' for the existing approaches.

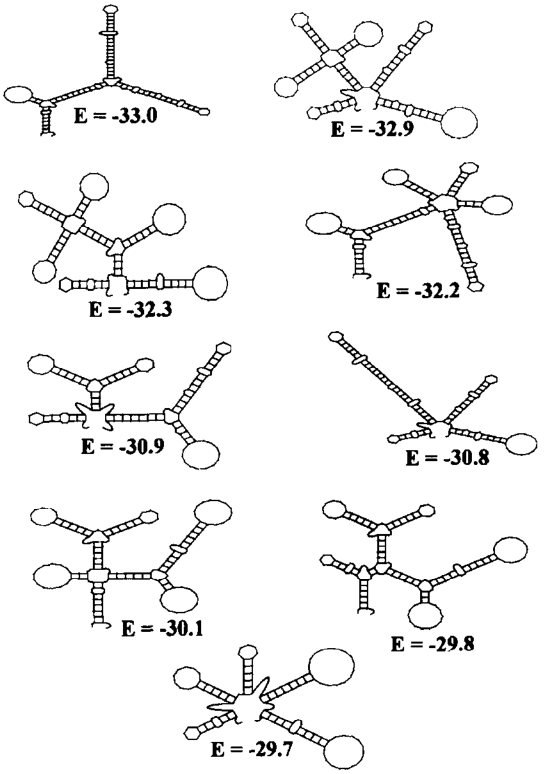

Fig. 5. Human U2 snRNA sequence. Optimal and suboptimal free energy secondary structures. The suboptimal structures are in a range of 10% from the best one. Only the foldings which are different are shown. Free energies are reported in kcal/mole.

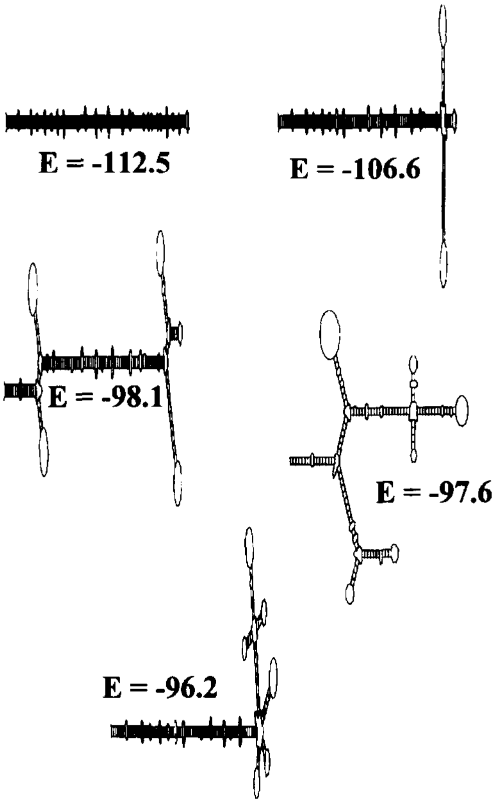

Fig. 6. The PSTV viroid sequence. Optimal and suboptimal free energy secondary structures. The suboptimal structures are in a range of 15% from the best one. Only the different foldings obtained are shown. Free energies are reported in kcal/mole.

The genetic algorithm approach seems very effective about the quickness of convergence. In spite of the complexity of the problem, 50 to 200 cycles are sufficient to reach convergence in the above-mentioned cases.

Experimental data (enzymatic or chemical cleavages, compensatory mutations) can be easily introduced in the algorithm, since they require only modification in the fitness function.

Acknowledgements

We thank Professor P. De Santis and Dr. R. Mastrangeli for helpful discussions. This work was supported by Ministero della Università e della Ricerca Scientifica e Tecnologica, Grant 92 20902 444.

References

[1] B. Wimberly, G. Varani and I. Tinoco, Jr., Current Opinions in Structural Biology, 1 (1991) 405–409.

[2] T.O. Diener, Proc. Natl. Acad. Sci. USA, 83 (1985) 58–62.

[3] P.M. Bingham, T.-B. Chou, I. Mims and Z. Zachar, Trends Genet., 4 (1988) 134–138.

[4] M. Gouy, in M.J. Bishop and C.J. Rawlings (Editors), Secondary Structure Prediction of RNA, Nucleic Acid and Protein Sequence Analysis, A Practical Approach, IRL Press, Oxford, 1987, pp. 259–284.

[5] P. Keese and R.H. Symons, in T.O. Diener (Editor), Physical–Chemical Properties: Molecular Structure (Primary and Secondary), The Viroids, Plenum Press, New York, 1987, pp. 37–62.

[6] D. Riesner, in T.O. Diener (Editor), Physical–Chemical Properties — Structure Formation, The Viroids, Plenum Press, New York, 1987, pp. 63–98.

[7] T.O. Diener, Proc. Natl. Acad. Sci. USA, 83 (1987) 58–62.

[8] G. Benedetti, P. De Santis and S. Morosetti, Nucleic Acids Res., 17 (1989) 5149–5161.

[9] M. Zuker, Science, 244 (1989) 48–52.

[10] M. Zuker and P. Stiegler, Nucleic Acids Res., 9 (1981) 133–148.

[11] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison Wesley, Reading, MA, 1989.

[12] S. Forrest, Science, 261 (1993) 872–878.

[13] L. Davies (Editor), Handbook of Genetic Algorithms, Van Nostrand–Reinhold, New York, 1991.

[14] R.K. Belew and L.B. Booker (Editors), Proceedings of the Fourth International Conference on Genetic Algorithms, Morgan Kaufmann, San Mateo, CA, 1991.

[15] S.M. Freier, R. Kierzek, J.A. Jaeger, N. Sugimoto, M.H. Caruthers, T. Nielson and D.H. Turner, Proc. Natl. Acad. Sci. USA, 83 (1986) 9373–9377.

[16] J. Devereux, P. Haeberly and O. Smithies, Nucleic Acids Res., 12 (1984) 387–395.

[17] P. Hogeweg and B. Hesper, Nucleic Acids Res., 12 (1984) 67–74.

[18] S.-H. Kim, in P.R. Schimmel, D. Soll and J. Abelson (Editors), Transfer RNA: Structure, Properties and Recognition, Cold Spring Harbour Laboratory, Cold Spring Harbour, New York, 1978, pp. 83–100.

[19] T. Maniatis and R. Reed, Nature, 325 (1987) 673–678.

[20] P.A. Sharp, Science, 135 (1987) 766–771.

[21] R. Reddy, D. Hanning, P. Epstein and H. Busch, Nucleic Acids Res., 9 (1981) 5645–5658.

[22] G. Benedetti and S. Morosetti, Eur. J. Biochem., 202 (1991) 241–248.

[23] D.H. Turner, N. Sugimoto, J.A. Jaeger, C.E. Longfellow, S.M. Freier and R. Kierzek, Cold Spring Harbor Symp. Quant. Biol., LII (1987) 123–133.